Author: Haazebroek, Pascal
Title: On the dynamic interplay between perception and action : a connectionist perspective
Issue Date: 2013-12-11

On the dynamic interplay between perception and action

A connectionist perspective

by

Pascal Haazebroek

On the dynamic interplay between perception and action - A connectionist perspective

Printed by Iskamp Drukkers B.V. Enschede

Layout: Pascal Haazebroek
ISBN: 978-94-6191-984-7
More information: **http://pascal.haazebroek.nl/hitec**

On the dynamic interplay between perception and action

A connectionist perspective

Proefschrift

ter verkrijging van
de graad van Doctor aan de Universiteit Leiden,
op gezag van de Rector Magnificus Prof. Mr. C.J.J.M. Stolker,
volgens besluit van het College voor Promoties
te verdedigen op woensdag 11 december 2013
klokke 10.00 uur

Door
Pascal Haazebroek
geboren te 's-Gravenhage

in 1980

Promotiecommissie

Promotor
Prof. Dr. B. Hommel (Universiteit Leiden)

Overige leden
Prof. Dr. J.M.J. Murre (Universiteit van Amsterdam)
Prof. Dr. S. Nieuwenhuis (Universiteit Leiden)
Dr. G. Wolters (Universiteit Leiden)

Table of Contents

Chapter 1
Introduction

How do we interact with our environment? We effortlessly turn door handles, reach for a cup of coffee, and use various kinds of tools and electronic equipment. But how do we coordinate our actions in response to these environmental demands? Intuitively, we first perceive an object, then we think for a very brief moment, and, finally, we perform actions on it. So, somehow in 'the thinking' our perception and action systems must 'connect'. The nature of this connection has been a central topic within the field of Cognitive Psychology (Ward, 2002). Indeed, actions that are not guided by perception would not only be inefficient but might also be rather dangerous. Moreover, coordinating perception and action is potentially very complex as natural environments offer an overwhelming number of perceivable objects and natural bodies allow for a virtually unlimited number of different responses. As the human cognitive system usually seems to cope quite well with this complexity, understanding its perception and action connection could be beneficial for developing artificial embodied cognitive systems (i.e., robots) that need to cope with similar challenges.

In this thesis I argue that perception and action planning do not represent separable stages of a unidirectional processing sequence, but rather emerging properties of highly interactive mental processes. In other words, information processing is the result of a (context modulated) dynamic interplay between perception and action.

Traditional views of human information processing

Traditionally, very much in line with the intuitive reasoning described above, responding to stimuli in our environment has theoretically been conceived as a *sequence of separable stages of processing* (e.g., Donders, 1868; Neisser, 1967; Sternberg, 1969; see Figure 1). The separation of information processing into a sequence of steps has a strong history in various theories of human information processing. Moreover, in many models and cognitive systems different steps are often realized by different modules.

For example, in their seminal work, Card, Moran and Newell (1983) describe the Model Human Processor being composed of three main *modules*: perception, cognition and action modules. Information processing is defined as a cyclic, sequential process from stimulus perception to cognitive problem solving to response execution. The perceptual system is considered to contain sensors and is responsible for coding the sensory input into symbolic representations. The cognitive system combines this symbolic input with long term memory and determines how to respond. Finally, the motor system is assumed to carry out the specified response.

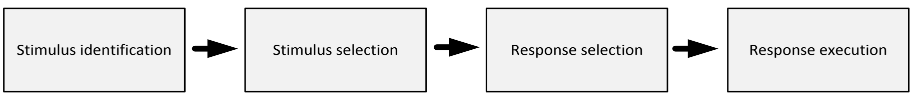


Figure 1. The perceive-think-act sequence is the basis of various theories of human information processing

In similar vein, the Seven stages of Action model (Norman, 1988) — a conceptual model of human task performance popular in the field of Human Computer Interaction — decomposes the interaction between people and their environment into the following seven *stages*: people (1) perceive the state of the world, (2) interpret their perception, (3) form evaluations based on these interpretations, (4) match these evaluations against their goals, (5) form an intention to act, (6) translate this intention into a sequence of actions and (7) execute this action sequence. Executing an action sequence subsequently results in a change in the world state which can again be perceived in the first stage.

More recently, cognitive architectures have been developed (e.g., ACT-R, Anderson, 1993; SOAR, Newell, 1990; EPIC, Kieras & Meyer, 1997) to address the challenge of computationally characterizing human information processing. Crucially, these architectures also separate processing in stages and mostly focus on the middle, cognitive steps of the perceive-think-act processing sequence. It is assumed that the first steps, perceiving and interpreting the world state, are performed relatively easily. The main focus is on comparing the world state with a goal state and deciding upon which action to take next in order to achieve the goal state. It is further assumed that once an action is chosen, its execution is easy, leading to a predictable new world state. The core mechanism used by most cognitive architectures is a production rule system (Byrne, 2003). A production rule defines the translation of a pre-condition into an action that is known to produce a desired post-condition. This can be interpreted as "IF (x) THEN (y)" rules. By specifying a set of production rules, a cognitive architecture can be given some prior knowledge resulting in response tendencies to choose those actions that eventually realize certain goals. When putting a cognitive architecture, endowed with a set of production rules, in interaction with an environment, however, conflicts between rules or unexpected conditions may present themselves. Moreover, by assuming a set of production rules, a cognitive architecture also assumes a set of action alternatives. However, when someone is interacting with a physical or virtual environment, it is often unclear which actions can be performed. Also, in certain contexts, people may not readily detect all action opportunities and action alternatives may differ in their availability, leading to variance in behavior (Kirlik, 2007). This is hard to capture in a cognitive architecture that assumes a predefined set of (re)actions.

Artificial Intelligence and Robotics

Responding to environmental demands in the environment has also been a major challenge in the fields of Artificial Intelligence and Robotics. In the 1960s – 1970s these fields started out with top down approaches focusing on robots that could reason about the world, create internal maps and figure out with hard computation how to navigate through the world. A well-known example was Shakey, a robot built in the late 1960s (Nilsson, 1984). Shakey was essentially a box on wheels with a camera. It was accompanied with an off-board computer that was programmed to make plans of 'what to do next'. The interaction with the environment started with a perception stage in which camera input was analyzed and a world model was computed in the off-board computer. Then, during the 'think' stage, the computer would go through all alternatives of what to do next, an algorithm taking minutes to compute. Finally, during the 'action' stage, essentially with eyes shut, Shakey would move a couple of feet, hoping that the world would remain stable. Then, in a new cycle, Shakey opened up its eyes again, looked at the environment, built a new world model and continued its journey. As demonstrated by rather hilarious scenes where culprits would come in and alter the environment precisely when Shakey was in its 'blind' action stage resulting in inaccurate internal models and inappropriate actions, this perceive-think-act architecture seemed to pose a problem for real world robotics: robots constructed like Shakey are limited by the need for all information from the sensors to pass through the modeling and planning modules before having any effect on the robot's actions (Brooks, 1991). As a result, Shakey could only cope with highly impoverished, static environments. Natural, dynamic environments would require too much time to construct a plan in response to ongoing, unexpected events.

In the decades that followed, some AI researchers took stronger notice of nature and observed that rather simple organisms such as bugs and insects are quite able to cope with environments that are too challenging for Shakey. Brooks (1986) proposed an activity-based decomposition of information processing. He reasoned that perception, cognition and action should be considered intertwined and suggested that a system might rather be decomposed in different behavior-producing subsystems and that *each* of these subsystems in itself forms a *complete perception, cognition, action pathway*. As these pathways may inhibit or suppress each other, such a system is able to exhibit a wide variety of complex behaviors. This approach resulted in a decade of developing insect-like robots that demonstrated much better performance in dealing with real environments than the earlier robots based on the top down approach, like Shakey. However, linking their behavior and internal representations to higher level cognitive activities such as planning, reasoning about and communicating with other robots or humans proved to be rather hard (Shanahan, 1998).

The issue of modularity is still a central topic in modern day robotics. Robots are complex systems and functional decomposition into hardware and/or software modules makes sense from an engineering point of view. In the last decade we have witnessed the dawn of highly advanced robot vision systems that recognize complex objects instantaneously (e.g., Detry & Piater, 2011) and reconstruct entire 3D scenes in internal world models (e.g., Baseki et al., 2010) . Moreover, video clips of robots showing immensely impressive behavioral repertoires (e.g., drumming, dancing, walking stairs) appear in the media weekly. How to architect and interconnect perceptual, cognitive and action systems, however, remains a matter of debate and an issue to be explicitly addressed by roboticists (e.g., the three-level architecture described in Kraft et al., 2008).

Information processing in the brain

Where traditional views on human information processing focus on the 'software' processing steps irrespective of the 'hardware' (i.e., the brain) that is assumed to perform these steps, connectionist theories stress that the structural and functional properties of the brain may have strong influences on human information processing. Indeed, the human brain does not contain a single complex central processor that does all the computations; it rather consists of billions of *simple computing units* (neurons) that are *interconnected* by trillions of connections and primarily engage in local interactions (i.e., with their directly connected 'neighbors').

Given the complexity of the brain early work on network models of cognitive performance was not aimed at modeling brain activity in complete detail. Researchers rather set out to model cognitive phenomena in systems that exhibited some of the same basic properties as networks of neurons in the brains. McCulloch and Pitts (1943) laid the foundation with networks composed of binary units and demonstrated (Pitts & McCulloch, 1947) that these networks could be used to perform pattern recognition tasks. Later approaches (e.g., Rosenblatt 1961) explored similar networks of units with connections of varying weights. In addition, following Hebb's (1949) suggestion that when two neurons in the brain were jointly active, the strength of their connection might increase, procedures came to be that allowed these networks to learn and demonstrate associative memory abilities (e.g., Taylor, 1956). The success of these early network models was, however, rather short-lived as there were strong limitations (e.g., demonstrated by Minskey & Papert, 1969) to what this type of networks could compute and serious learning algorithms were lacking. These limitations turned the focus of AI research towards symbolic models of information processing.

In the mid-1980s, however, important limitations of rule-based symbolic systems were identified (e.g., inflexibility, difficulty in learning from experience, inadequate generalization) and network-inspired approaches came back in vogue. Rumelhart, Hinton, and McClelland (1986) published their very influential *Parallel Distributed Processing* (PDP) work that essentially defined the connectionism. In the connectionist approach there is a network of elementary units, each of which has a certain degree of activation. The network is considered to be a *dynamical* system which, once provided with initial input, spreads activation among its units for a set period of *time* or until a stable state is achieved. Such a connectionist system is considered to 'perform' a cognitive task by interpreting the inputs as a problem and the resulting stable configuration of the system as the solution to that problem. Compared to the symbolic approach, that involves transformation of symbols according to specific rules, the connectionist approach focuses on causal processes by which the units spread activation to each other. Hence, information processing in connectionist networks is *distributed.*

As connectionism became increasingly popular in the late 1980s, some researchers (e.g., Fodor, 1983; Pinker, 1997) argued that connectionism actually constituted a reversion toward behaviorism (e.g., Watson, 1913) by focusing on mere input-output associations rather than addressing mental processes in terms of explicit logical algorithms. In their view, mental activity is computational; that is, performing operations on symbols. Indeed one could argue that connectionist-like hardware (i.e., the brain) may actually only implement the symbolic-like software algorithms. Hence, the mind could still very well be decomposed in separate (e.g., perception, cognition and action) subsystems. Indeed, to what extent the mind can be considered a *modular* system is still a matter of lively debate among cognitive scientists (Prinz, 2006).

Direct interaction between perception and action

Now, interestingly, empirical findings in psychology have demonstrated that parts of human information processing do not seem to involve conscious cognitive decision making. Features of perceived objects (such as location, orientation, and size) can influence actions *directly* and beyond (tight) cognitive control, as illustrated by stimulus–response compatibility phenomena, such as the Simon effect (Simon & Rudell, 1967). In the typical Simon task, stimuli vary on a spatial dimension (e.g., randomly appearing on the left or right) and on a non-spatial dimension (e.g., having different colors). Participants have to respond to the non-spatial stimulus feature by performing a spatially defined response (e.g., pressing a left or right key). Although the location of the stimulus is irrelevant for the response choice, it nevertheless influences response time and accuracy: participants respond faster (and more accurately) when the stimulus location is congruent with the response location than when the stimulus location is incongruent with the response location. This finding suggests that there is a direct interaction between stimulus perception and response planning. The Simon effect is a very robust finding, has been replicated numerous times and has been used frequently as a methodological tool to investigate perception, action, and cognitive control (for general overviews, see Hommel, 2011; Proctor, 2010).

To account for both controlled and automatic processing, various dual route process accounts have been proposed (e.g., Kornblum, Hasbroucq, & Osman, 1990; Zorzi & Umilta, 1995). These accounts propose that there is a second, direct route from perception to action that can bypass cognition, as explicitly modeled in various computational models of the Simon effect (e.g., Zorzi & Umilta, 1995; see Chapter 4 for a more elaborate discussion). Essentially, dual route accounts consider the observed direct stimulus-response interaction as an exception requiring an additional route next to the 'normal' one that does involve cognition. Moreover, they typically do not address the reason *why* some stimulus features directly influence action and others do not.

Representing perception and action using common codes

An alternative view that gives much more weight to this direct interaction between perception and action is the Theory of Event Coding (TEC, Hommel, Müsseler, Aschersleben & Prinz, 2001; illustrated in Figure 2). TEC is a general theoretical framework that addresses how perceived events (i.e., stimuli) and produced events (i.e., actions) are cognitively represented and how their representations interact to generate both perceptions and action plans. TEC holds that stimuli and actions are represented in the same way and by using the same 'feature codes'. These codes refer to the *distal* features of objects and events in the environment, such as shape, size, distance, and location, rather than to proximal features of the sensations elicited by stimuli (e.g., retinal location or auditory intensity; see Heider, 1959; Hommel, 2009). For example, a haptic sensation on the left hand and a visual stimulus on the left both activate the same distal code representing 'left'.

Crucially, these feature codes can represent the properties of a stimulus in the environment just as well as the properties of a response — which, after all, is a perceivable

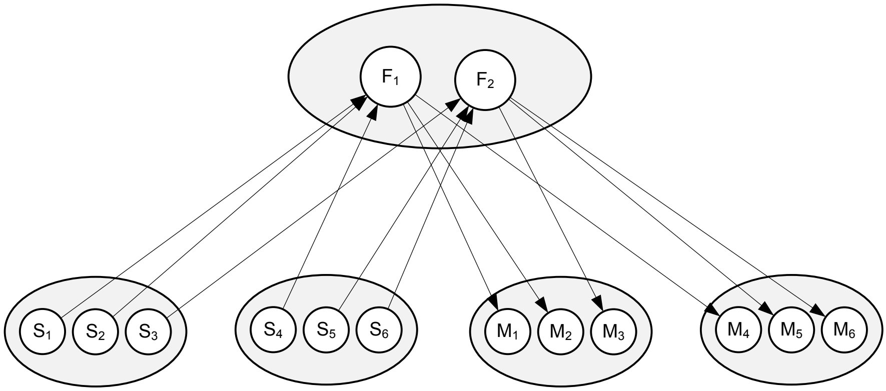


Figure 2. Sensory, feature and motor codes in TEC (adapted from Hommel et al., 2001). Multiple sensory codes can relate to the same feature codes (and vice versa). The same holds for motor codes and feature codes.

stimulus event itself. This theoretical assumption is derived from ideomotor theory (James, 1890; see Stock & Stock, 2004, for a historical overview), which presumes that actions are cognitively represented in terms of their perceivable effects. According to *the ideomotor principle*, when one executes a particular action, the motor pattern is automatically associated to the perceptual input representing the action's effects (action–effect learning; Elsner & Hommel, 2001). Based on these action-effect associations, people can subsequently plan and control (Hommel, 2009) a motor action by anticipating its perceptual effects, that is, (re-)activate a motor pattern by intentionally (re-)activating the associated feature codes. Thus, stimuli and actions are represented in a *common representational medium* (Prinz, 1990). Consequently, stimulus perception and action planning are considered to be similar processes: both involve activating[1] feature codes that represent external events.

Neuroscientific evidence for common codes at a distal feature level can be found in the response characteristics of mirror neurons in the premotor cortex (cf., Keysers & Perrett, 2004). In the macaque monkey, these neurons are active both when the monkey performs a particular action and when it perceives the same action carried out by another monkey or human, such as picking up food. Crucially, this overlap occurs at a distal representational level, that is, at the level where planned and perceived actions can be described as having the same goal or end state such as picking up an object (Rizzolati & Craighero, 2004). Also, various behavioral studies show that, in humans, action planning can actually influence object perception (e.g., Fagioli, Hommel & Schubotz, 2007; Stoet & Hommel, 2002; Wykowska, Schubö & Hommel, 2009), suggesting that perceptual processes and action processes overlap in time (see also Hommel, 1997) and influence each other.

[1] TEC also addresses how more complex cognitive codes ('event files') are created, an aspect that refers to the integra-tion of feature codes rather than their mere activation. This structure-building aspect will not be dealt with in this thesis but be left for future work.

Finally, TEC stresses the role of *task context* in stimulus and response coding. In particular, the responsiveness of feature codes to activation sources is considered to be modulated according to the task or goals at hand (the *intentional weighting principle*, Memelink & Hommel, 2013). For example, if the task is to grasp an object, feature codes representing features relevant for grasping (such as the object's shape, size, location and orientation) are assumed to be *enhanced*, while feature codes representing irrelevant features (such as the object's color or sound) appear to be attenuated (Hommel, 2010; Wykowska et al., 2009).

HiTEC connectionist model

In this thesis I aim to shed more light on the biological and computational plausibility of common representations underlying perception and action planning. To this end I have developed HiTEC, a connectionist model based on TEC. Our aim was to formulate a clear alternative to sequential models of perception and action and to develop a *minimal framework* for considering how perceptual and action processes may interact in the control of behavior. HiTEC extends and further specifies TEC's principles to account for a series of key experimental findings in a unitary theoretical framework and at a level of specificity that allows for computer simulation.

Outline of the thesis

The thesis is organized as follows. Chapter 2 presents HiTEC, the connectionist model developed to study the feasibility of common representations and interactive processing; in Chapters 3 to 5, various simulations of empirical phenomena are described. Here, the focus is on research questions that particularly challenge existing models of stimulus-response translation that assume separate modules or processing stages. Finally, general conclusions are described in Chapter 6. In this endeavor the following research questions are addressed in this thesis.

How do neuron-like representations realize stimulus-response translation?

This research question is addressed in Chapter 2. In this chapter, the HiTEC connectionist model is presented. In HiTEC, neuron-like representations are *distributed* over *multiple levels* and processing involves both feedforward and *feedback* interaction between lower and higher level representations. In addition, one of the HiTEC levels contains *common* representations; these representations are used both for stimulus perception and response planning. As a result, stimulus-response processing is fully interactive rather than in stages. The HiTEC model is used in all simulations discussed in this thesis.

How do situation-specific meanings of motor actions emerge?

In order to control its actions in response to demands in the environment the cognitive system needs to know what actions are possible and what these actions 'mean'. Various empirical findings suggest that for a cognitive system this 'meaning' is not a fixed fact; it rather depends on the (perceptual) effects within the task context. Consequently, in order to select and execute an appropriate response to a stimulus a plausible cognitive model must first learn (i.e., from experience) what the effects of its motor actions are and how to interpret these effects in the task context. How these situation-specific meanings of actions may emerge and how these meanings are used in action control is addressed in Chapter 3. Simulations in this chapter demonstrate that HiTEC allows for associating action effects with motor actions. Moreover, the strengths of these associations depend on the context allowing for the emergence of situation-specific meanings.

How and why do parts of stimulus–response translation occur automatically?
Some parts of the translation from stimulus to response are considered to occur automatically as demonstrated by stimulus–response compatibility (SRC) effects such as the Simon effect (Hommel, 2011; Simon & Rudell, 1967). How and why these effects may occur is addressed in Chapter 4. Simulations in this chapter demonstrate that HiTEC provides a parsimonious rationale for these effects, most notably in terms of the common representation level and the fact that task-relevance is considered to apply to both stimuli and responses.

How does the task context modulate stimulus-response translation?
How the task context may modulate stimulus-response translation is more explicitly addressed in Chapter 5, which includes both the simulation of an existing empirical study and a novel behavioral study and its simulation. The first simulation in this chapter demonstrates how the task context may modulate action control by means of (spatial) attention within the environment; the empirical study and the second simulation show how intentional weighting may also operate on a more abstract (distal) level.

Finally, in Chapter 6, these research questions and their interrelations are further discussed as not only perception and action are strongly interrelated, so are the different research questions addressed in this thesis.

Publications
Note that these chapters contain major parts of various articles published in the course of this research. Rather than presenting a collection of published and submitted articles divided over chapters, I have chosen to assist the reader with what I consider a more logical structure. Following this structure, one chapter is devoted to presenting the entire model, and the other chapters focus on various major aspects of the interaction between perception and action combining simulations from various articles. In my view, this structure better reflects the integrated character of the work and avoids unnecessary repetition of common or iteratively refined parts such as model implementations, theoretical background and simulation procedures.

The thesis is an integration of a number of articles I wrote in collaboration with co-authors. Note that this is reflected in the various chapters by the use of 'we' rather than 'I'. The interested reader is referred to these articles.

Haazebroek, P., & Hommel, B. (2009a). Anticipative control of voluntary action: Towards a computational model. *Lecture Notes in Artificial Intelligence, 5499*, 31-47.

Haazebroek, P., & Hommel, B. (2009b). Towards a computational model of perception and action in human computer interaction. *Lecture Notes in Computer Science, 5620*, 247-256.

Haazebroek, P., Raffone, A., & Hommel, B. *HiTEC: A Connectionist Model of the Interaction between Perception and Action Planning. Manuscript submitted for publication.*

Haazebroek, P., van Dantzig, S., & Hommel, B. (2009). Towards a computational account of context mediated affective stimulus-response translation. *Proceedings of the 31st Annual Conference of the Cognitive Science Society* (pp. 1012-1017). Austin, TX: Cognitive Science Society.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2011a). A computational model of perception and action for cognitive robotics. *Cognitive Processing*, *12*, 355-365

Haazebroek, P., van Dantzig, S., & Hommel, B. (2011b). Interaction between Task Oriented and Affective Information Processing in Cognitive Robotics. *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, *59*, 34-41.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2013). How task goals mediate the interplay between perception and action. *Frontiers in Psychology*, *4:247*.

As my PhD project was embedded into an interdisciplinary robotics project I also got the chance to collaborate with scientists from other disciplines. Some of this collaborative work has not been integrated in this thesis, even though it contains some of the ideas captured therein; the interested reader is referred to following articles.

Broekens, J. & Haazebroek, P. (2007). Emotion and reinforcement: Affective facial expressions facilitate robot learning. In *Proceedings of the IJCAI Workshop on AI for Human Computing (AI4HC'07, Hyderabad, India)* (pp.47-54).

Lacroix, J. P. W., Postma, E., Hommel, B. & Haazebroek, P. (2006). NIM as a brain for a humanoid robot. In *Proceedings of the Toward Cognitive Humanoid Robots workshop at the IEEE-RAS International Conference on Humanoid Robots 2006*. Genoa, Italy.

Spiekman, M.E., Haazebroek, P., & Neerincx, M.A. (2011). Requirements and Platforms for Social Agents that Alarm and Support Elderly living Alone. *Lecture Notes in Computer Science 7072.*, 226-235.

Chapter 2
HiTEC Connectionist Model

This chapter is an integration of major parts of the following articles:

Haazebroek, P., Raffone, A., & Hommel, B. *HiTEC: A Connectionist Model of the Interaction between Perception and Action Planning.* Manuscript submitted for publication.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2013). How task goals mediate the interplay between perception and action. F*rontiers in Psychology, 4:247*.

In this chapter we describe the HiTEC connectionist model in full detail. We start out with discussing the general cortical layering of the brain and our general connectionist modeling approach. Then we describe the specific HiTEC architecture, followed by its computational implementation. We proceed with discussing how HiTEC allows simulating behavioral studies and, finally, we compare our approach with related work and address how neuron-like representations may realize stimulus-response translation.

Cortical layering

Neurons in the primate cortex appear to be organized in numerous interconnected cortical layers. It is commonly assumed that this organization allows the brain to encode perceived objects in a *distributed* fashion. That is, different features seem to be processed and represented across different cortical layers (e.g., Cowey, 1985; DeYoe & Van Essen, 1988), coding for different perceptual modalities (e.g., visual, auditory, tactile, proprioceptive) and different dimensions within each modality (e.g., visual color and shape, auditory location and pitch). Each sensory cortical layer typically contains neurons that are responsive to specific sensory features (e.g., a specific color or a specific visual location). Cortical layers in the motor cortex contain neurons that code for more or less specific movements (e.g., the muscle contractions that produce the movement of the hand pressing a certain key, or more complex movement such as shifting one's weight to the right). Higher up in the processing stream there are cortical layers containing neurons that are receptive to stimulation from different modalities. In effect, they are considered to integrate information from different senses and modalities. Finally, the prefrontal cortex contains neurons that are involved in task-generic cognitive control (Duncan & Owen, 2000). These *levels of representation* are illustrated in Figure 3 and form the basis of the HiTEC model architecture.

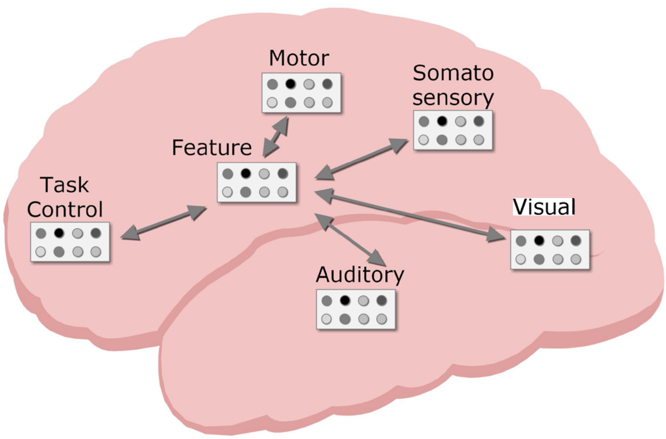


Figure 3. Tentative locations of various cortical layers in the primate brain with sensory layers in sensory regions, task control layers in the frontal lobe, motor layers in motor area and intermediate feature layers mediating between lower and higher region layers.

Crucially, cortical layers are not only interconnected by feedforward connections (i.e., from lower to higher level layers) but there are also dense neural pathways from centers of higher brain function back into perception centers (Braitenberg & Schüz, 1991; Young, 1995) suggesting *top-down influence* of higher level layers on processing within lower level layers (e.g., Prinz, 2006). This constraint of reciprocal connectivity between various levels of representation is taken seriously in the HiTEC connectionist model.

Connectionist approach

The cortical layers in the primate brain contain a vast amount of spiking neuron cells. The local interactions between these neurons are largely random, but on a group level – a neuron population – the global population activity (i.e., mean spike frequency) can be considered deterministic (Wilson & Cowan, 1972). That is, mean activation depends on various inputs and the decay of the neuron population (see Figure 4).

To model such neuron populations HiTEC follows[2] the interactive activation connectionist modeling approach (PDP; Rumelhart et al., 1986). In these connectionist models processing occurs through the interactions of a large number of interconnected elements called units. In HiTEC, these units may stand for the neuron populations described above and are organized into higher structures representing cortical layers. Each unit has an activation value indicating local activity. Processing occurs by propagating activity through the network; that is, by propagating activation from one unit to the other, via weighted connections. When a connection between two units is positively weighted, the connection is excitatory and the units will increase each other's activation. When the connection is negatively weighted, it is inhibitory and the units will reduce each other's activation. Processing starts when one or more units receive some sort of external input. Gradually, unit activations rise and propagate through the network while interactions between units control the flow of processing. Some units can be designated output units. When activations of these units reach a certain threshold the network is considered to produce the corresponding output(s).

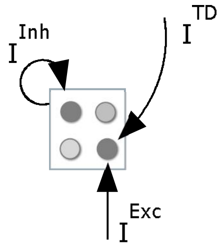


Figure 4. Cortical layer with neuron populations with various inputs (TD: top down, Inh: lateral inhibition, Exc: excitatory input)

[2] For our current work we have focused on representations and interactive processing. For this purpose, PDP principles provided the means we needed. This modeling framework, however, is not essential for TEC/HiTEC. Other aspects are probably hard to tackle within the limitations of PDP, such as binding and integration. In the future we may change to or add other modeling principles than PDP prescribes.

HiTEC architecture

HiTEC has a multiple-layer architecture (see Figure 5) and recurrent interactions at multiple levels, including feedback to lower level units. In HiTEC feedforward and feedback interactions are cooperative and lateral interactions (i.e., within layers) are competitive (see also Murre, Phaf & Wolters, 1992; van Dantzig, Raffone & Hommel, 2011). The HiTEC neural network is composed of excitatory and inhibitory neural units in each layer. The coding functions are implemented as excitatory units[3]. The inhibitory units are only involved in lateral competitive interactions; by contrast, the excitatory units can receive inputs from and send outputs to associated units in other layers, yielding cooperative interactions. Within each layer inhibitory units are activated by an associated excitatory unit and propagate inhibition to the excitatory units that implement other codes in the same layer (see Figure 6).

We now first describe the general model architecture, and then describe the model behavior and the computational specification of the network units. HiTECs general architecture contains sensory layers, feature layers, a task layer and a motor layer, as depicted in Figure 5. Each layer resembles a cortical circuitry and contains codes implemented as excitatory connectionist network units as described above. The different codes (and related units) are characterized as follows.

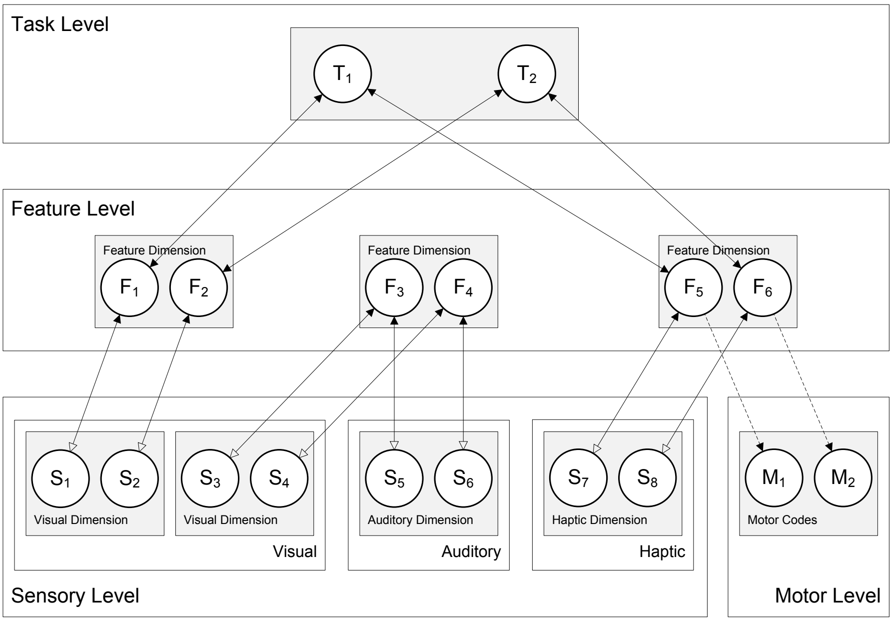


Figure 5. General computational structure of HiTEC. Codes are contained in layers at various levels, and are connected by excitatory connections. Solid lines denote fixed weights, dashed lines are connections with learned weights. Sensory codes receive modulated excitatory input from feature codes, denoted by the open arrows. Note that feature code – motor code associations are one-way connections and that feature code – task code connections are non-modulated both ways.

[3] We have opted for localist representations to keep HiTECs architecture and representations as simple as possible. There is, however, nothing that precludes the possibility that any of the codes could be distributed over many sub-codes/sub-units.

Sensory codes

In HiTEC, different perceptual modalities (e.g., visual, auditory, tactile, proprioceptive) are distinguished and different dimensions within each modality (e.g., visual color and shape, auditory location and pitch) are processed and represented in different sensory layers. Each sensory layer contains a number of sensory codes that are responsive to specific sensory features (e.g., a specific color or a specific location in the visual field). Sensory codes receive external input and feedback activation from feature codes.

Crucially, the responsiveness of sensory coding units is modulated by connected feature coding units. This is realized by making the inputs from feature units to a sensory coding unit dependent on that sensory coding unit's activation, which is primarily determined by its external stimulation. This way, a sensory coding unit cannot become highly active by mere top down input, which would be the equivalent of a hallucination.

Motor codes

The motor layer contains motor codes, referring to more or less specific movements (e.g., the movement of the hand pressing a certain key or producing a verbal utterance). Although motor codes could also be organized in multiple layers (e.g. reflecting different body parts), in the present version of HiTEC we consider only a single basic motor layer with a set of motor codes. Motor codes are activated by feature codes. When the activation level of one of the motor coding units reaches a set response threshold, the motor code is assumed to be selected and executed. Subsequent *action effects in the environment* are presented to the sensory coding units.

How motor actions are controlled is more explicitly addressed in Chapter 3. Note that our present account of motor information represents a dramatic simplification. Movements are unlikely to be represented by coherent, encapsulated motor programs (as considered by Keele, 1968) but, rather, in a rather complex, distributed fashion (Hommel & Elsner, 2009; Wickens, Hyland, & Anson, 1994). However, this simplification does not affect our main arguments and it helps keeping the model and its behavior reasonably transparent.

Feature codes

TEC's notion of feature codes (Hommel et al., 2001) is captured at the feature level by codes that are connected to and thus grounded in both sensory codes and motor codes. Crucially, the same (distal) feature code (e.g., 'left') can be connected to multiple sensory codes (e.g., 'left proprioceptive direction' and 'left visual shape'). Thus, information from different sensory modalities and dimensions is combined in one feature code representation. It is assumed that feature codes arise from regularities in sensorimotor experience, presumably by detecting co-occurrences of sensory features. The distal feature 'left', for example, could arise from perceptual experience of numerous objects that were visible and audible on the left. Future encounters of objects audible on the left activate the 'left' feature code which – by means of its connections to both 'left auditory location' and 'left visual location' – will enhance the processing of visual left locations. In other words, hearing something on the left will result in expecting to see something on the left as well, which seems to be quite useful, for example when visual sensory input is degraded. Although feature codes are considered to arise from experience, in the present HiTEC version we assume the existence of a set of feature codes (and their connections to sensory codes) to bootstrap the process of extracting sensorimotor regularities in interactions with the environment.

Since feature codes connect to both sensory codes and motor codes, they can be considered common codes in the sense of Prinz (1990), subserving both stimulus perception and response planning. When a certain feature code is used to represent a task stimulus and this same feature code is also used to represent a task response, the resulting code overlap may result in compatibility effects. Such compatibility effects are demonstrated in the simulations discussed in the next chapters, most notably in Chapter 4.

Task codes

The task layer contains generic task codes that reflect alternative stimulus-response combinations resulting from the task context. Different task codes reflect different stimulus-response choice options within the task context. Task codes connect bi-directionally to feature codes, both the feature codes that represent stimuli and the feature codes that represent responses, in correspondence with the current task context. Note that task codes themselves are task-generic (i.e., labeled 'T_1', 'T_2' et cetera); their meaning derives from their connections with specific feature codes.

The multiple-layer recurrent neural network architecture with different types of codes and the connections between associated codes is illustrated in Figure 5. Note that the connection weights can be different (asymmetrical) for corresponding 'forward' and 'backward' connections (e.g. different weights for the connection from feature codes to task codes, and the reciprocal connection from task codes to feature codes).

Basic model behavior

The presentation of a stimulus is simulated by feeding external input to the appropriate (excitatory) sensory codes. This results in a gradual increase of their activation level, which is translated into output to feature codes. Thus, activation flows gradually from sensory codes to (stimulus related) feature codes to task codes to (response related) feature codes to motor codes. Once a motor code is activated strongly enough it is assumed to lead to the execution of a motor response to the presented stimulus. The gradual passing of activation between codes in different layers along their connections is iterated for a number of simulation cycles, which allows for the simulation of reaction time (i.e., number of processing cycles from stimulus onset to response selection). Crucially, activation also propagates back from task codes to stimulus related feature codes that in turn modulate the sensitivity of sensory codes, thereby rendering an integrated processing system with both feedforward and feedback dynamics rather than a serial stage-like processing mechanism.

Ideomotor learning

In HiTEC, connections between feature codes and motor codes are learned according to the ideomotor principle (Hommel, 2009; James, 1890; Lotze, 1852). This principle states that when one executes a particular action and perceives the resulting effects in the environment, the active motor pattern is automatically associated to the perceptual input representing the action's effect. Based on these action-effect associations, people can subsequently plan and control a motor action by anticipating its perceptual effect.

In similar vein, learning in HiTEC is done by first randomly activating motor codes, not unlike the random movement behavior of newborn infants (motor babbling) or complete novices at a new task. When a motor code reaches a threshold of activation, we assume that the response is executed, resulting in perceivable changes in the environment (action effects). Perceiving these action effects constitutes stimulating the respective sensory codes; activation is subsequently propagated from these sensory codes towards feature codes (cf. Elsner & Hommel, 2001). Finally, associations are learned between these feature codes and the executed motor code. During subsequent stimulus-response translation these associations enable activation of the appropriate motor action by activating the associated feature codes. Thus, a motor action can be selected by 'anticipating its perceptual effects'. Ideomotor learning and its role in action control is addressed more elaborately in Chapter 3.

Task internalization

In behavioral experiments both stimuli and responses can have a variety of features. The task context dictates which of these features are relevant (i.e., the features to look for and to discriminate) and which are irrelevant. In HiTEC, a task instruction is implemented by connecting feature codes and task codes according to the actual task rules in terms of stimulus features and response (i.e., action effect) features. This procedure allows the task instruction to be readily internalized. An example task instruction "when you hear a high tone, press the left key" would then be implemented as connections from 'High' to 'T_1' and from 'T_1' to 'Left' and 'Key'. During the subsequent stimulus-response translation, these connections modulate the responsiveness of feature codes to bottom-up input from stimulated sensory codes and through these connections activation is propagated towards feature codes associated to the proper motor responses in accordance with task demands (cf., Miller & Cohen, 2001). This way, appropriate goal oriented behavior can take place within a certain task context.

In the present HiTEC version, these connections between feature codes and task codes units are set by hand in correspondence with the verbal task instruction. However, it is conceivable that these connections arise from external or internal verbal or nonverbal (self-) instruction and are maintained due to internal motivational drives. We hypothesize that feature codes could be accessed by means of verbal labels and that receiving a task instruction would activate these feature codes (e.g., Bargh & Gollwitzer, 1994; Hommel & Elsner, 2009; Logan & Bundesen, 2004) and connect them to generic task codes (i.e., some sort of internal simulation of the translation from stimulus features to response features). Note that apart from this instruction based wiring we do not assume any other type of task-specific addition to the model. That is no additional 'task inputs' or biases in code dynamics are required to control stimulus-response translation.

Computational implementation

HiTEC codes are implemented as (excitatory) neural network units, characterized by an activation level. These units, which may stand for neuronal groups, receive excitatory and inhibitory inputs from other units and background noise. Excitatory inputs can either be voltage independent or voltage dependent, i.e. with a modulatory role dependent on the voltage ('activation') of the receiving unit. Indeed, cortical feedback connections are generally voltage dependent, i.e. necessitate a sufficient level of feedforward (stimulus related) synaptic input to be effective. In addition, the activation of the units is characterized by a decay rate, so that in case of absence of any input the activation will decay exponentially towards a resting level. Units in the sensory layers can also receive an external (stimulus related) input. Thus, on every cycle unit activations are updated according to the following equation:

$$A_i(t+1) = (1-d_a) \times A_i(t) + \gamma_{exc} \times Exc_i \times (1 - A_i(t)) + \gamma_{inh} \times Inh_i \times A_i(t) \qquad (1)$$

In this equation, d_a is the activation decay rate, $A_i(t)$ is the activation level of unit i at time t, Exc_i is the sum of its excitatory input, Inh_i is its inhibitory input and both γ_{exc} and γ_{inh} are scaling terms. Note that both excitatory and inhibitory inputs are scaled in a way that the unit's activation may take on any real value between 0.0 and 1.0. The excitatory input is computed as follows:

$$Exc_i = ExcVI_i + ExcVD_i + Ext_i + Noise_i \qquad (2)$$

Here, $ExcVI_i$ is a voltage independent ('non-modulatory') input from other units in the network, which does not depend on the activation of the receiving unit; $ExcVD_i$ is a voltage dependent input, which is instead dependent on the activation of the receiving units (implicitly related to the membrane potential of receiving neurons). These different excitatory inputs stand for different synaptic currents in cortical networks: feedforward signaling takes place by voltage-independent synaptic currents, and feedback signaling by modulatory voltage dependent currents (e.g., Dehaene et al., 2003; Raffone & Pantani, 2010; Tononi, Sporns, & Edelman, 1992). Ext_i is input from external stimulation (only for units in the sensory layers) and $Noise_i$ is a noise term. This noise term is determined by drawing a random value from a Gaussian distribution[4] at each update cycle and for each unit independently.

[4] Determining the noise term by drawing from a Gaussian distribution sometimes (with our parameters, in < 5% of the cases) results in a negative value. In order to restrict the excitatory input to positive values, we replace any negative value by 0.0.

The voltage independent input is obtained by calculating the weighted sum of the outputs of all connected units (apart from units where voltage dependent input applies, see below):

$$ExcVI_i = \varphi \sum_k w_k^+ \; F(A_k(t) \tag{3}$$

Here, w^+ are the positive weights of the connections from other units k to unit i and φ is a scale factor. The output of a unit is a non-linear function of its activation value, using the following function (Grossberg & Grunewald, 1997; Grossberg & Somers, 1991), with parameters *na* and *qa*:

$$F(A_i) = \frac{A_i^{na}}{(qa)^{na} + A_i^{na}} \tag{4}$$

Crucially, the responsiveness of sensory coding units is modulated by connected feature coding units. This is realized by making the inputs from feature units to a sensory coding unit dependent on the sensory coding unit's activation, which is primarily determined by its external stimulation. This way, a sensory coding unit cannot become highly active by mere top down input. This voltage dependent input from feature coding units to sensory coding units is computed using the following equation (see Tononi et al., 1992, for a similar computation):

$$ExcVD_i = \sum_k w_k^+ \; F(A_k(t)) \times \frac{\max(A_i(t) \times (1 - d_a) - VT, 0)}{1 - VT} \tag{5}$$

Here, d_a is the activation decay rate and *VT* is the voltage threshold. When the sensory coding unit has a (scaled) activation level higher than this threshold, top down input from connected feature coding units is taken into account, rescaled in proportion to the voltage threshold and added to the sensory coding unit's excitatory input. If the sensory coding unit's scaled activation level is lower than the voltage threshold, this input is discarded.

Activation of units is competitive, so that coding units within the same layer (sensory layers, feature layers, task layer, or motor layer) inhibit each other. This is computationally realized by the involvement of 'paired units'. As shown in Figure 6, each of the inhibitory units receive activation from its excitatory paired unit, and propagates inhibition (i.e., their 'outgoing' connections are negatively weighted) to all other excitatory units within the same layer. Such inhibition is characterized by non-linearity, i.e. inhibitory units propagate inhibition when they approach a level of activation. This mechanism ensures that within a layer only one unit becomes highly active after a certain number of simulation cycles.

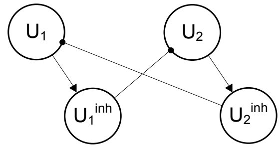


Figure 6. Inhibition between units within the same layer. In each layer, codes are implemented as excitatory units with additional paired inhibitory units. These inhibitory units receive activation from their excitatory paired unit (arrowed connections) and send inhibition (i.e., activation through negatively weighted connections; denoted with solid discs) to all other excitatory units within the same layer.

Inh_i is computed using the following equation:

$$Inh_i = \sum_k w_k^- \; F(A_k(t)) \tag{6}$$

Here, k denotes the inhibitory units belonging to any other unit than unit i in the layer, and w^- are the negative connection weights. The activation of inhibitory units is updated in a similar fashion as the excitatory units, but their input can only be excitatory from the associated paired unit.

Connections

Weights between sensory coding units and feature coding units are set by hand as are the weights of the connections between feature coding units and task coding units closely following the task instruction. The weights from feature coding units to motor coding units are modified using Hebbian learning. Specifically, at the end of each learning trial (see below), the connection weights from feature coding units to motor coding units are updated during a number of cycles according to the following set of equations:

$$\begin{aligned} w_{jk}(t+1) &= (1-d_w)\times w_{jk}(t) + LR\times Act_j(t)\times Act_k(t)\times\left(1-w_{jk}(t)\right) \\ Act_j(t) &= \frac{A_j(t)-LT}{1-LT} \quad if\ A_j(t) > LT \\ Act_j(t) &= 0 \quad if\ A_j(t) \le LT \\ Act_k(t) &= \frac{A_k(t)-LT}{1-LT} \quad if\ A_k(t) > LT \\ Act_k(t) &= 0 \quad if\ A_k(t) \le LT \end{aligned} \tag{7}$$

In these equations, w_{jk} is the weight from feature coding unit j to motor coding unit k, the d_w weight decay rate ensures that only repeated co-activations result in stable weight learning, LR denotes the learning rate (i.e., the magnitude of the change in weights for each learning trial), $Act_j(t)$ is a value based on the activation of feature coding unit j, $Act_k(t)$ is a value based on the activation of motor coding unit k, LT is the learning threshold (above which the activation levels of both units must be in order to engage in weight learning) and $A_j(t)$ and $A_k(t)$ are the actual activation levels at time t of feature coding unit j and motor coding unit k respectively. Note that we rescale the activation of both units to their respective proportion to the learning threshold and that the computed connection weights are bound to vary between 0.0 and 1.0.

The total number of codes (coding units) and connections varies with the specific instances of HiTEC used for the different simulations. All parameters and default values as used in the simulations are listed in the Appendix. In sum, these modeling equations and parameters allow for a biologically plausible simulation of activation propagation through a network of units. Higher decay rates make units decay faster; lower decay rates keep units very active for a longer period of time. Higher input values for external input and stronger weights between units result in faster activation propagation. Higher voltage thresholds make unit activation to a lesser extent enhanced by top down input; conversely, lower voltage thresholds lead to earlier and stronger influence of top down modulation on unit activation. Stronger weights between excitatory and inhibitory units strengthen the lateral inhibition mechanism. As a result, they reduce the time required to settle the competition between the units within a shared layer, after which only one unit remains strongly activated. Lower weights, conversely, lengthen this time to convergence.

Note that our ambition for HiTEC has not been to search for specific parameter values (e.g., thresholds, weight ranges and scaling parameters) in order to optimally fit specific data distributions. We rather set out to provide a proof of principle as to how neurally plausible representations and connectivity may realize stimulus-response translation while addressing critical theoretical issues such as action control (Chapter 3), automaticity (Chapter 4) and coping with task context (Chapter 6).

Simulating behavioral studies

To model a behavioral study in HiTEC, a specific instance of the HiTEC model is constructed with layers, codes (coding units) and connections that match the stimulus, response, and task characteristics of the simulated experiment. Crucially, connections between feature codes and task codes are set to reflect the exact task instructions.

In each simulation there are two phases: first, action effects are learned, reflecting the period in which the participants get acquainted with the keypresses and their effects, which is commonly part of behavioral experiments. In this learning phase, we allow the model a set number of learning trials to acquire the associations between feature codes and motor codes. Note that when a motor code is executed, the changes in the environment (i.e., its action effects) are presented by supplying input to the sensory codes. Propagating activation towards feature codes allows the model to learn the feature code – motor code associations.

In the subsequent, experimental, phase the model is presented with various stimuli by supplying input to specific sensory coding units. Gradually, activation spreads across all the involved coding units in the various network layers. The trial is terminated at the selection of a motor response and the reaction time is determined based on the number of cycles between stimulus onset and response selection. This enables comparing simulated reaction times with reaction times of human participants in behavioral experiments.

In each simulation, multiple simulated subjects are generated based on the same HiTEC model instance. Although the layers and codes in the networks of these simulated subjects are identical, the noise in activation propagation of coding units is random, resulting in individual differences in performance, as reflected in both varying reaction times and error trials. To be able to use between-subjects designs, simulated subjects are assigned to different group conditions (and receive, for example, different task instructions or stimuli). Mean reaction times and standard deviations are computed for each simulated subject and each condition.

Model dynamics

As depicted in Figure 7, when a stimulus is presented to the model, activation propagates from sensory codes to feature codes, involving task codes, other feature codes and motor codes simultaneously. In the figure, an example trial (incongruent trial in the Simon task; see Chapter 4 for the specific HiTEC model instance and actual simulation results) is shown. From the first cycle on a high, right stimulus tone is presented by feeding external input to the sensory codes 'Auditory high' and 'Auditory right'. During the subsequent cycles their activation levels rise accordingly. Simultaneously, activation propagates towards feature codes. Until cycle 21, these are predominantly the feature codes (e.g., 'Right' and 'High') connected to the active sensory codes.

Due to prior action-effect learning, feature code 'Right' propagates activation to motor code 'M_2', of which the activation level is rising during cycles 6 to 29. At the same time, activation propagates from the 'High' feature code towards the task codes, resulting in a relatively more strongly activated 'T_1' and less strongly activated 'T_2' from cycle 7 on. 'T_1' further propagates activation towards feature code 'Left'. As a result, this feature code's activation level rises from cycle 7 on and exceeds the activation level of 'Right' at cycle 24. At the same time activation propagates from 'Right' to the associated motor code 'M_1' which exceeds the activation level of 'M_2' at cycle 34 and reaches the response threshold at cycle 41. At that point, also feature codes 'Left' and 'Key' are highly activated.

Note that these feature codes resemble the action effect of the produced response. Also note that when 'M_2' would have been slightly more activated, this code could have reached the response threshold and the corresponding motor action could have been selected rather than 'M_1', constituting an error trial.

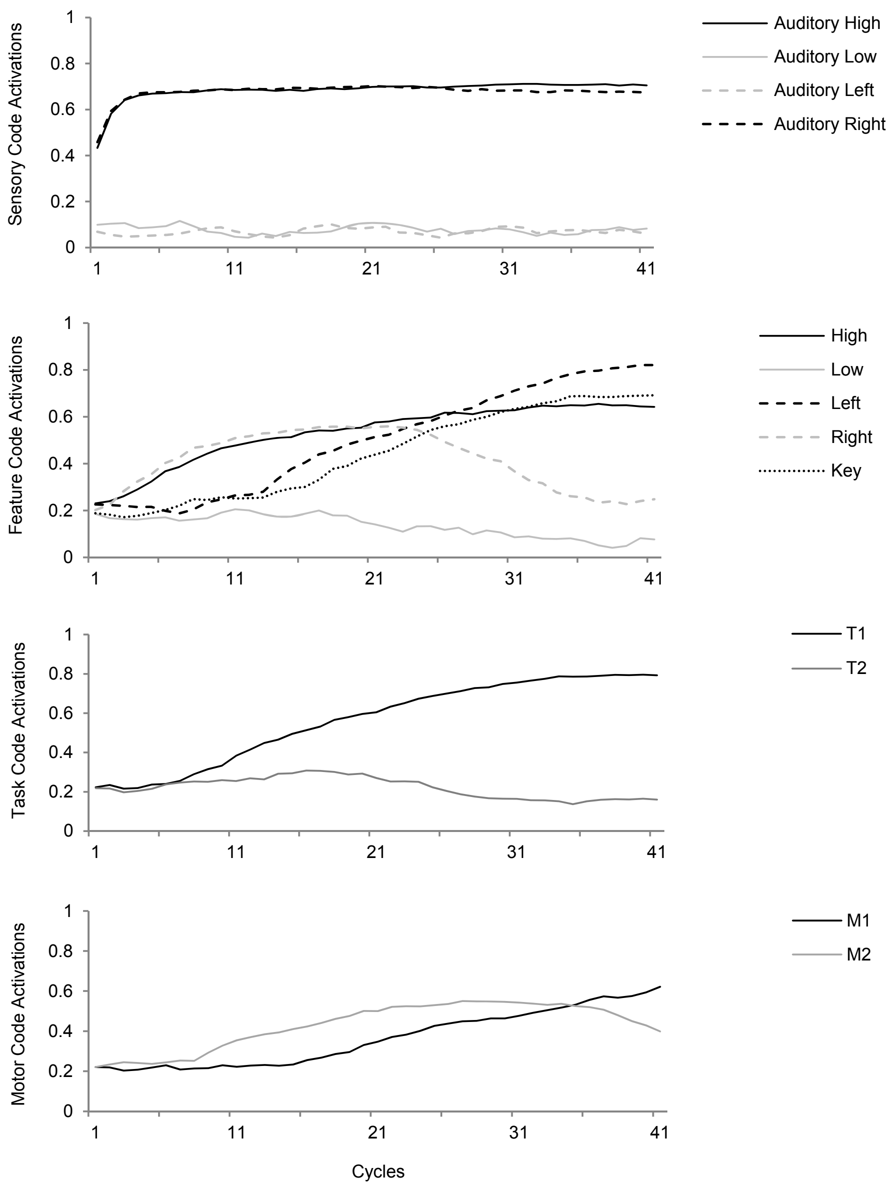


Figure 7. Interactive processing during a single stimulus-response translation trial (i.e., a high right auditory tone) involving representations at all levels simultaneously (example shown from Simon effect simulation. See Chapter 4 for more details).

Discussion

In HiTEC, neuron-like representations realize stimulus-response translation. Stimuli are presented by feeding external input to sensory codes. Responses are considered to execute when a motor code reaches the activation threshold. The connection between perception and action is realized by representations on multiple levels and interconnected by feedforward and feedback connections. The result is an integrated processing network that translates stimuli in responses by gradually propagating activation trough units in the model. Rather than a sequential stepwise process from sensory codes through intermediate representations to response codes, all representations at all levels cooperate and compete and together converge to a response outcome. Crucially, representations at higher levels modulate representations at lower levels. This allows both for direct interaction between perception and action representations and modulation by the task context.

Although the rather simple HiTEC model is not intended as a detailed neuroscientific model, it might be worth noting that its components, as well as their connectivity, do map in a gross way onto specific neural systems. The network architecture follows the same general form as more neurobiologically oriented models of visual attention and object selection do (e.g., Deco & Rolls, 2004).

Our approach is in line with the *integrated competition hypothesis* (Duncan, Humphreys, & Ward, 1997). This hypothesis proposes that visual attention results from competition in multiple brain systems and rests on the three following principles. First, different objects are considered to compete for activation within multiple brain systems. Second, although this competition takes place in multiple brain systems, it is integrated between these systems in such a way that units responding to the same object in different brain systems support each other's activity, whereas units responding to different object compete. Finally, competition is considered to be directed on the basis of relevant object properties based on the current task demands. Duncan et al. (1997) suggest top-down neural priming as a possible control mechanism. HiTEC could be considered both a generalization and specification of this hypothesis. Due to the common coding nature of feature codes, not only visual attention but also action anticipation (and thus action control) are considered to compete for activation, hence generalizing the scope of the integrated competition account. HiTEC further specifies a possible method of directing this competition using task set connections rather than priming. HiTEC explicitly addresses how the task instruction could implement such a task set and how task instruction could influence both perception and action planning.

The notion of interactive processing with mutual influences among multiple subsystems is shared by other models. For example, Ward (1999) proposes in his Selective Action Model that action plans may bias selective perceptual processing towards relevant objects. In his model, representations of a single object and its implications for actions are selected due to gradual and coordinated processing in multiple systems of perception and action. Similarly to HiTEC, the model aims at formulating an alternative to the sequential models of perception and action. To this end, the model follows the integrated competition hypothesis of visual attention and further integrates action systems. In similar fashion as HiTEC, selected representations receive external input and activation gradually spreads among various units coding through the reciprocal connections converging to a selected object and action. Task context is encoded by priming the units that represent the object feature (e.g., the color red) to look for or the action feature (e.g., a grabbing action) to execute. This biases the global competition resulting in response time differences between different conditions. The most important difference between Ward's (1999) model and HiTEC concerns the model architecture: Ward has explicitly taken the ventral 'what' and dorsal 'where' pathways (Milner & Goodale, 1995) into account resulting in two hardwired pathways between perceptual and action systems. HiTEC, in contrast, is based on TEC and thus contains a common coding level of feature representations that are used both for perception and action planning. Another major difference between the models is how a task is internalized. In Ward's model, a selection of codes receives a priming bias input. In this sense, stimulus presentation and task instruction occur simultaneously and using the same mechanism of applying external input. In HiTEC, in contrast, task context is internalized by interconnecting feature codes and generic task codes. These pathways subsequently modulate the propagation of activation resulting from stimulus presentation. Crucially, this allows HiTEC to internalize multiple task rules that compete during subsequent stimulus-response translation, whereas the Ward model seems to be confined to executing one specific task rule depending on the code(s) that receive additional input bias. Finally, although Ward, in accord with our approach, aimed at addressing the interaction between perception and action, his model assumes the implications for action of a given object by fixed connections between object features (e.g., vertical object orientation) and specific actions (e.g., vertical grasp). This connection between perception and action planning is addressed more explicitly in HiTEC using the notion of common codes and ideomotor learning (see Chapter 3). Moreover, these mechanisms allow addressing the issue of automaticity (see Chapter 4), which is not a matter of interest (or readily possible to account for) in the Ward model.

More recently, Botvinick et al. (Botvinick, Buxbaum, Bylsma, & Jax, 2009) further developed the Ward model. In their simulations, they explicitly link specific object features (e.g., color, shape, location) to specific response representations (e.g., reach actions, manual actions, color naming, respectively). In accord with Ward, they find that implementing a task set (i.e., priming specific actions) results in top down input to object features and, hence, in selective attention for objects having these features. Most points of comparison between HiTEC and the Ward model also apply here: the connections between object features and specific actions are assumed, the task set is implemented as additional input to action codes and automaticity is not addressed.

Summarizing, in addressing the interaction between perception and action these models extend (visual) attention for objects with a system that takes action features into account by means of reciprocal connections between perception and action subsystems. How these connections follow from experience or the task context, however, is not explicitly addressed. Moreover these models do not address empirical findings of automaticity (i.e., stimulus-response compatibility) which is key considering their implications for direct interaction between perception and action.

Well-known models of automaticity (e.g., Cohen, Dunbar, & McClelland, 1990; Kornblum et al., 1990; Kornblum, Stevens, Whipple, & Requin, 1999; Zorzi & Umilta, 1995) typically share the general (PDP) connectionist approach with units and excitatory and inhibitory connections. The model of the Simon effect by Zorzi and Umilta (1995), for example, contains stimulus feature codes and response codes. The stimulus feature codes propagate activation towards the response codes. The response codes compete for activation due to their mutual inhibitory connection. In contrast with the models described above, the connections between stimulus codes and response codes are one-directional. That is, stimulus codes activate response codes, not the other way around. In general, these models are more focused on the *process* of translating stimuli to responses and aim at fitting their simulation results to behavioral data. In this endeavor, Kornblum et al. (1999) explicitly divide processing in two distinct sequential stages: stimulus processing and response production. This division ensures that no processing takes place in the response-production stage until activation in the stimulus stage has reached threshold; this is in sharp contrast with HiTEC and the models of the interaction between perception and action discussed above. It does, however, allow them to fit their model to behavioral data on specific stimulus onset asynchrony effects in time courses in SRC effects. In contrast to these dual route process models, HiTEC and both the Ward and Botvinick et al. models take (neurally inspired) representations and reciprocal connectivity into account. The dual route models are discussed more elaborately in Chapter 4 where we discuss the topic of automation.

Finally, it must be stressed that HiTEC has a fairly simple architecture, modeling only a minimal basis of neuroscientific findings. The human brain has many more mechanisms known to mediate perception and action (e.g., subcortical structures such as the superior colliculus and the thalamus). In addition, processing in cortical areas is mediated by a variety of factors (e.g., neurotransmitters) and top down influences and lateral competition, central in HiTEC's interactive processing, are shown to await a first, fast feedforward sweep of activation in visual processing (Lamme & Roelfsema, 2000) suggesting distinct modes of vision and that assuming immediate interaction between multiple levels is rather simplified.

However, despite these simplifications, HiTEC's key assumptions – multiple level representations, common coding level, ideomotor learning, biased competition, reciprocal connections – lead to rather complex and interesting dynamics. We believe that these dynamics may shed light on the interaction and coordination of perception and action planning in human behavior. More specifically, we address how situation-specific meanings of actions emerge in action control (Chapter 3), how and why automaticity occurs (Chapter 4) and how task context may modulate perception and action planning in order to coordinate behavior (Chapter 5).

Chapter 3
Action Control

This chapter is an integration of major parts of the following articles:

Haazebroek, P., & Hommel, B. (2009a). Anticipative control of voluntary action: Towards a computational model. *Lecture Notes in Artificial Intelligence, 5499*, 31-47.

Haazebroek, P., Raffone, A., & Hommel, B. *HiTEC: A Connectionist Model of the Interaction between Perception and Action Planning*. Manuscript submitted for publication.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2011). A computational model of perception and action for cognitive robotics. *Cognitive Processing, 12*, 355-365.

Human behavior is commonly proactive rather than reactive. That is, people do not await particular stimulus events to trigger certain responses but, rather, carry out planned actions to reach particular goals. Planning an action ahead and carrying it out in a goal-directed fashion requires prediction and anticipation: in order to select an action that is suited to reach a particular goal presupposes knowledge about relationships between actions and effects, that is, about which goals can be realized by what action. Under some circumstances this knowledge might be generated ad hoc. For instance, should your behavior ever make a flight attendant to drop you by parachute in a desert, your previously acquired knowledge may be insufficient to select among reasonable action alternatives, so you need to make ad hoc predictions to find out where to turn to. But fortunately, most of the situations we encounter are much more familiar and, thus, much easier to deal with. We often have a rough idea about what actions may be suitable under a given goal and in a particular context, simply because we have experience: we have had and reached the same or similar goals and acted in the same or similar situations before.

How experience with one's own actions generates knowledge that guides the efficient selection of actions, and how humans carry out voluntary actions in general, was the central issue in ideomotor approaches to human action control. Authors like Lotze (1852), Harless (1861), and James (1890) were interested in the general question of how the mere thought of a particular action goal can eventually lead to the execution of movements that reach that goal in the absence of any conscious access to the responsible motor processes (*executive ignorance*). Key to the theoretical conclusion they came up with, the ideomotor principle, was the insight that actions are means to generate perceptions (of wanted outcomes) and that these perceptions can be anticipated. If there would be an associative mechanism that integrates motor processes (m) with representations of the sensory effects they produce (e), reactivating the representation of the effect by voluntarily "thinking of it" may suffice to reactivate the associated motor processes (e→m). In other words, integrating movements and their sensory consequences presumably provides a knowledge base that allows for selecting actions according to their anticipated outcomes—for anticipative action control that is.

This chapter deals with action control in the HiTEC model (see Chapter 2). The HiTEC model allows for associations between motor codes and feature codes in accord with the ideomotor principle described above. Now, three questions arise. The first question relates to how HiTEC allows for acquiring these associations from sensorimotor experience. This is addressed in the first simulation. Second, actions typically lead to multiple perceivable effects or effect features at the same time (e.g., auditory, visual, proprioceptive). and the task context determines which features (or effect feature dimensions) actually matter and how they should be interpreted. Thus, it would be expected that the task set in the model modulates this sensorimotor experience and, hence, influences the situation-specific 'meaning' of actions. Third, how do these meanings (i.e., associations) subsequently influence action control? These latter two questions are explicitly addressed in Simulation 2.

Simulation 1: Action-effect learning

Original experiment

Action-effect acquisition is assumed to occur on-the-fly. Indeed, Elsner and Hommel (2001) showed that people learn action-effect associations spontaneously. In their Experiment 1, participants responded to a visual cue stimulus by pressing a randomly chosen left or right key. One keypress produced a high tone and the other a low tone, which according to the ideomotor principle should have induced bidirectional associations between motor patterns and tone/pitch representations. In the second phase, participants responded to the tones that previously served as action effects by pressing the same two keys, but now according to a specific instruction (e.g., 'when hearing a high tone, press the left key'). In one ('non-reversal') group, the new instruction heeded the learned relationship between tones and keys, so that the tone that was previously produced by a particular keypress was now signaling that keypress. In another ('reversal') group, these relationships were reversed, so that the tone that was previously produced by one keypress was now signaling the other keypress.

It was found that if the tone-key combinations in the second phase matched the key-tone combinations from the first phase, participants were faster than if the combinations did not match up. This suggests that in the first phase, the tones were spontaneously associated with the keypresses that caused them, biasing the keypress responses to the tone stimuli in the second phase. Indeed, later brain imaging studies demonstrated that experiencing action--effect sequences renders the perceived effect an automatic retrieval cue of the corresponding action (Elsner et al., 2002; Melcher, Weidema, Eenshuistra, Hommel, & Gruber, 2008).

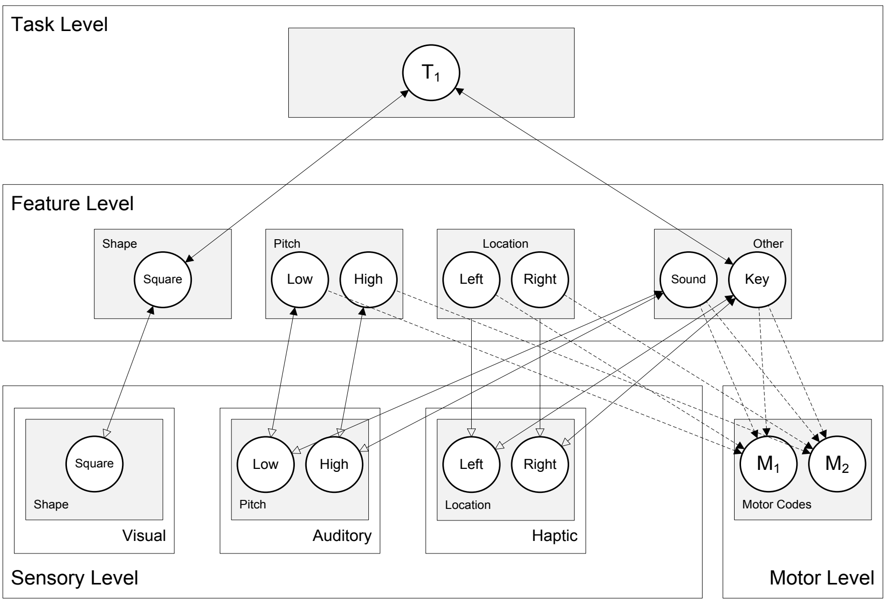


Figure 8. Specific HiTEC Model for Learning Trials of Simulation 1. Feature codes are included to code for the visual cue, the key responses and the auditory action effects. Connections (dashed lines) between feature codes and motor codes are learned. Note that in principle any feature code can be connected to any motor code. However, only some of these possible connections actually become (strongly) weighted as a result of the perceived regularities in action effects. These connections are depicted in the figure.

HiTEC simulation

To simulate Elsner and Hommel's (2001) finding, we created an instance of the HiTEC model with sensory codes for the registration of the visual cue, the auditory pitch levels, and the haptically perceived locations of the keys, with feature codes for the square shape, the pitch levels, the locations, and the 'Sound' and 'Key' in general[5], and with motor codes for the two keypressing actions. This is illustrated in Figure 8. During the learning phase motor patterns 'M_1' and 'M_2' are activated alternately and their respective action effects are presented to the model. As a result, associations are learned between the motor codes and the active feature codes: action-effect binding in the sense of ideomotor theory and TEC (Hommel, 2009).

Figure 9a shows a learning trial in which the motor code 'M1' is activated. This leads to the simultaneous perception of both a keypress and an auditory tone, resulting in a relatively strong activation of some of the feature codes, including 'Left' and 'Low'. This enables the learning of associations between feature codes and motor codes, such as between 'Left' and 'M1' and between 'Low' and 'M1'. The regularity in combinations of motor actions and their perceivable effects results in systematic co-activation of specific motor codes and feature codes. As a consequence, specific motor code - feature code connections are strengthened over time, as is illustrated by Figure 9b. Note that we label the motor codes 'M1' and 'M2' (rather than "Left" and "Right") on purpose, as the motor codes themselves have no intrinsically spatial connotations. They acquire spatial meaning only through the learning of associations between 'M1' and its 'Left' perceptual action effect and between 'M2' and its 'Right' perceptual action effect. In other words, the perceived 'left'-ness and 'right'-ness of motor actions emerges through experience.

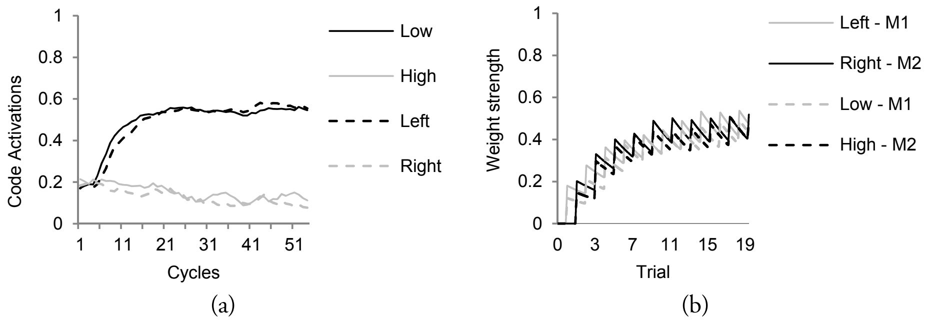


Figure 9. Code Activation and Connection Weight Time Courses in Learning Trials of Simulation 1. (a). During learning trials, motor codes (i.e., 'M1') are activated resulting in increased activation levels of sensory codes and feature codes (i.e., 'Low' and 'Left'). Note that with 'activation level' we refer to the code's excitatory unit's activation level. (b) Connections between motor codes and feature codes (e.g., between 'M1' and 'Low' and 'M2'and 'Left') are gradually strengthened due to repeated experience of the perceptual regularities during the learning trials. After 20 trials, each motor code is strongly connected to those feature codes that were repeatedly co-activated with the respective motor code (i.e., its perceptual effect). Note that feature codes 'Key' and 'Sound' are omitted from the figures for the sake of clarity.

[5] For the sake of simplicity, these feature codes are taken to represent all object characteristics that are not represented by other, specific feature codes, such as its color or location.

In the second phase, we have let the model respond to auditory stimuli with high or low pitch. Note that the change of task (i.e., 'press a random key' vs. 'respond selectively to auditory tones') is reflected in the change in connections between feature codes and task codes only as illustrated in Figure 10. The remainder of the model is kept unchanged, most notably the learned associations between feature codes and motor codes. For the second phase, two different groups of simulated subjects were to respond to stimuli according to different instructions. The 'non-reversal' group was to respond to the learning-compatible stimuli (i.e., what had been the effect on an action now became the stimulus signaling this action), whereas the 'reversal' group was to respond to auditory tones with responses previously produced the alternative tone. In the model, stimulus tones are presented by stimulating auditory sensory codes. Activation flows from these sensory codes towards 'Pitch' feature codes, task codes and to the 'Location' feature codes and the 'Key' feature code. Also, activation flows through the learned associations towards the motor codes. On average, in the non-reversal condition the model reaches the response threshold faster than in the reversal condition.

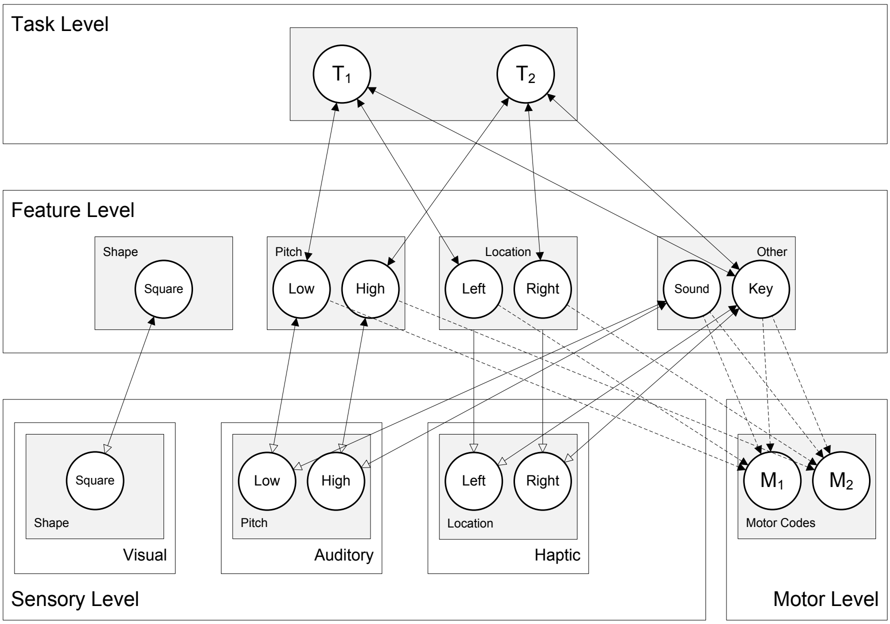


Figure 10. Specific HiTEC Model for Experimental Trials of Simulation 1. Note that only the feature code – task code connections have been changed as compared to Figure 3, reflecting a new task instruction with the same simulated subject. Now, two task codes are present representing two alternative response choices to the task stimuli. Crucially, the learned connection weights (dashed lines) between feature codes and motor codes are kept unchanged. Connections between location feature codes and motor codes allow the model to choose the appropriate motor action. Connections between pitch feature codes and motor codes result in a compatibility effect with respect to the stimulus pitch and the chosen response. Note that in principle any feature code can be connected to any motor code. However, only some of these possible connections actually become strongly weighted as a result of the perceived regularities. These connections are depicted in the figure.

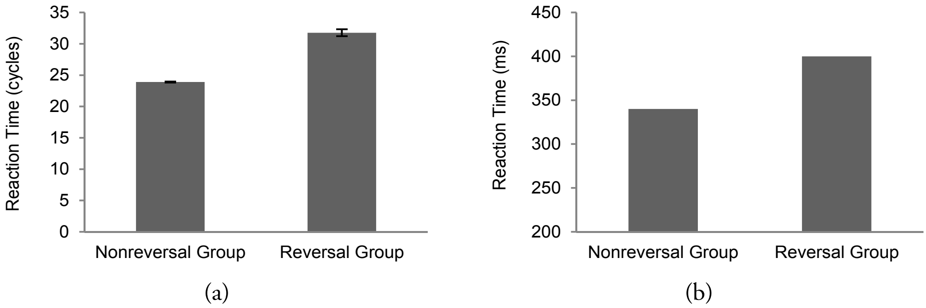


Figure 11. Results of Simulation 1 compared with behavioral data (adopted from Elsner & Hommel, 2001), showing average reaction time means and standard deviations. Human variance data was not available.

Simulation results

The simulation results are provided in Figure 11, where 11a shows the mean reaction times in cycles for both groups and 11b the mean reaction times in milliseconds as found in the original behavioral study. In the simulation, the non-reversal group (15 simulated subjects, 20 experimental trials each) responded on average in 23.90 cycles (SD = 0.32) on average, while the reversal group (15 simulated subjects, 20 experimental trials each) needed 31.77 cycles (SD = 2.10) on average. No errors were made and no simulated subjects were excluded from analysis. Overall, our simulation shows a good fit with the available empirical data and provides insight in the internal dynamics of action-effect learning. It demonstrates that the model automatically learns novel action-effect associations, a fundamental aspect of ideomotor learning. And it demonstrates how the acquisition of action-effect associations creates a basis for stimulus-response compatibility effects: Any stimulus that shares features with a previously learned action effect will tend to activate the associated action. Stimulus-response compatibility is addressed more explicitly in Chapter 4.

Simulation 2: Action planning

The observations of Elsner and Hommel (2001) confirm the claim from ideomotor theory that action-effect associations are automatically acquired as demonstrated in Simulation 1. However, this does not yet speak to the further-reaching claim of ideomotor theory that action effects play an important role in the *planning* of intentional actions. Evidence supporting that claim was provided by Kunde, Koch and Hoffmann (2004), who showed that choice performance is affected by the compatibility between haptic action effects of the responses proper and novel (auditory) action effects.

Original experiment

In their experiment, for one group of participants, responses were followed by a compatible action effect; the loudness of the tone matched the response force (e.g., a loud tone appeared after a forceful key press). In the other group of participants the relationship between actions and action effects was incompatible (e.g., a soft tone appeared after a forceful key press). In the following test phase subjects had to respond to a visual cue stimulus by pressing the key softly or forcefully. It was found that the group with action-compatible action effects was faster on average than the group with incompatible action effects. Given that the tones did not appear before the responses were executed, this observation suggests that the novel, just acquired action effects were anticipated and considered in the response-selection process.

HiTEC simulation

This effect of response-effect compatibility was simulated in HiTEC. According to ideomotor theory, actions are planned in terms of their perceptual features (i.e., the features of their perceptual consequences), so that the critical compatibility relationship exists between the natural haptic and/or kinesthetic action effects of the keypress response and the auditory action effects of the tone presentation. Accordingly, the model, as shown in Figure 12, contains sensory codes for the visual colors, auditory intensities and haptic intensities. Motor codes 'M_1' and 'M_2' represent forceful and soft keypresses, respectively. Again, to the model these motor codes are not intrinsically forceful or soft but are associated with (and acquire their meaning from) these perceptual characteristics through learning.

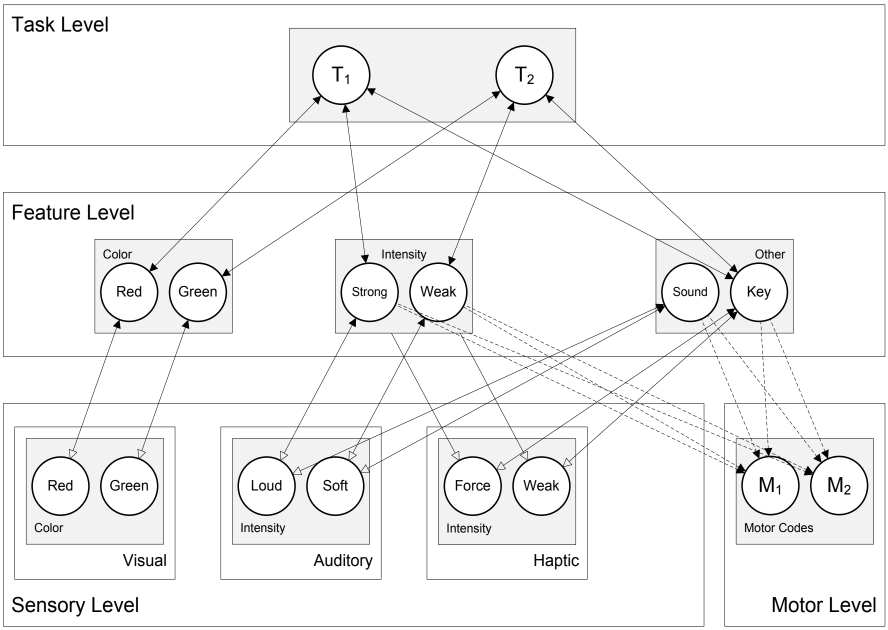


Figure 12. Specific HiTEC Model for Simulation 2. The model contains feature codes that code for the stimulus color and the intensities of both the key press itself and the additional auditory perceptual effects. Note that the task instruction is already internalized before presenting the learning trials as reflected in the connections between feature codes and task codes. This biases the learning of connections between feature codes and motor codes. Note that in principle any feature code can be connected to any motor code. However, only some of them actually become (strongly) weighted reflecting the specific perceptual regularities. In this experiment, activating motor codes can have inconsistent action effects resulting in activating both 'Strong' and 'Weak' feature codes (albeit not both to the same extent). Therefore, all possible connections between motor codes and intensity feature codes are depicted in the figure.

Moreover, the learning of these associations is influenced by the consistency of the simultaneously presented action effects; if a consistent action effect is presented (e.g., a forceful keypress with a loud tone), then both action effects result in the activation of the same intensity feature code (e.g., 'Strong'). As a consequence, the association with the active motor code becomes strong. Conversely, when inconsistent action effects are presented (e.g., a forceful keypress together with a soft tone), both 'Strong' and 'Weak' are activated and associations with the motor codes are only learned weakly. Indeed, because both feature codes are activated and inhibit each other, they are less active than when only one of the feature codes is activated. Crucially, already during learning the 'Key' feature code is connected to the task codes (whereas the 'Sound' feature code is not, as it is not part of the task instruction). When the 'Key' feature code is activated due to the perception of a haptic action effect, it sends activation to both task codes 'T_1' and 'T_2' that both again further enhance the task-relevant 'Key' code. Thus, this mere connectivity makes the system enhance the perception of haptic intensity—which is relevant for action control—over auditory intensity—which is not (in accordance with empirical findings demonstrating this kind of impact of action-control requirements on perception and attention: Hommel, 2010). As a result the haptic intensity becomes the major determinant in the weight learning of connections between 'Intensity' feature codes and the motor codes, whereas the auditory intensity moderates this process.

Simulation results

After running the simulation (two groups of 15 simulated subjects performing 20 experimental trials each), 6 subjects were excluded[6] (all in the incompatible group) from analysis because of error percentages higher than 30. After removal of error trials, the results revealed that the compatible group responded faster (M = 24.16 cycles, SD = 0.20) than the incompatible group (M = 30.64 cycles, SD = 4.10). Figure 13 shows a good fit of the outcome with the empirical data obtained by Kunde, Koch and Hoffmann (2004). This confirms that HiTEC is able to produce response-effect compatibility effects without introducing new kinds of representations or processing principles.

Discussion

The simulations in this chapter deal with action control. Action control in HiTEC is based on the ideomotor principle which stresses both the acquisition of action-effect associations and the use of these associations in action planning. Simulation 1 addresses how novel action-contingent perceivable effects are (spontaneously) associated to the actions that yield these

[6] In this and the following simulations a substantial number of simulated subjects were to be excluded because of excessive error rates. It is interesting to note that this is a very common observation in investigations of compatibility effects in children and infants, where the reaction time effects are often not too different from adults while the error effects are dramatic (e.g., Eenshuistra, Weidema, & Hommel, 2004; Kray, Eenshuistra, Kerstner, Weidema, & Hommel, 2006). In fact, some children are virtually unable to follow the instruction in the face of stimuli suggesting an alternative response. Such observations are commonly attributed to the still developing frontal cortex, which plays an important role in maintaining instructions and action goals (Miller & Cohen, 2001).

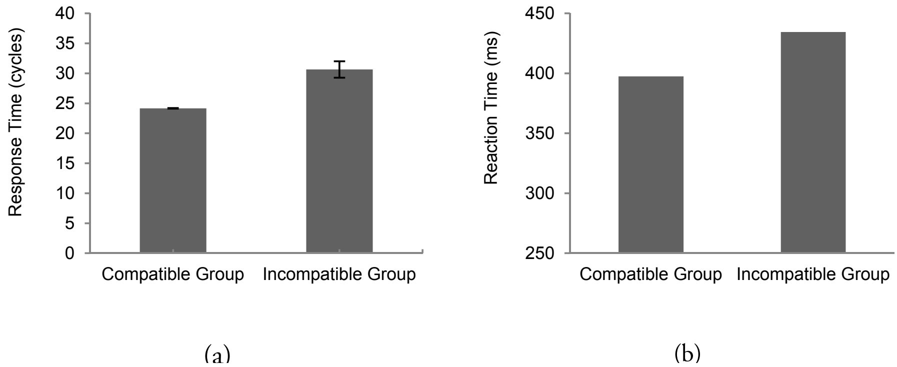


Figure 13. Results of Simulation 2 compared with behavioral data (adopted from Kunde, Koch, & Hoffmann, 2004), showing average reaction time means and standard deviations. Human variance data was not available.

effects. Simulation 2 demonstrates how the (internal) consistency of these effects influences the representations of these effects. As action-effect learning depends on the activation of both motor codes and feature codes, the consistency of feature code activation has consequences for the resulting association strengths. And because these associations have a crucial role in planning actions in response to stimuli, subsequent stimulus-response translation is influenced by the strengths of these associations. As a result, action planning takes the contextual meaning (e.g., consistency among action effect features) of motor actions into account as represented in the acquired action-effect associations.

Importantly, Simulation 1 demonstrates that in HiTEC novel action effects become (at the distal level) associated to motor codes. This is schematically illustrated in Figure 14a. Here the coupling of sensory and motor code to a shared feature code is depicted. The connection between sensory and feature code is the result of (earlier) grounding processes; the connection between motor code and feature code results from action-effect learning as demonstrated in both simulations. That is, execution of the motor code M results in changes in the environment E that are 'picked up' by sensory code S and distally represented by feature code F. Multiple encounters of this sensorimotor co-occurrence is considered to strengthen the connection between motor code M and feature code F.

Figure 14b illustrates the fact that different sensory codes may activate the same feature code. This is the case in Simulation 2 where both auditory and haptic sensory codes project to the same distal feature codes that code for 'intensity'. Consistent action effects activate these feature codes more strongly and therefore lead to stronger action-effect associations as compared to inconsistent action effects, as suggested by the findings of Kunde and colleages and as demonstrated in Simulation 2. Crucially, in the HiTEC simulation of this experiment, the auditory and haptic sensory codes receive equal external stimulation upon perceiving the action effect. It is due to the top down modulation of the task-feature connections with the 'key' code that the haptic sensations are enhanced and thus play a dominant role in representing the action effect as compared to the auditory sensations. This task set modulation of action effect representation (and thus its influence on action-effect learning

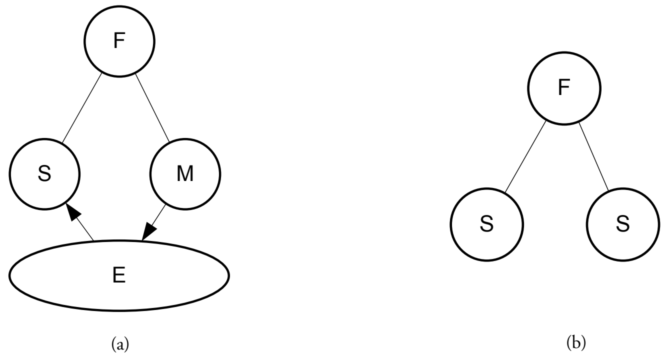


Figure 14. Schematic illustrations of couplings between sensory code S, feature F and motor codes M in HiTEC relevant for the simulations in this chapter. (a) depicts the coupling of sensory and motor code to a shared feature code as a result of sensorimotor co-occurrences in environment E. (b) illustrates the fact that different sensory codes may activate the same feature code.

and subsequent stimulus-response translation) by means of mere connectivity rather than additional bias inputs is a key mechanism resulting from the multiple-level characteristic of the HiTEC architecture and we will turn to this more explicitly in Chapter 5 where the task set is manipulated experimentally.

Summarizing, executing motor actions results in changes in the environment. These changes are perceived and represented using sensory codes and feature codes. Sensorimotor co-occurrences lead to co-activation of motor codes and feature codes representing their perceptual effects. This co-activation is considered to strengthen the action-effect (m-f) associations. The representation of action effects is modulated by the same influences as 'normal' stimulus perception, such as consistency among different sensory modalities and top down modulation (cf. priming) of task context resulting in *situation-specific meaning of actions*. The action-effect associations can further be used as a means to *control actions based on their perceptual effects*. That is, by means of anticipating the effect of an action, the appropriate motor action can be planned (cf. Schütz-Bosbach & Prinz, 2007). In other words, the action is now considered to be under 'intentional control'.

Note that we use a very limited notion of 'learning' here. Adult participants are unlikely to learn how to press a key in a lab experiment but rather bring that knowledge to the lab. And yet, what we can achieve with an action and what sensory effects an action produces can change very rapidly, so that even already acquired action-effect associations need to be updated frequently. In this thesis we mainly focus on this kind of short-term learning, which allows old actions to be adapted to the current situation and new action effects to become associated with the motor patterns they were produced by.

HiTEC's differentiation between motor codes and codes representing action effects is in line with Wallace's (Wallace, 1971) conclusions. In his experiment, in one condition, participant's left and right hands were placed on left and right keys, respectively. In the other condition, the hands were crossed. That is, the left hand was on the right key and the right hand on the left. His results showed that compatibility effects were related to the position of the response keys and not to the particular hand used to produce the response. This lead

Wallace to the conclusion that there must be an abstract 'response code' that relates to the stimulus code (in terms of compatibility) and that this response code is associated to the given output. The degree of compatibility between stimulus and response code, he argues, then somehow must affect the processing of this output, through this response code. In his conclusions he speculates that this response code might be considered a 'body code' and for his specific experiment he considers the 'hand itself' as this body code. Thus compatibility between stimulus and hand location (i.e., the position of the hand above the keys; in other words: the key locations) influences the associated output process, which is the actual keypresses involving the muscles of that hand and limb.

In the decades that followed more evidence of such intermediate 'response codes' that can match stimuli to varying degrees has been accumulated. For instance, Riggio, Gawryszewski and Umilta (1986) reported that when participants responded with sticks that were either parallel or crossed, the Simon effect was found to relate to the stick end position, not to the hands holding the sticks. In experiments where participants operated the left and right keys with fingers of the same hand holding their hand in palm up or palm down position. As it turned out, compatibility effects were independent of the chosen hand or finger. In a study by Guiard (1983) participants had to respond with a steering wheel. These results suggest that not the position of the hands but the steering direction (as in a car) determines the Simon effect, indicating that the notion of 'left' or 'right' responses drives on prior experience. In a study by Beckers and colleagues (Beckers, De Houwer, & Eelen, 2002) novel, affective action effects (i.e., electroshocks) were presented following a motor action. In subsequent experimental trials, these novel action effects (i.e., distally represented in terms of positive vs. negative) were shown to yield SRC effects with valenced stimuli, suggesting that these novel action effects were taken into account during action planning (Haazebroek, van Dantzig, & Hommel, 2009; 2010). Finally, Hommel (1993) experimentally manipulated the task instruction and demonstrated significant differences in SRC direction, suggesting that the coding of responses directly depended on this instruction (see Chapter 5). Summarizing, the coding of responses seems to depend on direct context, recent and prior experience, and the task instruction. In the case of the Simon task, it is the rather flexible notion of 'left'-ness or 'right'-ness, rather than the actual physical location of a response, that seems to interact with the spatial location of the stimulus. This constraint of flexible response codes has been taken seriously in the HiTEC connectionist model.

Despite this flexibility in human action planning as demonstrated in various empirical findings, most models of perception and action (e.g., Botvinick et al., 2009) as well as models of stimulus-response compatibility (e.g., Zorzi & Umilta, 1995; Kornblum et al, 1999), however, do not differentiate between response codes and motor codes. The Ward model (Ward, 1999), as a notable exception, does contain codes for both 'grasping horizontally' and 'grasping'. This somewhat resembles the notion of 'intermediate' response codes that mediate between stimulus features and action codes proper and refer to some perceptual property of the response. Still, it is unclear how associations between these intermediate codes and the action codes emerge. Botvinick et al. (2009), however, eliminated these intermediate codes in their adaptation of the Ward model, and directly connected stimulus feature codes (e.g., location, shape, color) to specific response codes (e.g., reaching, manual actions, color naming, respectively). In similar vein, the spatial SRC models (e.g., Zorzi & Umilta, 1995; Kornblum et al, 1999) generally contain direct connections between stimulus feature codes and response codes. The latter codes are assumed to relate to left or right responses. How these (spatial) response codes have bearing on actual motor actions is not explicitly addressed. If, however, these codes should be interpreted as motor codes that do directly relate to patterns of muscle contractions, then the question arises how these codes could be spatially connotated in the first place, as assumed in these models; and, how they can be directly related to specific (spatial) stimulus feature codes. In addition, in such a scheme, it would not be possible to cope with the flexibility of (spatial) coding of responses as clearly demonstrated in a variety of empirical studies.

Summarizing, in most existing models containing action systems action codes refer to both the motor programs that are executed and to the response codes that can be directly related to stimulus features. This, however, does not allow such a model to account for the shown flexibility in response coding and, thus, in action control. More specifically, it is unclear how these action codes acquire their meaning and how this could depend on the current (task) context as demonstrated in the simulations in this chapter.

It could be argued that action control as modeled in HiTEC is still rather limited. Indeed, a motor code does not seem to represent a complete specification of a motor action (e.g., trajectory, speed, acceleration, and deceleration). Empirical findings by Prablanc and Pélisson (1992), however, may suggest the human brain does not do this either. In their experiment, participants were instructed to move their hands to a goal position indicated by a light. This light was sometimes shifted by a few centimeters after they had begun their hand movement. Although participants were not able to notice the shift (which was carried during eye movements), they moved their hand straight to the new goal location without abrupt changes in the movement trajectory. This suggests that, once an action has been linked to an object location, any change in this location leads to an automatic update of the movement's parameters, even if the change occurs outside the actor's awareness. Hence, it makes sense to interpret motor codes as blueprints of motor actions that need to be filled in with this specific, on line, information when executing the movement, much like the schemas put forward by Schmidt (1975) and Glover (2004). In HiTEC, action effect anticipation acts as a rich retrieval cue for associated motor programs. At the same time, forming this anticipation (i.e., activating the distal feature codes belonging to the action effect) may reflect a (distal) specification of an action plan that can be used to fill hook up online movement parameters during action execution. In Simulation 2, one of the action anticipation consists of a highly active 'strong' and 'key' feature codes (resulting in an active motor code). These feature codes could be considered (distal) representations of the action plan that allows for controlling the action in similar way as in the experiment by Prablanc and Pélisson (1992). In addition, one could imagine that by activating distal features, the related proximal sensory codes are top down moderated to 'focus their attention' towards specific aspects of the environment (e.g., visual object location). This distinction between offline plans and online close-loop control is in line with the notion of different pathways in the brain (i.e., ventral 'what' vs. dorsal 'where' pathways, Milner & Goodale, 1995). Here, the ventral pathway relates to offline selection and the dorsal pathway to online action control. These ventral and dorsal pathways are included in the Ward model (Ward, 1999) although it is unclear to what extent these labeled routes actually constitute offline vs. online action control in this model.

In HiTEC, action-effect learning may be seen as a rather reactive/passive process explicitly triggered by a learning phase. The principle itself, however, is consistent with the notion of a more active actor/perceiver as brought forward by O'Regan and Noë (2001). In their sensorimotor theory, perceiving is a way of acting, actively exploring the environment rather than merely registering and representing the outside world. Vision, for example, requires eye movements that directly influence retinal stimulation. Hence, seeing would need to account for both the actual changes in the environment and those due to oculomotor actions. The active perceiver/actor, they argue, would need to learn how his/her own actions influence perceptions (sensorimotor contingencies). Perceiving the world thus builds on this knowledge. Although HiTEC does not model learning such contingencies per se, it does share the idea of acquiring grounded representations of sensorimotor regularities in interactions with the world (as illustrated in Figure 14a) and using those representations both for perception (as suggested in the sensorimotor theory) and actions, which indeed lead to perception, both in the sensorimotor account and in HiTEC.

One possibility to endow HiTEC with a more active learning strategy is by means of action monitoring. The anticipated action effects are a trigger for action selection, but also form an expectation of the perceptual outcome of the action. Differences between this expectation and reality lead to adjusting the action on a lower sensorimotor level than is currently modeled in HiTEC. What matters now, is that the feature codes are interacting with the sensory codes, making sure that the generated perception is within the set parameters, as determined by the expected action outcome. If this is not (well enough) the case, the action should be adjusted. However, when a discrepancy of this expectation drastically exceeds 'adjustment thresholds', it may actually trigger action effect learning (phase 1). Apparently, the action-effect associations were unable to deliver an apt expectation of the actual outcome. Thus, anticipating the desired outcome falsely led to the execution of this action. This may trigger the system to modify these associations, so that the motor codes become associated with the correct action effect features. Such a monitoring system could work along the lines of (Botvinick, Braver, Barch, Carter, & Cohen, 2001). The experience of response conflict and/or of negative feedback might strengthen the activation state of goal codes and their impact on stimulus-response processing, which would tend to prevent errors in the future (van Steenbergen, Band, & Hommel, 2009).

Furthermore, anticipation based action control (see also Butz & Pezzulo, 2008) is consistent with basic concepts in research on human motor control (e.g., Wolpert & Ghahramani, 2000) Here, the motor system is considered to form a loop in which motor commands lead to muscle contractions which cause sensory feedback, which in turn influences future motor commands. Neural circuits are considered to form internal models that control motor action: forward models are considered to model the causal relationship between actions and their consequences and inverse models determine the motor command required to achieve the desired outcome. These concepts clearly resonate with the role of action-effect associations in the HiTEC model.

Finally, it should be noted that in the current simulations we have focused on paradigms where the participant (and, hence, the model) produces an action in response to a stimulus. Other paradigms that include non-stimulus-driven action planning or sequences of actions may stress the model's ability to control its actions by anticipating action effects (i.e., goal-directed actions; see also Hommel, 2009).

To conclude, the meaning of action seems to be rather flexible as demonstrated by various findings on action control. In HiTEC, this flexibility is addressed explicitly by action-effect learning which involves motor codes, their effects in the environment and registration of these effects by HiTEC's sensory codes and feature codes. This allows for acquiring associations between motor codes and feature codes, effectively representing action effects, in line with ideomotor theory. Action-effect learning takes the (task) context into account allowing situation-specific meanings of actions to emerge that are subsequently used to control actions in response to stimuli. These aspects of action control are generally not addressed in models of perception and action, or in models of SRC.

Moreover, using anticipations to select and plan actions is in line with various other lines of research on action control. In HiTEC action effect anticipations are encoded using the same (distal feature) representations used for representing stimuli. This sharing of representations between stimuli and responses may sometimes lead to notable consequences for stimulus-response-translation in terms of compatibility effects as will be discussed in the next chapter.

Chapter 4
Automaticity

This chapter is an integration of major parts of the following articles:

Haazebroek, P., Raffone, A., & Hommel, B. *HiTEC: A Connectionist Model of the Interaction between Perception and Action Planning.* Manuscript submitted for publication.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2011a). A computational model of perception and action for cognitive robotics. *Cognitive Processing, 12*, 355-365.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2011b). Interaction between Task Orient-ed and Affective Information Processing in Cognitive Robotics. *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, 59,* 34-41.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2009). Towards a computational account of context mediated affective stimulus-response translation. P*roceedings of the 31st Annual Conference of the Cognitive Science Society (pp. 1012-1017)*. Austin, TX: Cognitive Science Society.

Traditional views on human information processing hold that responding to stimuli in our environment follows a sequence of separable stages of processing (e.g., Donders, 1868; Neisser, 1967; Sternberg, 1969) from stimulus perception, to decision making, up to response execution. Numerous empirical findings, however, have demonstrated that parts of human information processing do not seem to involve conscious cognitive decision making. Features of perceived objects (such as location, orientation, and size) can influence actions *directly* and beyond (tight) cognitive control, as illustrated by stimulus–response compatibility (SRC) phenomena (for general overviews, see Hommel & Prinz, 1997; Prinz & Hommel, 2002; Proctor & Vu, 2006), such as the Simon effect (Simon & Rudell, 1967) as simulated in Simulation 3.

To account for both controlled and automatic processing, various dual route process accounts have been proposed (e.g., Zorzi & Umilta, 1995; Kornblum, et al., 1990; but see Hasbroucq & Guiard, 1991, for a strictly perceptual account). These accounts propose that there is, next to the first cognitively controlled route, a second, direct route from perception to action that can bypass cognition, as explicitly modeled in various computational models of the Simon effect. Essentially, dual route accounts consider the observed direct stimulus-response interaction as an exception requiring an additional route. Moreover, they typically do not address the reason *why* some stimulus features directly influence action and others do not.

In this chapter we attempt to explain how and why automaticity occurs in the HiTEC connectionist model (see Chapter 2). We explicitly address how representational and processing characteristics of HiTEC *inevitably* lead to SRC effects. Here, common codes play a crucial role. Building upon this notion of common codes, HiTECs structure and processes allow stimulus features, both task relevant and task irrelevant, to be registered, processed and translated into responses. In this endeavor we focus on two key paradigms. In Simulation 3, a HiTEC instance is constructed to simulate the Simon task. In Simulation 4, we model the Stroop effect. As HiTEC treats stimulus and response representation in a similar way, it is to be expected that a model instance similar to the one used in Simulation 3 would be able to account for the Stroop effect as well. The empirical findings accounted for in this chapter have been modeled before by other (dedicated) computational models. We conclude this chapter with a comparison of some of these models with our approach.

Simulation 3: Simon effect

Original experiment

Simon and Rudell (1967) showed that people respond faster to stimuli if the location of the stimulus is compatible with (corresponds to) the response location, even when stimulus location is not task relevant. In the standard Simon task, stimuli with a non-spatial stimulus feature (e.g., auditory pitch) are presented at different locations (e.g., left or right). Participants are instructed to respond to the non-spatial feature by giving a spatially defined response (e.g., pressing a left or right key). Even though the location of the stimulus is not relevant

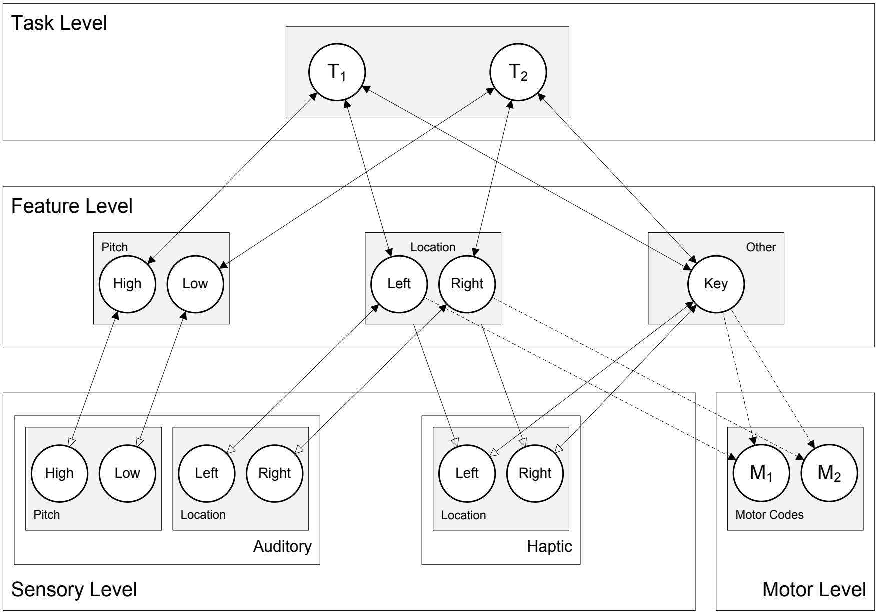


Figure 15. Specific HiTEC Model for Simulation 3. Feature codes are present for stimulus pitch and location. Note that location feature codes are used for encoding both stimulus location and response location. The task instruction is already internalized before presenting the learning trials. This biases the learning of connections between feature codes and motor codes. Note that in principle any feature code can be connected to any motor code. However, only some of them actually become (strongly) weighted reflecting the specific perceptual regularities.

for this task, performance is facilitated when the chosen response corresponds spatially to the stimulus location.

HiTEC simulation

The Simon effect was modeled in HiTEC using sensory codes for auditory pitch[7], auditory locations and haptic locations. At the feature level there are feature codes for pitch, location and for 'Key'. The model, as shown in Figure 15, contains two motor codes, 'M1' and 'M2', representing pressing the left and the right key. During the learning phase, 'M1' and 'M2' are activated alternately and their respective action effects are presented to the model. As a result, associations are learned selectively between the motor codes and the 'Left' and 'Right' feature codes.

In the experimental trials, tones are presented and are responded to by anticipating and executing left or right keypresses (i.e., by activating 'Left' or 'Right' feature codes respectively). Crucially, the 'Left' and 'Right' feature codes are also activated when the tone stimulus is presented on the left or right, yielding a compatibility effect as demonstrated in Figure 16 and as reflected in the results. Because 'Left' and 'Right' are features that are relevant for

[7] We decided to simulate the auditory version of the Simon task, rather than the more common visual version, because that will make it easier for the reader to relate it to the auditory version of the Simon task that we modeled in Chapter 5. However, the logic of our modeling applies to visual versions just as well.

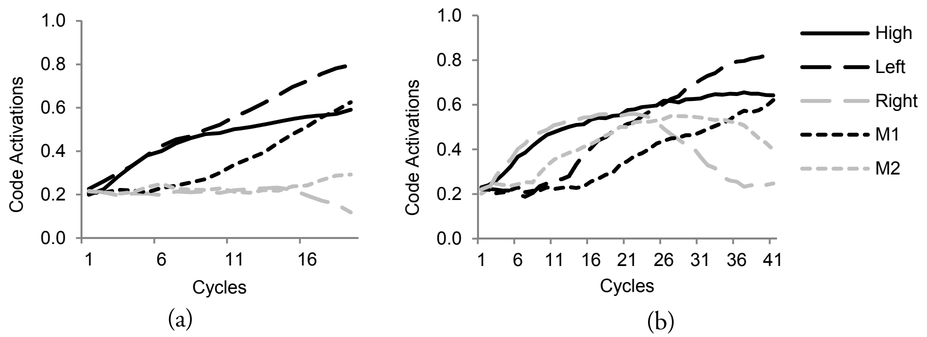


Figure 16. Time courses of feature code and motor code activations in the experimental trials of Simulation 3. Panel A depicts the activations in the compatible condition. Here 'M1' reaches threshold in 19 cycles. Panel B depicts the dynamics in the non-compatible condition. Here 'M1' reaches threshold in 41 cycles. In the latter condition, activating 'Right '(as stimulus feature) biases the model into planning a 'right' action. This, however, is overcome due to the task connections so that 'Left' becomes stronger and eventually wins over 'Right'. Similarly, first the incorrect motor response, 'M2' becomes active, but eventually 'M1' reaches threshold. In effect, the model takes longer to respond in the non-compatible condition than in the compatible condition. Activations of the remaining feature codes, task codes and sensory codes are omitted for sake of clarity.

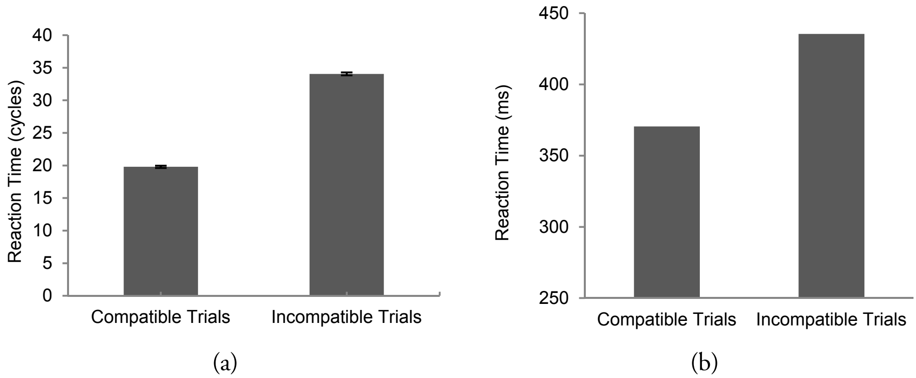


Figure 17. Results of Simulation 3 compared with behavioral data (adopted from Simon & Rudell, 1967), showing average reaction time means and standard deviations. Human variance data was not available.

response coding, they are part of the task connections. As a consequence, stimulus location becomes of influence in the overall stimulus-response translation. As shown in Figure 16, in the compatible condition, the stimulus location already activates the correct spatial feature code and thereby speeds up response selection, on average.

Conversely, in the incompatible condition, stimulus location activates the wrong spatial feature code, which also already activates the wrong motor code. Meanwhile, however, the stimulus pitch is translated — through the task codes — into the correct spatial feature codes and the correct motor code. This latter pathway typically overcomes the head start due to the overlap-pathway, but the code overlap does slow down the overall translation as reflected in the results.

Simulation results

In the simulations (15 simulated subjects, each performing 20 trials in each condition), no errors were made and no subjects were excluded from analysis. Compatible trials yielded faster responses (M = 19.79 cycles, SD = 0.18) than neutral trials (M = 25.28 cycles, SD = 0.23), which again produced faster responses than incompatible trials (M = 34.04 cycles, SD = 0.73). The results are shown in Figure 17, where 17a shows the averaged simulated reaction times in cycles and 18b the empirical data from the study by Simon and Rudell (1967) in milliseconds. Overall, the simulation results fit well with the available behavioral data, demonstrating that and how code sharing between stimulus and response results in compatibility effects. Note that the processing logic according to which SRC effects are produced are identical to that responsible for action-effect compatibility effects as assessed in Simulation 1 (see Chapter 1).

Simulation 4: Stroop effect

As we do not differentiate between perceptual and action stages, one could argue that stimulus–response compatibility and stimulus–stimulus compatibility would need to work similarly in HiTEC.

Original experiment

Stroop (1935) showed that if people are instructed to name the ink color of color words, they are slower if the word (e.g., “blue”) appears in an incompatible ink color (e.g., red). This compatibility effect is dramatically reduced if non-verbal responses are required (MacLeod, 1991), suggesting that the task-irrelevant words interfere (at least partly) with verbally naming the colors. Note that this interpretation of the Stroop effect bears a strong resemblance to the Simon effect as the effect is now attributed to incompatibility between a stimulus feature (ink color) and a response feature (verbal sound).

HiTEC simulation

In HiTEC the Stroop effect is simulated by having the model, as depicted in Figure 18, structured very similarly to the model used in Simulation 3 to simulate the Simon effect. The connections from visual shape to word feature codes have been made slightly stronger (weight of 0.45 instead of 0.4; see Appendix for further details) in order to take into account the richer experience of word reading as compared to color naming. During the learning trials, the model alternately executes ‘M_1’ and ‘M_2’, reflecting the ‘physical’ pronunciation of the respective words. The model is subsequently presented with the auditory feedback (i.e., reflecting the perception of this pronunciation) and associations are learned between motor codes and feature codes. During experimental trials, naming ink color of compatible color words benefits from facilitation whereas naming the color of incompatible color words suffers from interference.

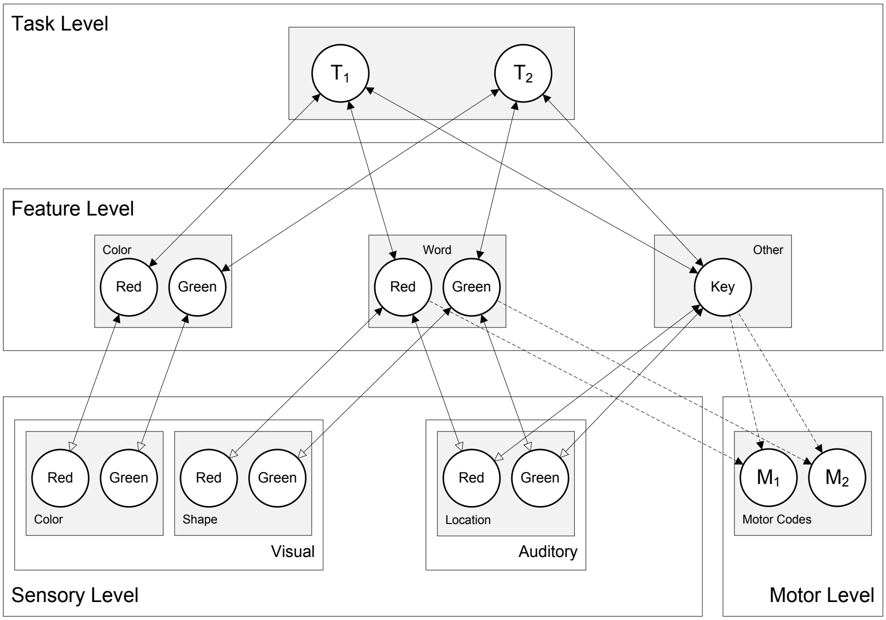


Figure 18. Specific HiTEC Model for Simulation 4. Feature codes are present for stimulus colors and words. Crucially, word feature codes are used for encoding both stimuli (i.e., the color words) and responses (i.e., the words to name the ink color). Note that this structure is in essence identical to the structure of the model used for Simulation 3. Connections between word feature codes and motor codes are learned during learning trials (i.e., pronouncing the words).

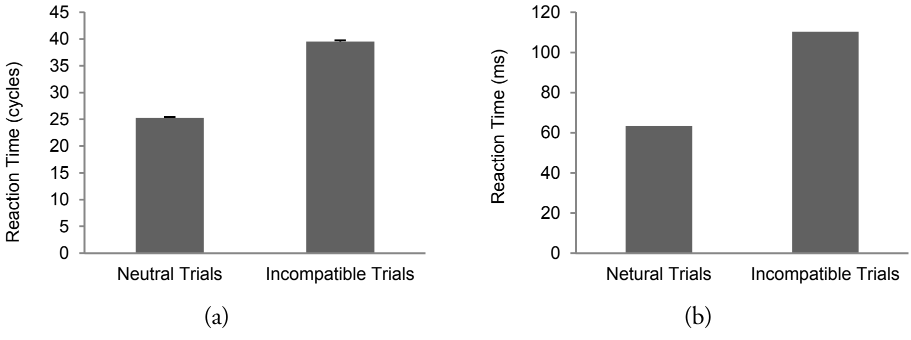


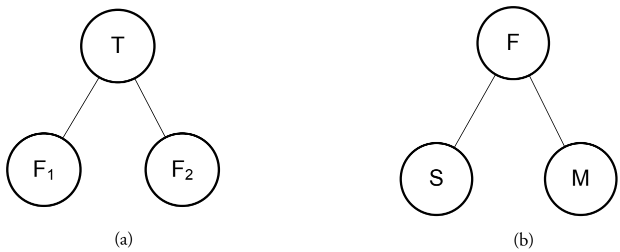
Figure 19. Results of Simulation 4 compared with behavioral data (adopted from MacLeod, 1991), showing average reaction time means and standard deviations. Human variance data was not available.

Simulation results

In the simulation (15 simulated subjects, each performing 20 trials in each condition) 5% errors were made on average (all during incompatible trials) and one subject was excluded from analysis due to having more than 30% error trials. After removal of error trials, the results showed that responses were fastest with compatible trials (M = 19.22 cycles, SD = 0.14), intermediate with neutral trials (M = 25.27 cycles, SD = 0.24) and slowest with incompatible trials (M = 39.53 cycles, SD = 0.65). The global fit between simulation results and behavioral data is depicted Figure 19. Note that the Stroop simulation results point more strongly to an interference effect with non-compatible stimuli than to facilitation with compatible stimuli, a result that is also found in behavioral studies (MacLeod, 1991). In our simulation this is due to the stronger weights from visual shape sensory codes to word feature codes (see Appendix).

Discussion

This chapter attempts to address how and why compatibility effects arise in stimulus-response translation. These effects demonstrate that some aspects of stimulus-response translation occur automatically. As demonstrated in the simulations, HiTEC is able to account for these effects. In fact, SRC is an *inevitable* consequence of HiTECs structures and processing characteristics as we will now explain. First, in order to internalize task instructions into a task set, both stimuli and responses need to be represented on a distal level and associated through task codes (see Figure 20a). Secondly, actions are represented in terms of perceptual effects and therefore use the same distal codes as stimuli and, consequently, are grounded in the same perceptual world (Prinz, 1992; illustrated in Figure 20b).This means that code overlap is possible and – to the extent that stimuli and responses overlap in the external environment, such as spatial correspondence — very probable. Finally, HiTEC assumes integrated processing which means that stimulus coding and response coding also overlap in time. Thus, the task set results in a pathway mediated by task codes and defined in distal features, and in probable code overlap of these same distal features; as stimulus processing and response planning occur simultaneously, the cognitive system inevitably needs to combine task-driven and automatic feature code activation. As a result, code overlap between stimulus and response features results in either facilitation or interference effects (Hommel, 2004).



(a) (b)

Figure 20. Schematic depiction of couplings between sensory codes, motor codes, feature codes and task codes. Panel (a) depicts a relation between task codes and feature codes that is part of the task set. F1 refers to a task relevant stimulus feature and F2 refers to an action effect feature of the required response. Panel (b) shows that feature codes are common codes, relating to both sensory and motor codes.

In Simulation 3, the simulation of the Simon effect, stimulus-response compatibility follows from the fact that responses are coded in terms of their spatial perceptual consequences (due to ideomotor learning, see Chapter 3). That is, left or right keypresses. In order to plan one of these keypress actions, the model needs to activate either the 'left' or 'right' feature code. Now, when a stimulus is presented left or right, the 'left' and 'right' feature codes will be activated both due to the exogenous excitation resulting from the presented stimulus and due to the endogenous excitation due to their roles as action effect features. When both stimulus perception and response anticipation activate the same 'left' or 'right' feature code, overall stimulus-response translation is faster, constituting a compatible trial. When they do not activate the same but competing codes, stimulus-response takes longer, constituting an incompatible trial.

In similar vein, in the simulation of the Stroop effect, the task irrelevant word feature only has influence because the response is coded using these features (which is a result from the action–effect learning). If the response is not verbally defined (e.g., in terms of key presses) the compatibility effect is dramatically reduced in behavioral studies (MacLeod, 1991). In HiTEC this would result in a different set of action effect features to be associated to the motor codes. Hence, code overlap with stimulus features would cease to occur, effectively eliminating the compatibility effect.

In typical computational models of SRC effects, such as the Simon effect, stimuli are represented in terms of non-spatial task-relevant codes (e.g., 'high tone' and 'low tone') and spatial task-irrelevant codes (e.g., 'left tone' and 'right tone'), and responses are also represented in terms of spatial codes (e.g., 'left key' and 'right key'). As depicted in Figure 21, stimulus codes and response codes are connected using two routes (e.g., Kornblum et al., 1990; Zorzi & Umilta, 1995; De Jong, Liang, & Lauber, 1994). A direct route connects the spatial stimulus codes to the corresponding spatial response codes, which is assumed to reflect the automatic process. The task instruction (e.g., "*when you hear a high tone, press the left key*") is implemented as a soft-wired connection from the non-spatial stimulus code (e.g., 'high tone') to a spatial response code (e.g., 'left key'), following the task instruction. This is assumed to reflect the controlled process. When a stimulus is presented, activation is propagated through the model towards the response codes. The response code that first reaches an activation threshold will be selected for execution. Now, when a compatible stimulus is presented (e.g., a high tone presented on the left), both the hard-wired spatial connections and the soft-wired task instruction-based connections contribute to a speedy activation of the correct response code. Conversely, when an incompatible stimulus is presented (e.g., a high tone presented on the right), the direct route activates the incorrect response. The controlled route, however, activates the response determined by the task instruction, which eventually is assumed to win this competition. As a result, processing incompatible stimuli results in longer reaction times than processing compatible stimuli. In sum, in dual route models, the stimulus–response compatibility effect arises from the interplay between the direct route, reflecting automatic comparison between spatial stimulus and response codes, and the controlled route, reflecting the task instructions. Thus, to account for SRC effects, these models drive on three main assumptions: (1) responses are represented by spatial codes, (2) attending to a stimulus automatically produces a spatial stimulus code, and (3) the outcome of a comparison between the spatial stimulus code and the spatial response code produces the compatibility effect. Here, this comparison is assumed to occur automatically and arise from the idea that stimuli and responses are similar (e.g., '*have dimensional overlap*', Kornblum et al., 1990; 1999).

Clearly, there are some strong similarities between these dual route models and HiTEC. First, the basic dynamic activation mechanisms of these models (i.e., codes, connections, activation levels) are very similar to HiTEC's connectionist implementation, and second, the general structure of the HiTEC model instance used to model the Simon (and Stroop) effect also shows some resemblance to 'two routes' (i.e., a route through the task codes and a route through the common codes). However, HiTEC does not share the main assumptions of the (strictly feedforward) dual route models and provides a different rationale for SRC. With respect to the main assumptions listed above, HiTEC assumes that (1) motor codes and representations of their perceptual effects are learned, allowing for the emergence of situation-specific meanings of actions (see Chapter 3), (2) task sets are implemented using common distal feature codes and recurrent connections with task codes. Including a feature code as response feature automatically makes it susceptible to stimulus based exogenous excitation and (3) compatibility between stimuli and responses (i.e., action effects) is due to the degree they are represented using the *same* common codes. These assumptions follow directly from key characteristics of the HiTEC model and do not require a notion of 'dimensional overlap' or 'similarity' that selectively applies to some combinations of stimuli and responses and not to others.

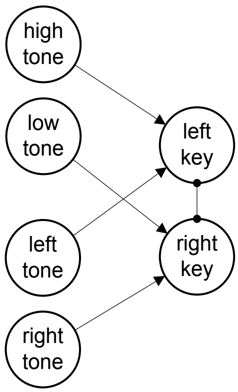


Figure 21. Dual route account of Simon effect (adapted from Zorzi & Umilta, 1995).

Also, the ideomotor learning of action-effect associations as employed in the simulations in this chapter allows for the flexibility and context dependence that is shown in a variety of SRC studies (see Chapter 3 for an elaborate overview). Moreover, in HiTEC task sets are implemented using recurrent connections only. These connections strictly follow the actual task instructions. In comparison, the Dimensional Overlap model (Kornblum et al., 1990; 1999), in addition to the controlled and automatic connections, assigns different activation dynamics to task relevant stimulus features than to task irrelevant stimulus features. Hence, we argue that HiTEC allows for a more parsimonious approach to controlled and automatic stimulus-response translation and provides a rationale – based on representations and processes – for *why* these SRC effects occur.

A related model of the Stroop effect (Cohen et al., 1990) also contains two routes. In this model, however, 'automatic' and 'controlled' is considered to depend on experience which they address explicitly. The model further allows for modeling multiple tasks (naming the ink color vs. naming the color word), showing somewhat of the task flexibility demonstrated by the HiTEC model. Task implementation in this model, however, is confined to injecting additional input to either one out of two task nodes thereby biasing the model to either one of the two implemented tasks.

The SLAM model (Phaf, van der Heijden, & Hudson, 1990) for attention in visual selection tasks is also used to model the Stroop effect. This connectionist model consists of multiple interacting levels of representation and employs two main processes, object selection and attribute selection, to perform a variety of filtering tasks. In order to account for the Stroop effect additional connections between stimulus features and response aspects are assumed ("privileged links") in similar vein as the automatic route in the dual process models described above.

Other models that include perception and action systems, such as the models by Ward (1999) and by Botvinick et al. (2009) do not address SRC; in these models stimulus features are simply connected to action features according to the task at hand; hence, stimulus features are just straightforwardly translated into action features. In contrast to the dual route models described above, however, connections in these models are recurrent. Hence, action activation can also influence stimulus perception, in similar spirit as HiTEC (see Chapters 2 and 3 for a more detailed comparison).

Another well-known SRC effect, which we did not explicitly model in HiTEC, is the Flanker effect (Eriksen & Eriksen, 1974). This effect is observed when participants are required to respond to a visual target with close-by distractors (flankers) which they are unable to ignore. For instance, if a discriminative response is required for a central target letter that is flanked by distractors, participants are faster if target and distractors are associated with the same response than with different responses. This result suggests that also for distractors the associated responses are activated and that this activation interacts with producing the response to the target. The Flanker effect is modeled by Cohen and Shoup (1997). In their model, displays of multiple stimuli are processed in terms of their individual features, which include location information. This process works separately for each feature dimension. At this stage, response competition is assumed to occur possibly yielding congruency effects. Finally, response activation from multiple dimensions is combined into a single actual response. Cohen and Shoup (1997) propose that the Flanker effect results from within-dimension competition. This set up somewhat resembles HiTECs architecture. Motor codes (responses) are associated to feature codes (features in dimensions). In contrast, however, HiTEC does not confine response competition within dimension, but rather assumes a model-wide integrated competition process. Crucially, to simulate the Flanker task, a model must be able to process a display of multiple objects and selectively treat one object as the 'target' and the others as 'distractors based on their location in the display. The HiTEC model currently does not provide for such differentiation but see (Cohen, Servan-Schreiber, & McClelland, 1992) for a PDP model of the Flanker effect.

To summarize, existing models of congruency in stimulus-response translation typically assume spatial response codes and special links between stimulus features and these response codes based on a certain 'similarity'. HiTEC does not need such assumptions as congruency effects follow naturally and inevitably from using common codes for both stimulus and response (i.e., action effect) representation.

Interestingly, dual route systems have also been proposed to account for fast and automatic responses to *affective stimuli* (LeDoux, 1996). In such a system, a 'low road', associated with the amygdala, automatically translates stimuli to responses. In parallel with this subcortical pathway there is a 'high road', associated with the cortical structures of the brain. This pathway analyzes the stimulus in a more fine-grained, but slower way. Together, these routes enable someone to respond quickly to affective stimuli and to process these stimuli in more detail in order to adjust behavior at a later point in time. Recent studies show that automatic processes may be affected by top-down influences (e.g., Beckers et al, 2002). The simulations in this chapter show that HiTEC is able to account for such influences. In Haazebroek et al. (2009b; 2011b) this is more explicitly applied to affective processing in a simulation of an affective version of the Simon effect (Beckers et al., 2002).

Although HiTEC accounts for some aspects of automatic processing, it must be noted that automaticity is a much broader field than these SRC effects alone suggest (see Moors & De Houwer, 2006 for an overview). Indeed, there is a long history of theorizing on the struggle between human will and habit (for a prototype, see Ach, 1910). With respect to the SRC effects discussed in this chapter, alternative explanations for automatic, uncontrolled or unconscious behavior include storing and retrieving action instances (Logan, 1988), integrating 'chunks' of behavior (Anderson, 1992) and over-learning of stimulus-response translation (Proctor and Lu, 1999; Tagliabue, Zorzi, Umiltà, & Bassignani, 2000). In this thesis, however, we have focused on aspects of automaticity that naturally follow from a set of key characteristics of our connectionist model of perception and action planning. Moreover, in HiTEC, important components of cognitive control are actually assumed to be exerted already before responding to any stimuli. This includes the prerequisites for code overlap, so that—somewhat paradoxically—automaticity is the result of control (Hommel, 2000a). In effect, we have eliminated the difference between automatic and controlled information processing in the model (i.e., everything is automatic). One could argue that this is there is more to cognitive control than modeled in current simulations. With respect to the simulated experimental paradigms, however, it seems that other types of (online) control are unnecessary.

Although we have explained how and why automaticity occurs in the HiTEC model by means of code overlap, one could still wonder *why* this would be beneficial for coordinating our behavior. Clearly, being slower or faster in a Simon task does not provide one immediate evolutionary advantages. However, even though the presence of such effects is convenient for the scientific study of perception-action relationships, their real benefit is prevalent in everyday life: object properties (e.g., location, shape) must often be translated into very similar action parameters (location, shape of hand) in order to efficiently interact with the environment. Perceiving an object and internally coding its features would therefore be likely to specify and literally prepare important components of the action plan that the given object affords (Hommel, 2009). Thus, rather than explicitly translating these stimulus features into response features (e.g., 'if big object, use large grasp action'), automaticity – in our framework using common codes (e.g., 'big') – allows for implicit, effortless translation of matching features.

To conclude, we have addressed how and why automaticity occurs in stimulus-response translation. In the HiTEC connectionist model stimuli and responses are represented using common codes. In typical SRC tasks, responses are defined in terms of features that are shared by the stimuli to be responded to. This means that a task set not only defines a controlled pathway but also an automatic translation path through the common codes used both for stimuli and responses (cf. Hommel, 2000b). In this chapter we focused on automaticity, in the next chapter we will discuss the role of task context more explicitly.

Chapter 5
Task Context

This chapter is an integration of major parts of the following articles:

Haazebroek, P., Raffone, A., & Hommel, B. *HiTEC: A Connectionist Model of the Interaction between Perception and Action Planning*. Manuscript submitted for publication.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2013). How task goals mediate the interplay between perception and action. F*rontiers in Psychology, 4:247*.

Laboratory tasks in behavioral research are commonly not self-explaining and human participants are thus commonly 'task ignorant' until they receive the task instruction. That is, they do bring the general ability to perceive and act, but the task instruction makes them respond to particular stimuli with particular responses, just as needed. As a consequence, the same individual is able to participate in various experiments as long as he/she receives appropriate task instructions. Now, how does the cognitive system configure itself to perform a specific task? Here, intuition might suggest that cognitive control follows perception and precedes action planning, so that a stimulus is being responded to appropriately. Interestingly, findings on two-interactions between perception, cognition and action suggest otherwise.

The interaction between perception and cognition, for example, can be demonstrated by so-called spatial congruency effects. Several studies have found interactions between the meaning of words and the spatial position of those words on the computer screen. For example, people respond faster to a word such as *helicopter* or *stork* when it is presented at the top of the computer screen than when it is presented at the bottom of the screen (Šetić & Domijan, 2007). Other studies showed that the spatial meaning of a word may attract attention to a particular location on the screen (e.g., Estes, Zerges, & Barsalou, 2008; Zanolie et al., 2012). Spatial congruency effects are also found with words referring to abstract concepts that are metaphorically connected to spatial locations, such as power (Schubert, 2005; Zanolie et al., 2012), valence (Meier & Robinson, 2004), divinity (Meier et al., 2007) or magnitude (Fischer, Castel, Dodd, & Pratt, 2003; Pecher & Boot, 2011), but see Lakens (2012) for an alternative explanation, based on polarity correspondence (Proctor & Cho, 2006). Furthermore, studies have shown that perceiving motion in a particular direction interacts with the processing of sentences or words describing motion in the same direction (e.g., Kaschak et al, 2005; Meteyard et al., 2007; 2008).

Likewise, spatial congruency effects also occur in the interaction between cognition and action. For example, participants are faster to respond to a sentence when the direction of the response matches the direction of the action described in the sentence. This so-called *action compatibility effect* has been found with different kinds of movement, such as moving the hand toward or away from the body (Glenberg & Kaschak, 2002) and rotating the hand (Zwaan & Taylor, 2006). These results are taken as evidence that the representations underlying conceptual processing partially overlap with the representations underlying the preparation and execution of action.

Finally, spatial congruency effects occur in the interaction between perception and action. Much research has been devoted to stimulus–response compatibility (SRC) effects; the canonical example being the Simon effect (Simon & Rudell, 1967; Hommel, 2011). In the typical Simon task, stimuli vary on a spatial dimension (e.g., randomly appearing on the left or right) and on a non-spatial dimension (e.g., having different colors). Participants have to respond to the non-spatial stimulus feature by performing a spatially defined response (e.g., pressing a left or right key). Although the location of the stimulus is irrelevant for the response choice, it nevertheless influences the response time and accuracy, suggesting interaction between stimulus perception and response planning. Participants respond faster (and more accurately) when the stimulus location is congruent with the response location than when the stimulus location is incongruent with the response location. The Simon effect is simulated and discussed more elaborately in Chapter 4.

Although spatial congruency effects seem to demonstrate compatibility effects that However, the various spatial congruency effects discussed above suggest an interaction between cognition and perception and between cognition and action. Hence, it is to be expected that the (cognitive) task set may influence the automatic translation from spatial stimulus codes to spatial response codes. Indeed, various studies have demonstrated that SRC effects are strongly influenced by the task. For instance, Riggio et al. (1986) reported that when participants responded with sticks that were either parallel or crossed, the Simon effect was found to relate to the stick end position, not to the hands holding the sticks. In a study by Guiard (1983), participants had to respond with a steering wheel. Their results suggest that not the position of the hands but the steering direction (as in a car) determines the Simon effect, indicating an even more abstract notion of left or right responses. It is this task- and intention-dependent left-ness or right-ness, rather than the actual physical location of a response, that seems to interact with the spatial location of the stimulus and thereby yields the Simon effect (see Chapter 3 for more elaborate discussion on the situation-specific meaning of actions).

Moreover, Hommel (1993) showed that the Simon effect as described in Simulation 3 can be inverted by just changing the task instruction. In this study participants responded with left or right keypresses to the high vs. low pitch of tones which were presented left or right. When a key was pressed a flash light was presented on the opposite side of the keypress. One group was instructed to "press the left/right key" in response to the low/high pitch of the tone, whereas another group was instructed to "flash the right/left light" in response to the low/high pitch. In other words, all participants carried out exactly the same movements in response to the same stimuli, but one group did that "in order to press the keys" while the other did it "in order to flash the lights". This seemingly minor manipulation had a major impact on the Simon effect. Whereas the Key group showed a standard Simon effect with faster responses for spatial correspondence between tones and keys, the Light group showed the opposite effect: faster responses for spatial correspondence between tones and lights. This observation demonstrates the crucial role of task instruction in stimulus and response coding.

The Theory of Event Coding (TEC) stresses that perception and action are flexible; that is, they are tuned to the current context and are subject to cognitive control (Hommel et al., 2001). Codes are assumed to be 'intentionally weighted'; that is, the strength of their activation depends on the task context (Memelink & Hommel, 2013). In this chapter, we explicitly address how the task context may be internalized into a task set and how this task set may modulate both controlled and automatic aspects of stimulus-response translation.

In the first simulation we model the empirical study by Hommel (1993) described above. Here, the task set wires the model in such a way that either the light or the key locations are more strongly perceived and integrated in action-effect associations. This has consequences for subsequent stimulus-response translation. Interestingly, the HiTEC model of this study raised a new question: on what level does 'intentional weighting' operate? To empirically test this, we have conducted a novel experiment using a Wii Balance Board. To assess the differences between this experiment and the Hommel (1993) study, we have also simulated the new experiment. Finally, discuss both simulations and compare our approach with related work on assessing the influence of task context in stimulus-response translation.

Simulation 5: Inverting the Simon effect

HiTEC simulation

The empirical study by Hommel (1993) discussed above, was simulated in HiTEC using two instances of the model. One instance is configured according to the Key instruction, the other the Light instruction. The latter condition is depicted in Figure 22. Note that the difference in task instructions is reflected in the task connections alone. Crucially, in the Key condition the mere connections between 'Key' and the task codes enhance the processing of haptic locations. In contrast, in the Light condition, the connections between 'Light' and the task codes enhance visual locations. This specific wiring biases the action-effect learning and the direction of the compatibility effect during subsequent experimental trials.

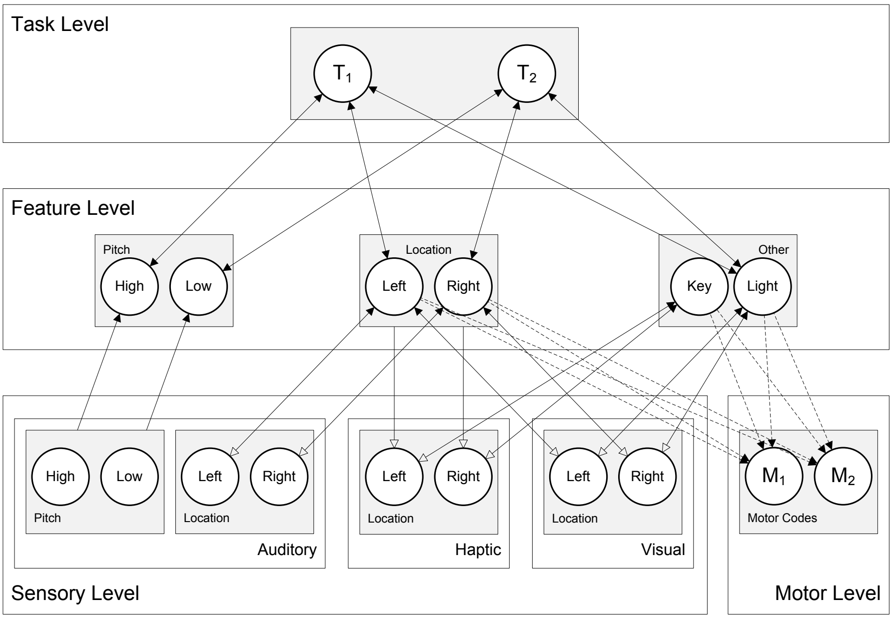


Figure 22. Specific HiTEC Model for Simulation 5. This is the model instance for the Light condition. The Key condition differs only in the connections from 'T1'and 'T2' to 'Key' instead of 'Light'. ¬These task code – feature code connections moderate the learning of feature code – motor code connections as the 'Light' feature code (in the "Light" condition; the 'Key' feature code in the Key condition) sends activation to and therefore receives activation from the task codes during learning. As the location feature codes are used to code for stimulus location (i.e., location of the tone) and both action effects (haptic and visual), compatibility effects arise. The task instruction determines which action effect (haptic vs. visual) is top down enhanced and therefore determines the major determinant (and consequently the direction) of the compatibility effect during the experimental trials. Note that the connections between feature codes and motor codes (dashed lines) are depicted in a way that both action effects are taken into account. Some of these connections (as determined by the enhancement of the 'Light' or the 'Key' code) become stronger than others, but this is a relative weighting, as both action effects are present in the environment.

Simulation results

The two groups consisted of 15 simulated subjects each, each performing 60 trials. On average 17% of the trials were errors (all of them in the stimulus-key incompatible condition). No simulated subjects were excluded from the analysis. After removal of error trials, the Key group showed fastest responses with congruent stimulus-key trials (*M* = 19.81 cycles, SD = 3.23), intermediate responses with neutral trials (*M* = 23.74 cycles, SD = 2.35), and slowest responses with incongruent stimulus-key trials (*M* = 26.67 cycles, SD = 1.13). In contrast, in the 'Light' group congruent stimulus-key trials (i.e., trials in which the stimulus location was incongruent with the action-effect light location) yielded the slowest responses (*M* = 25.83 cycles, SD = 1.15), neutral trials intermediate responses (*M* = 23.35 cycles, SD = 1.13), and incongruent stimulus-key trials (i.e., trials in which the stimulus was congruent with the action-effect light) the fastest responses (*M* = 19.75, SD = 3.9). As depicted in Figure 23a the simulation results provide a good fit with the empirical data shown in Figure 23b. This pattern demonstrates that a slight change in the instruction can generate a different task set, which again leads to an inversion of the commonly robust Simon effect; and it shows that HiTEC is equipped to simulate the resulting pattern very closely. It also demonstrates that an associative account of perception-action interactions need not be inconsistent with cognitive flexibility and the possibility of adaptive task-set configuration as evidenced by the inversion reported by Hommel (1993).

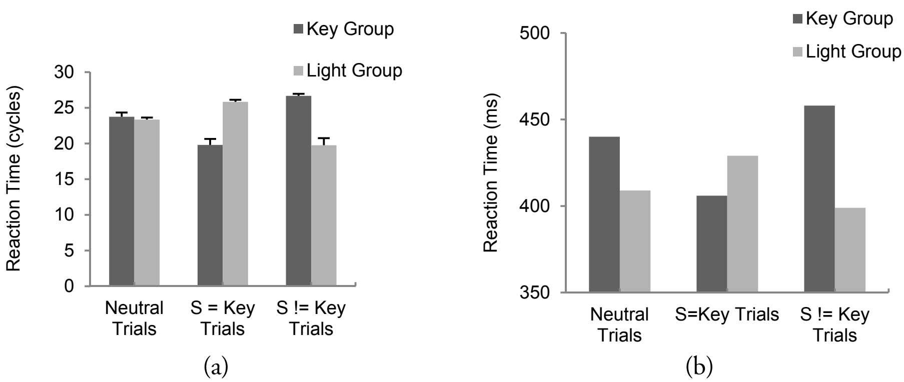


Figure 23. Results of Simulation 5 compared with behavioral data (adopted from Hommel, 1993). Human variance data was not available. The horizontal axis denotes the different trial congruency levels.

Conclusion

As the empirical findings by Hommel (1993) suggest and Simulation 5 demonstrates, intentional weighting can also affect the coding of response representations. In Hommel (1993) it can be argued that the task set results in stronger weighting of key vs. light location, depending on the instruction. One could ask, however, whether this actually implies weighting of feature dimensions. Indeed, on closer examination in the HiTEC model instance used in Simulation 5 (Figure 22), both the key and the light locations are represented by the *same* spatial feature dimension (i.e., left-right). Therefore one could argue that not feature dimensions, rather the respective *sensory dimensions* are selectively enhanced by top down task influences by means of the 'key' vs. 'light' instruction. In other words, the task instruction determines whether a participant attends to either the (visual) light locations or the (haptic) key locations. Subsequently, the attended locations get encoded on the single spatial left-right feature dimension. The fact that this same left-right feature dimension is also used to encode the stimulus location forms the basis of the observed SRC effect.

Empirical study: Feature weighting

As discussed above, our simulation of the Hommel (1993) experiment suggests that in that experiment, intentional weighing operated *within* the same feature dimension ('key' vs 'light'), effectively modulating the contributions of multiple *sensory* dimensions to the *same* 'left'/'right' feature dimension. Indeed, results from Memelink and Hommel (2005) demonstrated that mere task instruction may *not* be sufficient to affect action coding if the manipulation does not change the task *goal*. The question then arises: what constitutes a task goal? Does one need to attend to different objects (e.g., key vs. light) in the environment to selectively enhance sensory coding? Or does the intentional weighting principle apply to more abstract feature codes as well? In the present study we assess the influence of task instruction on automatic processes in stimulus–to–response translation at the feature level. In the design of the task there are two important criteria to take into account: (1) the experimental set up needs to employ a *single object* and a *single sensory dimension* which can be encoded in *two different feature dimensions*, based on the task instruction. In this way, we can rule out the role of purely object based attention; (2) the experimental set up needs to use a task in which two different interpretations of the same ambiguous movement are – to a certain extent and in the eyes of the participant – equally intuitive and applicable to the observed (sensory) effects of the physical movements. Otherwise, if participants can easily recode the variations in these dimensions into a single intuitive dimension, they will do so; the influence of task instruction will then disappear (cf., Memelink & Hommel, 2005).

With these criteria in mind we opted for a relatively natural scenario rather than responding by pressing keys (see Wang, Proctor & Pick, 2007; Yamaguchi & Proctor, 2011 for similar approaches). In a natural scenario – we hypothesized – participants would be more strongly compelled to adhere to the action coding specified by the task instruction. In the present study, participants stood on a Wii balance board and were instructed to imagine standing on either a snowboard or a pair of skis. They had to respond to stimuli by leaning sideways. In the ski condition, this lateral movement was presented as moving the skis to the *"left"* or *"right"*, whereas in the snowboard condition, it was presented as moving the snowboard *"backward"* or *"forward"*. In performing the task, participants could draw on their own motor experience if they had any experience with skiing or snowboarding. Participants who had never skied or snowboarded could still form a mental representation of what it means to be skiing or snowboarding, by combining elements from partial or similar experiences (Barsalou, 2008; Taylor & Zwaan, 2009). For example, they could draw on visual experience (e.g., watching snowboarders on TV), and combine this with related motor experience (e.g., surfing or skateboarding).

In the experiment, the Wii balance board was oriented diagonally towards the screen displaying the stimuli (Figure 24). The critical stimuli consisted of colored arrows pointing in one of four directions (backward, forward, left or right). The study used a between-subjects design; participants were either instructed to imagine standing on a pair of skis or on a snowboard, and to respond to the stimulus color by leaning sideways. Given the diagonal orientation of the balance board, the responses simultaneously varied on the left– right dimension and on the forward– backward dimension. We expected that the weighting of the (feature) dimensions would depend on the instruction given to the participant. A skier stands in the same direction as her skis. When she leans to the left or right, this causes the skis to turn into the respective direction. Therefore, participants in the ski condition would encode the lateral leaning movements as 'left' and 'right'. In contrast, a snowboarder stands on a snowboard perpendicular to its direction of movement. When she leans sideways, the snowboard will slide forward or backward. As a result, we expected that participants in the snowboard condition would not only encode the movements as 'left' and 'right', but also as 'forward' or 'backward'. Therefore, we expected a forward– backward congruency effect to occur in the snowboard condition, but not in the ski condition.

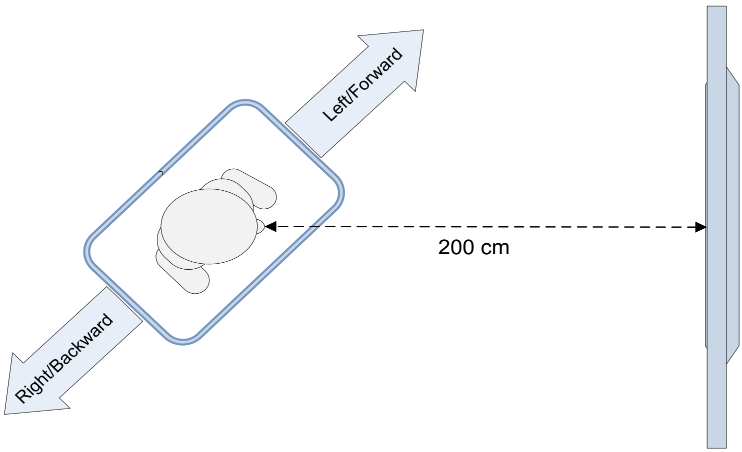


Figure 24. Setup of the experiment.

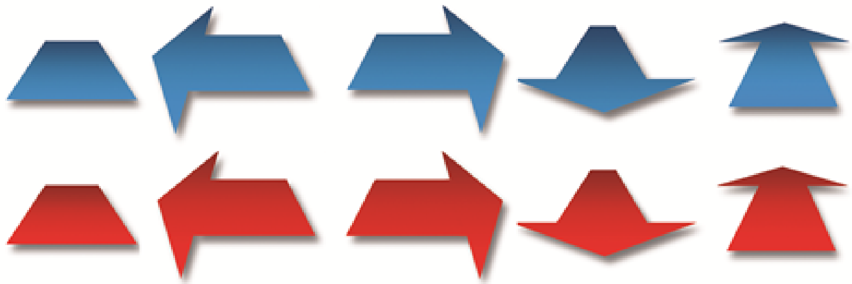


Figure 25. Experimental stimuli. Arrows pointing forward, backward, left, right and direction-neutral stimulus.

In the next section we describe the methods of the behavioral experiment. We continue with presenting the results, followed by a HiTEC simulation of the study. Finally, we discuss the implications of both our empirical findings and simulation results.

Material and methods

Participants. A total of 83 Dutch undergraduate psychology students from Leiden University (65 women, 18 men) took part in the experiment. In return for their participation they received course credits or a monetary reward of EUR 4.50. Mean age of the participants was 19.8 (SD 2.3).

Apparatus and Stimuli. The instructions and stimuli were presented on a television monitor with a diameter of 107 cm and a refresh rate of 60 Hz. E-Prime software was used to present the stimuli. Stimuli were blue or red symbols, consisting of one direction-neutral stimulus and arrows pointing in one of four different directions; left, right, forward or backward (Figure 25). On screen, each stimulus measured approximately 30 x 30 cm.

Participants stood on a Wii balance board (51 cm long x 32 cm wide x 5 cm high), which was placed diagonally, at an angle of 45° or -45°, in front of the monitor. In order to be able to face the monitor, participants who were positioned at the 45° angle always had their left foot forward (i.e., closest to the monitor), and participants at the -45° angle always had their right foot forward. Thus, the participant's position with respect to the computer screen was determined by the orientation of the balance board.

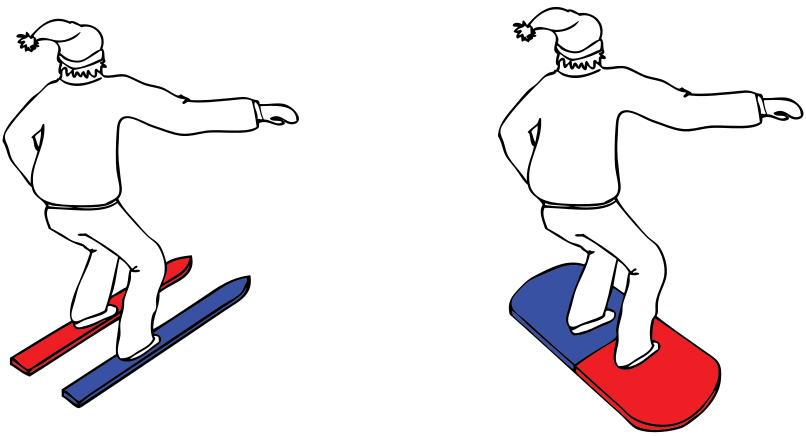
The distance between the monitor and the center of the balance board was 200 cm (Figure 24). The orientation of the balance board was counterbalanced across participants. Half of the participants stood with their left foot forward, the other half stood with their right foot forward. The participant's weight distribution on the left–right axis and front–back axis of the balance board was recorded at a frequency of 100 Hz. This was done by custom-made software that polls the sensor values of the balance board, using a Bluetooth connection. To respond to a stimulus, participants had to lean sideways far enough to exceed a predefined threshold on the left–right axis of the balance board. When this threshold was exceeded, the response time and accuracy of the response were logged.

Procedure. The complete experiment lasted approximately 30 minutes. Upon arrival to the lab, participants were randomly assigned to one of eight counterbalance versions (see Table 1), defined by the instruction (snowboard or ski), the orientation of the balance board (45° or -45°) and the stimulus–response mapping (red–left/blue–right or red–right/blue–left). Participants in the snowboard condition received the following instruction: "*Imagine that you're standing on a snowboard, which you can move forward or backward by leaning on your front or back leg*", whereas participants in the ski condition received the alternative instruction: "*Imagine that you're standing on skis, which you can move to the left or right by leaning on your left or right leg*". To enhance the context of the task, an illustration of a skier or a snowboarder was presented, standing in the same position as the participant on the balance board (see Figure 26).

The instruction was followed by a practice block, which contained 24 trials. Each practice trial started with the presentation of the sentence "*Take the start position*" for 1000 ms. Next, the instruction to lean into a particular direction (e.g., *"Move the skis to the left (left leg)*" or *"Move the snowboard forward (front leg)"*) was presented until the participant responded by leaning into the respective direction. In the snowboard condition, the directions were *"backward"* or *"forward"*, whereas in the ski condition the directions were *"left"* or *"right"*. To enhance the encoding of the movements in the appropriate dimension, participants were instructed to mention out loud the direction in which they had to lean. Following a correct response, the word "*correct*" was presented for 1000 ms. Following a response that was incorrect or too slow (more than 5000 ms), the word *"error"* or *"too slow"* was presented for 1000 ms.

Table 1. Overview of the eight different counterbalance versions of the experiment.

Task	Position	Instruction
Ski	45° (left foot forward)	If the image is blue, lean to the left If the image is red, lean to the right
		If the image is blue, lean to the right If the image is red, lean to the left
	-45° (right foot forward)	If the image is blue, lean to the left If the image is red, lean to the right
		If the image is blue, lean to the right If the image is red, lean to the left
Snowboard	45° (left foot forward)	If the image is blue, lean forward If the image is red, lean backward
		If the image is blue, lean backward If the image is red, lean forward
	-45° (right foot forward)	If the image is blue, lean forward If the image is red, lean backward
		If the image is blue, lean backward If the image is red, lean forward



(a) (b)

Figure 26. Illustrations of (a) skier and (b) snowboarder used during instruction.

After completing the practice trials, participants received the instruction for the experimental trials. They were instructed to respond to the stimulus color by leaning into a particular direction. In the snowboard condition, participants had to respond to red or blue stimuli by leaning forward or backward (e.g., *"If the image is red, lean forward"*). In the ski condition, participants had to respond to red or blue stimuli by leaning to the left or right (e.g., *"If the image is red, lean to the left"*). The actual mapping of color to direction was counterbalanced across participants. In addition, participants were urged to respond as quickly and accurately as possible.

The instruction was supported by the illustration of the skier or snowboarder, in which the two skis or the two sides of the snowboard were colored in the corresponding stimulus color (for example, a skier with a red left ski and a blue right ski, see Figure 26).

Each trial was either neutral (the neutral shape), left–right congruent (left- or right-pointing arrow, corresponding to the horizontal direction of the response), left–right incongruent (left- or right-pointing arrow, opposite to the horizontal direction of the response), forward–backward congruent (forward- or backward-pointing arrow, corresponding to the forward–backward direction of the response) or forward-backward incongruent (forward- or backward-pointing arrow, opposite to the forward–backward direction of the response).

The experiment was divided into four blocks with 50 trials each. Since there were 10 different stimuli (two colors; red and blue, and five orientations; backward, forward, left, right and neutral), each stimulus was repeated five times during each block. Stimuli were presented in random order. A trial started when the participant had taken the start position and his/her balance was centered on the Wii balance board. After 500 ms, a black fixation cross was presented for 1000 ms, followed by the experimental stimulus. The stimulus remained on the screen until the participant's response was recorded or until 5000 ms had elapsed. If the response was incorrect or too slow, a feedback screen was presented for 2000 ms, displaying the word "*error*" or "*too slow*". If the response was correct, no feedback was given. After completing a trial, participants had to return their balance to the center of the balance board. Following each block of 50 trials, there was a short break of 10 seconds, during which the instruction was repeated. The instruction was visually supported by the same illustration of the snowboarder or skier that had been shown in the initial experimental instruction (Figure 26).

After completing the experimental trials, participants indicated whether they had any experience with skiing or snowboarding. Experienced snowboarders also indicated whether they preferred to snowboard with their left foot forward or their right foot forward

Results

The data from eight participants were discarded because they had an overall accuracy level lower than 0.70. For the remaining participants (38 in the Ski condition and 37 in the Snowboard condition) we computed mean reaction times and accuracy for the responses. Incorrect responses (7.8 %) were excluded from the reaction time analysis. Furthermore, based on Tukey's criterion, reaction times below 415 ms and above 1590 ms (5.3%) were also discarded. Mean trimmed reaction times and error rates are presented in Table 2. The reaction times were analyzed with a 2 x 2 x 2 repeated measures ANOVA, with dimension (backward–forward vs. left–right) and congruency (congruent vs. incongruent) as within-subject variables, and instruction (ski vs. snowboard) as between-subject variable.

The majority of participants (27 in the ski group, 18 in the snowboard group) had no experience with snowboarding or skiing, 14 participants had only ski experience (6 in the ski group, 8 in the snowboard group), 5 participants had only snowboard experience (2 in the ski group, 3 in the snowboard group), and 11 participants had both ski and snowboard experience (3 in the ski group, 8 in the snowboard group). Because of the small number of participants in some of the groups, we ignored this factor in the analysis.

Table 2. Mean response times (ms) and standard deviations for the different trials in the two instruction conditions.

Instruction	Dimension	Congruent	Incongruent	Effect
Ski	Left-right	970 (154.6)	1056 (179.8)	84 ms
	Forward-backward	1006 (159.1)	1011 (169.7)	5 ms
Snowboard	Left-right	922 (134.5)	981 (153.4)	59 ms
	Forward-backward	950 (120.3)	966 (143.0)	16 ms

There was a main effect of congruency, with congruent trials being faster than incongruent trials, $F(1,73) = 108.4$, $p < .001$, $\eta_p^2 = .60$. In addition, there was a significant interaction between congruency and dimension, $F(1,73) = 72.5$, $p < .001$, $\eta_p^2 = .50$. The congruency effect was larger for the left–right dimension than for the backward–forward dimension. This finding is in line with the left–right prevalence effect found in other studies (e.g., Nicoletti & Umilta, 1984; 1985; Nicoletti, Umilta, Tressoldi, & Marzi, 1988). Different accounts are given for this effect (see e.g., Hommel, 1996; Proctor, Vu, & Nicoletti, 2003; Rubichi, Gherri, Nicoletti, & Umiltà, 2005). We will turn to this matter in the simulation results section. Most interestingly, there was a significant three-way interaction between congruency, dimension and task instruction, $F(1,73) = 7.1$, $p = .01$, $\eta_p^2 = .09$. On the left–right dimension, the congruency effect was significantly larger in the ski condition than in the snowboard condition, $F(1,73) = 4.5$, $p = .04$, $\eta_p^2 = .06$. The opposite result appeared on the front-back dimension; there was a significant congruency effect in the snowboard condition, $t(36) = 2.4$, $p = .02$, but not in the ski condition, $t(37) = 1.0$, $p = .33$. Although responses in the snowboard condition appeared to be faster in the snowboard condition than in the ski condition, there was no significant main effect of task, $F(1,73) = 2.7$, $p = .11$, $\eta_p^2 = .03$, because the between-subject differences were quite large.

Conclusion

Concluding, significant spatial congruency effects were found both in the left–right dimension and in the forward–backward dimension. Although the instructions did not cause a complete switch of the congruency effects, they modulated the relative size of the effects. On the left–right dimension, the effect was significantly larger in the ski condition than in the snowboard condition. On the forward–backward dimension, the effect was larger in the snowboard condition than in the ski condition. These results suggest that participants in the ski condition may have encoded the movements predominantly as ‘left’ and ‘right’, whereas participants in the snowboard condition may have encoded the movements also as ‘forward’ and ‘backward’. Before discussing our results in more detail, we will first present the HiTEC model and explain how this model can account for our findings.

Simulation 6: Feature weighting

In the experiment, a two-dimensional Simon task was used, with critical stimuli being colored arrows pointing in one of four directions (backward, forward, left or right). Participants stood on a Wii balance board, oriented diagonally towards the screen displaying the stimuli. They were either instructed to imagine standing on a snowboard or on a pair of skis and to respond to the stimulus color by leaning towards either the left or right foot. We expected that participants in the snowboard condition would encode these movements as forward or backward, resulting in a Simon effect on this dimension. This was confirmed by the results. The left–right congruency effect was larger in the ski condition, whereas the forward–backward congruency effect appeared only in the snowboard condition. The results can be readily accounted for by our connectionist model, HiTEC (see Chapter 2). Together, the

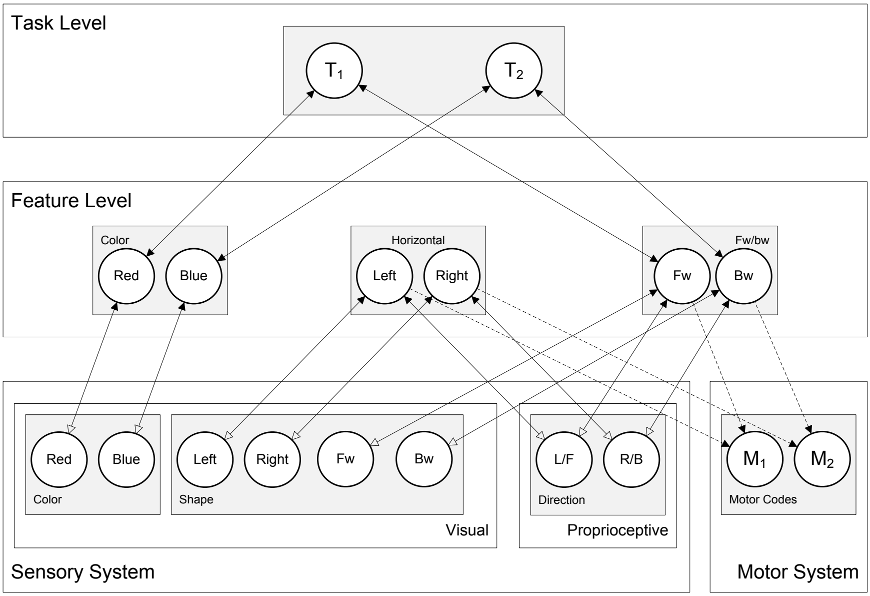


Figure 27. HiTEC model of the balance board task. Solid lines depict fixed connections, dashed lines are connections that are learned during action–effect learning. Depicted is the model in snowboard instruction condition, where the left leg is the front leg, and where a red stimulus requires a forward response (and a blue stimulus a backward response). Note that 'forward' and 'backward' feature codes are abbreviated as 'Fw' and 'Bw' and that 'L/F' denotes the ambiguous left/forward sensory code and 'R/B' the right/backward sensory code.

empirical study and the simulation using the connectionist model may contribute to a better understanding of the complex interaction between perception, cognition, and action.

HiTEC simulation

The current study involves colored arrow-shaped stimuli and responses that require a participant to move his/her balance to a certain direction (left/forward and right/backward). In order to be able to register these sensations, the HiTEC model is equipped with sensory maps for color, shape and proprioceptive direction. In addition, two movements are included in the motor map.

The task context includes instructions for responding to the stimulus color (*"red"* or *"blue"*), by moving either *"left"* vs. *"right"* or *"forward"* vs. *"backward"*, depending on the instruction group. We have included feature codes for these terms and have connected these codes to task codes appropriately. For each simulated subject, there are only two task rules to choose from, reflected by the two task codes in the task map. Figure 27 depicts the codes and connectivity for a simulated subject in the snowboard condition who was instructed to respond to red stimuli by moving forward, and to blue stimuli by moving backward, as can be seen by the connections between feature codes and task codes.

As illustrated in Figure 27, sensory codes are connected to feature codes. Stimulus related feature codes are connected to task codes and task codes to response related feature codes allowing activation to propagate from sensory codes to stimulus related feature codes to task codes to response related feature codes. Connections between feature codes and motor codes are explicitly learned. Importantly, in the current simulation, we have taken into account that the cognitive system has more experience with coding for 'left' and 'right' than is the case for 'forward' and 'backward'. In the model this is realized by setting the weights from sensory codes towards 'forward' and 'backward' slightly lower (0.3 rather than 0.4; see Appendix for all parameters).

Note that the sensory codes for proprioceptive direction (i.e., proprioceptive map in Figure 27) are not considered 'left' vs. 'right' or 'forward' vs. 'backward' by themselves. They represent two ambiguous sensations that can activate feature codes in both feature dimensions. We shall see that task context (i.e., the connections between feature codes and task codes, in close correspondence with the task instruction) determines to what extent this sensation is perceived as 'left' vs. 'right' or 'forward' vs. 'backward'.

The HiTEC simulation of the current empirical study consists of 40 simulated subjects in the ski condition and 40 simulated subjects in the snowboard condition. For each simulated subject, first the instruction is internalized by setting its task code–feature code connections appropriately; then, during 20 training trials feature code–motor code connections are learned, and finally, 20 repetitions of the 10 experimental trials (i.e., 2 colors x 5 shapes) are performed. This corresponds to the design of the empirical study as discussed above. Each individual simulated subject has its own random noise resulting in subtle individual differences in processing and in variance in behavior (i.e., varying reaction times) as is the case with individual human participants.

Simulation results

Table 3 shows the average number of cycles from stimulus onset until response selection for both instruction conditions and both congruency levels. As accuracy was 1.0 for all simulated subjects, it was not regarded in the analysis. The three-way interaction between congruency, dimension and task instruction found in the experiment was replicated in the simulation, as depicted in Figure 28. The left-right congruency effect was larger in the ski condition, whereas the forward-backward congruency effect was larger in the ski condition. We now explain how these results arose in the simulation by discussing the model dynamics in more detail.

Note that the HiTEC simulation only covers a part of the entire process of stimulus to response production in humans. The actual movements, for example, are included in the empirical reaction times (Table 2) but are not part of the simulation reaction times (Table 3). This results in larger relative effect sizes in the simulation results as compared to the empirical data.

Table 3. Average number of processing cycles from stimulus onset until stimulus selection in the HiTEC model, based on all 80 simulated subjects.

Instruction	Dimension	Congruent	Incongruent	Effect
Ski	Left-right	14.7	29.2	14.5
	Forward-backward	17.1	18.0	0.9
Snowboard	Left-right	15.6	27.0	11.4
	Forward-backward	15.7	21.0	5.3

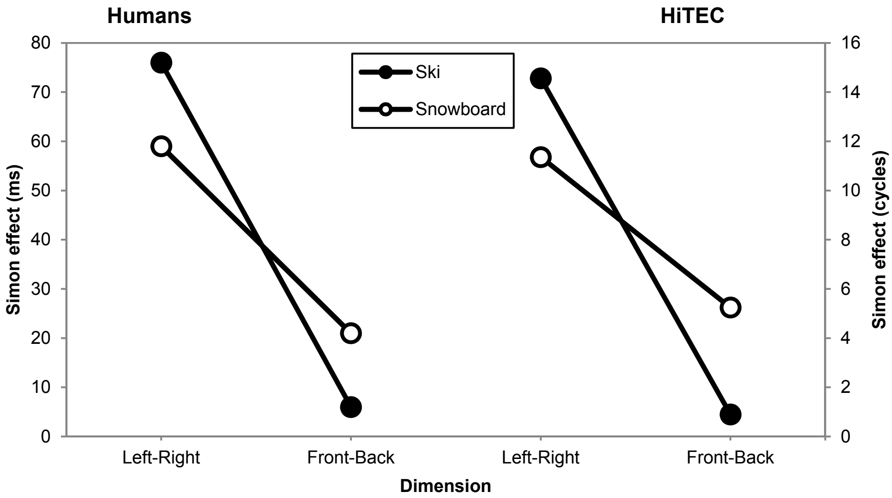


Figure 28. Comparison between human data (left) and simulation results (right). Lines depict the effect sizes for both instruction groups (ski and snowboard) and both congruency dimensions (left–right and forward–backward).

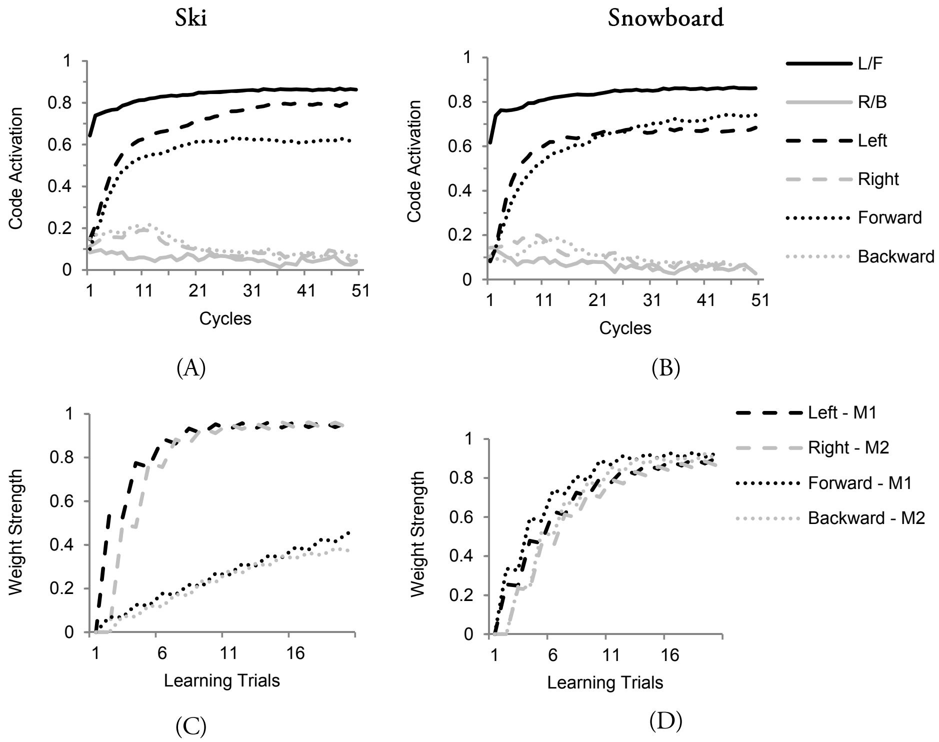


Figure 29. HiTEC simulation graphs of one simulated subject in the ski condition (panels A and C) and one simulated subject in the snowboard condition (panels B and D) during learning trials. Panels A and B show code activations resulting from the perception of the ambiguous action effect (balance towards 'left'/ 'forward'). Due to differences in task code – feature code wiring there is difference in recurrency and therefore slight differences in code activation ('left' vs. 'forward') in the two instruction conditions. In the panels C and D, that show the weight strength of a selection of feature code – motor code connections during all learning trials, it is clear that during the learning trials this difference in code activation accumulates to a substantial difference in the learned action–effect weights.

Model dynamics during simulation

Although the stimuli and responses are equal for both instruction groups, the congruency effects differ. These differences between the groups are the result of several dynamics of the model, as we will now explain.

The task instruction is reflected by connections between task codes and feature codes. These connections are bidirectional. As a consequence, activating a feature code will activate each connected task code, which on its turn will activate or enhance all connected feature codes, including the feature code that activated the task code in the first place (i.e., recurrent connectivity). This means that the mere fact of being connected to a task code will further enhance the activation of a feature code. For the ski instruction group, this means that 'left' and 'right' feature codes receive this enhancement, for the snowboard group this is the case for the 'forward' and 'backward' feature codes.

Crucially, this selective enhancement is already at play during the learning trials. When a motor code is activated during a learning trial, and its effects are presented to the model, the mere connections between feature codes and task codes will enhance either the 'Left' and 'Right' feature codes (in the ski condition) or the 'Forward' and 'Backward' feature codes (in the snowboard condition) and thereby determine the coding of the ambiguous sensation. When the action effect produced by 'M_1' is presented (i.e., activating the 'L/F' proprioceptive code) this results in a slightly higher activation for the 'Left' feature code in the ski condition and a slightly higher activation for the 'Forward' feature code in the snowboard condition, as shown in Figure 29 panels A vs. B. When the action effect produced by 'M_2' is presented (i.e., activating the 'R/B' proprioceptive code), this works in similar fashion.

During the 20 learning trials, this minimal difference in feature code activation results in pronounced differences in the weights learned (see Figure 29, panel C vs. panel D) and prepares the model for the experimental trials. Note that in the ski condition, the weights between the 'Left'/'Right' feature codes and motor codes are strong and the weights between the 'Forward'/'Backward' feature codes and motor codes are rather moderate (Figure 29, panel C). This is due to both the connections between the task codes and the 'Left'/'Right' feature codes and the stronger connections between sensory codes and the 'Left'/ 'Right' feature codes (as compared to the connections between sensory codes and the 'Forward' / 'Backward' feature codes). In the snowboard condition, the weights between the 'Left' / 'Right' feature codes and the motor codes are roughly equally strong as the weights between the 'Forward' / 'Backward' feature codes and the motor codes (Figure 29, panel D). This is due to the 'Forward'/ 'Backward' feature codes being connected to the task codes, resulting in top down enhancement of these feature codes. At the same time, the 'Left' / 'Right' feature codes receive more excitatory input due to their stronger connections with the sensory codes.

During the subsequent experimental trials, the model is set to respond to stimulus color and automatically takes stimulus direction into account (stimulus–response compatibility, SRC). This is a result from the fact that the model codes for responses and stimuli using common spatial feature codes. In the ski condition, the feature codes 'Left' and 'Right' are used to encode the responses. When perceiving a horizontal arrow stimulus, however, 'Left' and 'Right' are also used to encode this stimulus. When a congruent stimulus is presented, the corresponding feature code is already activated to encode this stimulus and therefore speeds up the encoding of the response. When an incongruent stimulus is shown, the wrong feature code is activated which slows down the activation – by means of lateral inhibition – of the correct response feature. This results in longer reaction times for incongruent than for congruent stimuli.

Now, the overlap between feature codes of stimulus and response obviously depends on the spatial coding of the response. As a result of task instruction and subsequent action–effect learning, this is different for the ski group and snowboard group. We now describe in detail the dynamics of the model during the experimental trials in both ski and snowboard conditions and for each type of stimulus (left-right congruent and incongruent, forward-backward congruent and incongruent) as depicted in the panels of Figure 30.

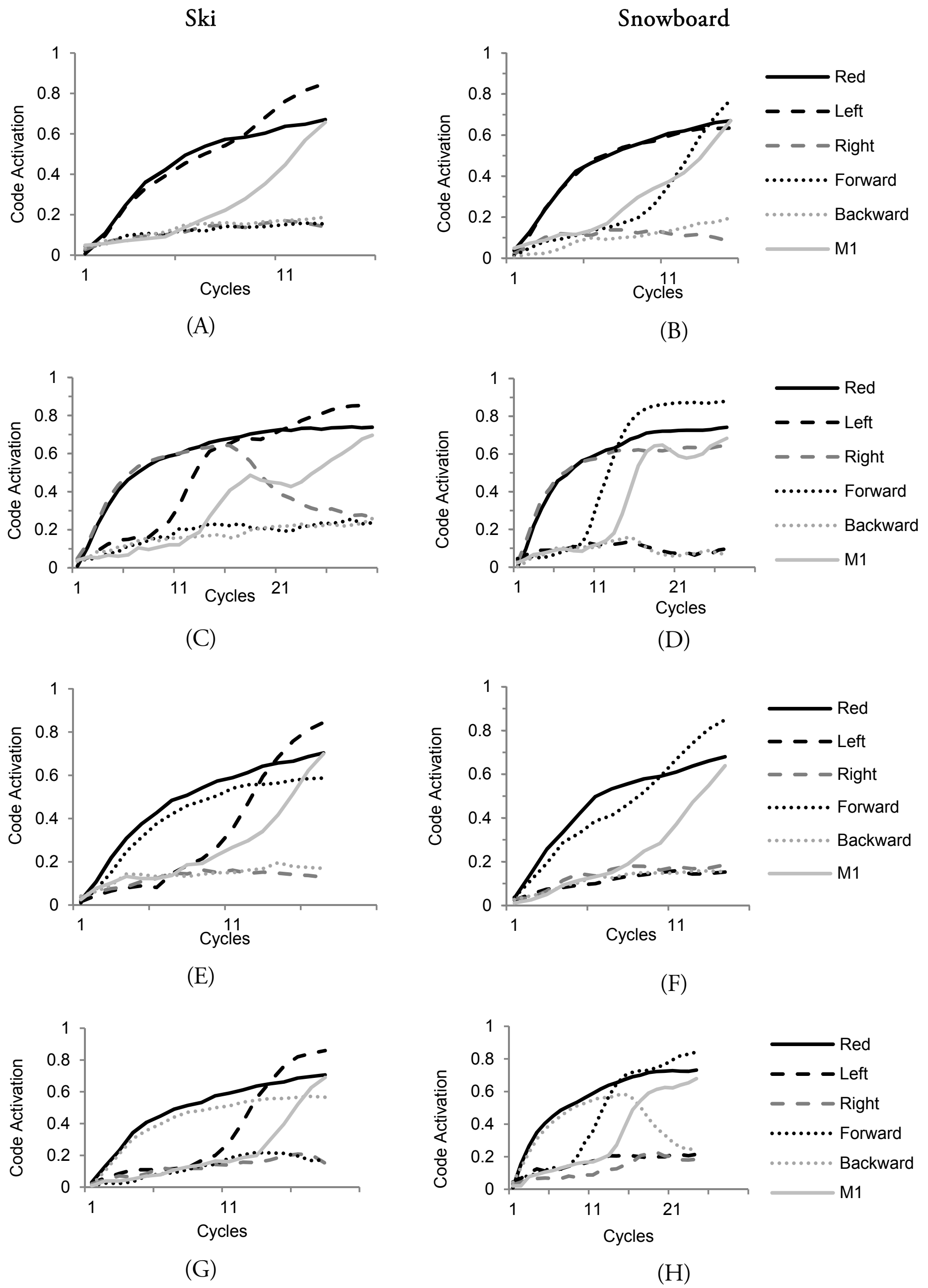


Figure 30. HiTEC simulation graphs of one simulated subject in the ski condition (panels A and C) and one simulated subject in the snowboard condition (panels B and D) during learning trials. Panels A and B show code activations resulting from the perception of the ambiguous action effect (balance towards 'left'/ 'forward'). Due to differences in task code – feature code wiring there is difference in recurrency and therefore slight differences in code activation ('left' vs. 'forward') in the two instruction conditions. In the panels C and D, that show the weight strength of a selection of feature code – motor code connections during all learning trials, it is clear that during the learning trials this difference in code activation accumulates to a substantial difference in the learned action–effect weights.

In panel A, a red left arrow stimulus is presented to the model in the ski condition, resulting in an initial increase of activation of 'Red' and 'Left' feature codes. In line with the ski task set, activation propagates from 'Red' to a task code and to the 'Left' feature code. This overlap results in a fast increase of activation of the 'Left' feature code. In the Ski condition the 'Left' feature code is strongly connected to 'M_1', resulting in fast activation propagation towards motor code 'M_1' and fast action selection. This explains the relatively shorter reaction times for the left-right congruent trials in the ski condition.

In panel B, a red left arrow stimulus is presented to the model in the snowboard condition, resulting in an initial increase of activation of 'Red' and 'Left' feature codes. In line with the snowboard task set, activation propagates from 'Red' to a task code and to the 'Forward' feature code; hence the subsequent increase in activation of the 'Forward' feature code. In the snowboard condition, 'Left', 'Right' *and* 'Forward' and 'Backward' feature codes are strongly connected to the motor codes (as depicted in Figure 29, panel B). Thus, both 'Left' and 'Forward' now propagate activation toward motor code 'M_1' resulting in fast action selection. This explains the relatively shorter reaction times for the left-right congruent stimulus trials in the snowboard condition.

In panel C, a red right arrow is presented to the model in the ski condition, resulting in initial increase of activation of 'Red' and 'Right' feature codes. In line with the ski task set, activation propagates from 'Red' to a task code and to the 'Left' feature code; hence the subsequent increase in activation of the 'Left' feature code. Now, both 'Left' and 'Right' feature codes are active and highly competing. They are both strongly connected to different motor codes that both receive activation and also compete with each other. This competition takes time and lengthens the trial.

In panel D, a red right arrow is presented to the model in the snowboard condition, resulting in initial increase of activation of 'Red' and 'Right' feature codes. In line with the snowboard task set, activation propagates from 'Red' to a task code and to the 'Forward' feature code, hence the subsequent increase in activation of the 'Forward' feature code. Now, the 'Forward' feature code is strongly connected to the 'M_1' motor code, the motor code to be selected. The 'Right' feature code, however, is (even more) strongly connected to the 'M_2' motor code. As both 'Forward' and 'Right' feature codes are highly activated and propagate activation to both motor codes, it takes longer for the system to settle this competition. This explains the relatively longer reaction times for the left-right incongruent stimulus trials in the snowboard condition.

In panel E, a red forward arrow is presented to the model in the ski condition, resulting in an initial increase of activation of 'Red' and 'Forward' feature codes. In line with the ski task set, activation propagates from 'Red' to a task code and to the 'Left' feature code; hence the subsequent increase in activation of the 'Left' feature code. Now, in the ski condition the 'Left' feature code is strongly connected to the 'M_1' motor code, the motor code to be selected. The 'Forward' feature code, however, is very weakly connected to the 'M_1' motor code. Thus the activation mainly propagates from the 'Left' feature code towards the 'M_1' motor code resulting in a speedy selection of the 'M_1' motor code, whereas the activation of the 'Forward' feature code has minimal influence. This explains the unaffected reaction times for the forward-backward congruent stimulus trials in the snowboard condition.

In panel F, a red forward arrow is presented to the model in the snowboard condition, resulting in an initial increase of activation of 'Red' and 'Forward' feature codes. In line with the snowboard task set, activation propagates from 'Red' to a task code and to the 'Forward' feature code. This overlap results in fast increase of 'Forward' feature code activation. In the snowboard condition the 'Forward' feature code is strongly connected to 'M_1', resulting in fast activation propagation towards 'M_1' and fast action selection. This explains the relatively shorter reaction times for the forward-backward congruent trials in the snowboard condition.

In panel G, a red backward arrow is presented to the model in the ski condition, resulting in an initial increase of activation of 'Red' and 'Backward' feature codes. In line with the ski task set, activation propagates from 'Red' to a task code and to the 'Left' feature code, hence the subsequent increase in activation of the 'Left' feature code. Now, in the ski condition the 'Left' feature code is strongly connected to the 'M_1' motor code, the motor code to be selected. The 'Backward' feature code is connected to the 'M_2' motor code, introducing competition. However, in the ski condition this latter connection is very weak. Thus the activation mainly propagates from the 'Left' feature code towards the 'M_1' motor code resulting in a speedy selection of the 'M_1' motor code, whereas the activation of the 'Backward' feature code has minimal influence. This explains the unaffected reaction times for the forward-backward incongruent stimulus trials in the snowboard condition.

In panel H, a red backward arrow is presented to the model in the snowboard condition, resulting in an initial increase of activation of 'Red' and 'Backward' feature codes. In line with the snowboard task set, activation propagates from 'Red' to a task code and to the 'Forward' feature code. Now, both 'Forward' and 'Backward' feature codes are active and highly competing. They are both strongly connected to different motor codes that also compete. This competition takes time and lengthens the trial, explaining the relatively longer reaction times for the forward-backward incongruent stimulus trials in the snowboard condition.

Finally, the data from the empirical study and the results from the simulation both clearly show a stronger congruency effect for the left–right dimension than for the forward–backward dimension (see Figure 28, depicted effect sizes are listed in Tables 2 and 3). As mentioned in Section 3, the asymmetry in the empirical data is in line with the left–right prevalence effect found in other studies (e.g., Nicoletti & Umiltà, 1984; 1985; Nicoletti et al., 1988). In the current study, we hypothesize that the use of left and right feet — for both left–right and forward–backward responses — may have yielded this prevalence effect (cf. Hommel, 1996). In more general terms, it could be argued (Rubichi et al., 2005) that the right–left discrimination is overlearned and produces faster processing than discriminations on other dimensions. In the model, the left–right dimension was enhanced by strengthening the connections between the sensory codes and 'Left' and 'Right' feature codes as compared to 'Forward' and 'Backward' feature codes. This resulted in a left–right prevalence effect, similar to the effect found in the empirical data.

In sum, the stronger connections between sensory codes and the 'Left'/'Right' feature codes (as compared to the weaker connections between sensory codes and the 'Forward'/'Backward' feature codes) together with the differences in mere connectivity between feature codes and task codes — which results from different task instructions — yield a pattern of left-right and forward-backward SRC effects that is comparable to the findings from the empirical study.

Discussion

In line with the general 'intentional weighting' principle (Memelink & Hommel, 2013), HiTEC explicitly addresses how task instructions are implemented in terms of representations and connections, and how they affect subsequent processing. The model is initially as task ignorant as humans are, until it 'receives the task instruction'. A task instruction is implemented by connecting feature codes and task codes following the actual task rules in terms of stimulus features and response (i.e., action effect) features (illustrated in Figure 31a). Here, we hypothesize that feature codes can be accessed by means of verbal labels and that receiving a task instruction can activate these feature codes and connect them to generic task codes (i.e., some sort of internal simulation of the translation from stimulus features to response features; see Chapter 2 for a more elaborate discussion of task internalization). This allows the task instruction to be readily internalized as connections from feature codes to task codes to feature codes.

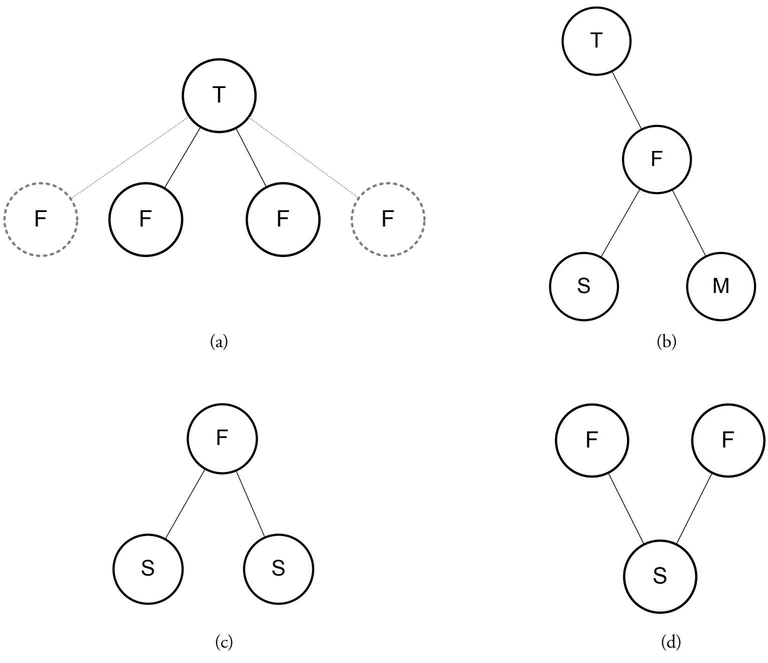


Figure 31. Schematic couplings between sensory codes, motor codes, feature codes and task codes.(a) a task set selectively relates task codes to feature codes representing stimulus or response features (b) feature codes representing responses in the task set may also relate to sensory codes (c) multiple sensory codes can relate to the same feature code (d) the same sensory code can relate to multiple feature codes

An example task instruction, taken from Simulation 5, "when you hear a high tone, press the left key" would then be implemented as connections from 'High' to 'T_1' and from 'T_1' to 'Left' and 'Key'. These connections subsequently enable the model to produce stimulus–response translations in accordance with task demands. Note that apart from this instruction based wiring we do not assume any other type of task-specific addition to the model (i.e., no additional 'task inputs' or biases in code dynamics).

These feature code–task code connections have two main consequences for subsequent processing: (1) they propagate activation from stimulus features towards response features that in turn excite motor codes and (2) they top down modulate lower level processing due to their recurrent nature. Arguably, this essentially constitutes 'attention': sensory codes that are connected to feature codes are enhanced, sensory codes that are not connected (i.e., stimulated by 'the other' elements in the scene) are not enhanced. Moreover relative higher activation of feature codes also results in relative stronger enhancement of sensory codes. This enhancement of lower level representations is crucial both in stimulus–response translation (i.e., responding to a stimulus is an integrated process) and during action–effect learning where it focuses attention on the relevant action effect features (see Chapter 3).

Moreover, as stimuli and responses are both defined in terms of common feature codes, it could happen that a response feature code included in the task set may be activated by a sensory code due to perceiving a (albeit task-irrelevant) stimulus feature (illustrated in Figure 31b). In that case the feature code would receive both exogenous excitation directly originating from a sensory code due to stimulus perception and endogenous excitation originating from response planning. Note that the latter form of excitation indirectly also originates from the stimulus but is mediated by the task set. As a result, response planning may be facilitated or hampered due to interaction between these pathways, yielding stimulus-response compatibility (SRC) effects (see Chapter 4).

Crucially, these effects depend on both stimulus and response coding. In HiTEC, responses are coded in terms of their perceptual effects. During the action-effect learning phase, associations are strengthened between motor codes and feature codes. The strengths of these associations depend on the co-activation of these motor codes and feature codes, which depends on both external stimulation (not explicitly modeled in current simulations, all sensory codes receive the same external input when excited) and top down modulation (i.e., connectivity between sensory codes, feature codes and task codes). Thus, as stated above, the task set not only determines actual stimulus-response translation (both controlled and automatic), it also influences how responses are coded (during action-effect learning) and thereby how (controlled and automatic) subsequent stimulus-response translation is carried out.

This influence of task instruction on stimulus-response translation is explicitly demonstrated in Simulation 5, where the instruction based connections automatically result in specific recurrency that selectively enhances either the 'key' or 'light' feature codes that in turn selectively enhance either the haptic location codes or the visual location codes when perceiving action effects. This leads to differences in action–effect weight learning and subsequently in how a response is encoded. These differences in response coding, in turn, influence the degree in which the feature codes representing stimuli and responses overlap, giving rise to different SRC effect directions across conditions.

Likewise, the empirical study presented in this chapter uses a (two-dimensional) Simon task with two groups of participants who only differ in the instruction (i.e., ski vs. snowboard) they received. Here, the presence and size of the Simon effect is also strongly dependent on the instruction: the left–right congruency effect is larger in the ski condition than in the snowboard condition, while the forward–backward effect only appears in the snowboard condition. Obviously, then, the task instruction moderates the internal translation process from stimulus to response.

Simulation 6, using the HiTEC model, shows how this result may emerge: task instruction is implemented as connections between feature codes and task codes, closely following the verbal instructions. This mere connectivity automatically results in specific recurrency that selectively enhances either the 'Left' vs. 'Right' or the 'Forward' vs. 'Backward' feature codes when perceiving (ambiguous) action effects. This leads to differences in action–effect weight learning and subsequently in how a response is encoded. These differences in response coding, in turn, influence the degree in which the feature codes representing stimuli and responses overlap, giving rise to different SRC effects across conditions.

Note that both the Hommel (1993) study and the empirical study presented in this thesis demonstrate influences of the task instruction on the reported SRC effects. These outcomes can be taken as evidence for the intentional weighting principle. However, the simulations of these studies show explicitly that this principle applies on multiple levels. In Simulation 5, different aspects of the action effect (i.e., light vs. key) contributed selectively to the *same feature dimension* (i.e., left–right; illustrated in Figure 31c) depending on the task instruction. Describing that task in terms of "*key pressing*" focused the (spatial) attention on the keys and increased the contribution of key location to the left–right dimension, whereas describing it in terms of "*light switching*" focused attention on the lights and increased the contribution of light location to the left–right dimension. Subsequently, the stimuli were encoded using this *same* left–right dimension. This resulted in either facilitation or interference yielding the observed stimulus–response compatibility effect. In Simulation 6, in contrast, a *single sensory dimension* (i.e., ambiguous proprioceptive balance) was assumed to map onto *two distinct feature dimensions* (i.e., left–right and forward–backward; illustrated in Figure 31d). Here, task instruction modulated the relative *weighting of these two feature dimensions* in the coding of the response. Subsequently, left vs. right directed stimuli were encoded using the left–right feature dimension and forward vs. backward directed stimuli were encoded using the forward–backward feature dimension. The relative weighting of these feature dimensions — modulated by task instruction — determined the relative sizes of the left–right SRC effects and forward–backward SRC effects, as observed in both the empirical data and simulation results. To conclude, the results of the empirical study presented in this thesis and the dynamics demonstrated in Simulation 6 together suggest that intentional weighting is not limited to weighting sensory dimensions, as demonstrated by Hommel (1993) and simulated in Simulation 5, but also extends to weighting *abstract feature dimensions*.

Summarizing, processing a task instruction is assumed to activate feature codes which are grounded in sensorimotor experience. Implementing a – in principle abstract – task set automatically wires these feature codes into a stimulus–to–response processing pathway using task codes. The fact that these feature codes also represent (prior) sensorimotor experience (i.e., by virtue of their connections to sensory codes) allows the task instruction to modulate subsequent sensorimotor processing (i.e., by top–down enhancing feature codes and therefore sensory codes), even on the automatic level of stimulus–response compatibility.

In a related study, Memelink and Hommel (2005) assessed the effects of both dimension priming and task instruction. Participants were presented stimuli that varied on both horizontal and vertical dimensions and performed actions that were also defined on both horizontal and vertical dimension. Prior to this task, participants performed an unrelated task for which only one of the spatial dimensions was relevant. In addition, instructions were varied by describing the responses in spatial terms or in terms of non-spatial features or response keys. The results showed that priming one of the dimensions increased the Simon effect on that dimension. Instructions, however, did not have any effect. These findings suggest that drawing attention to a specific dimension results in a stronger contribution of those features in response coding, which is fully compatible with HiTEC. At the same time, these findings show that mere task instruction does *not* affect action coding. Presumably, participants tend to recode the variations in these dimensions into a single intuitive dimension; hence, the influence of task instruction will then disappear. In the empirical study presented in this chapter, we have attempted to prevent this recoding by (1) use rather ambiguous balance responses and (2) reconfirming the response dimensions ("left"-"right" or "forward"-"backward" depending on the condition) in the recurring task instructions.

As in Hommel (1993) and the empirical study presented in this chapter, Yamaguchi and Proctor (2011) also found that the SRC effect depends on the attentional demands of the task. In their study participants controlled a simulated aircraft. A response yielded action effects on multiple dimensions: movement of the aircraft, movement of the horizon and the physical joystick movement. In this study, SRC effects depended on whether the (visual) emphasis was on the orientation of the aircraft (i.e., aircraft tilt, fixed horizon) or of the horizon (i.e., fixed aircraft, horizon tilt), which resonates well with our findings. In similar vein, Santiago, Ouellet, Román, and Valenzuela (2012) showed that conceptual congruency effects only appeared when participants attended to the relevant conceptual dimension, either through task instruction or by means of exogenous attentional cueing.

Note that these studies are based on the main assumption that in experimental settings human participants generally only respond to particular stimuli with particular responses because they are instructed to do so. Indeed, it has been shown that the stimulus-induced response activation that underlies SRC effects can only be obtained after the participant has implemented the required task set (Valle-Inclán & Redondo, 1998). In this study, the stimulus-response mappings of the task (i.e., the task instruction) varied randomly from trial to trial. In some trials, the mapping instruction was presented followed by the target stimulus.

In other trials, the stimulus preceded the mapping instruction. Their results showed that the Simon effect was only observed in trials where the mapping instruction was presented *before* the stimulus. This suggests the task set must be implemented before SRC can occur.

Thus, understanding the task demands somehow configures the cognitive system to modulate both stimulus perception and response planning. This involves attending to task-relevant stimulus features (e.g., a high or low auditory pitch) and preparing a small selection of motor schemas (e.g., pressing keys). In more general terms this process of configuring the cognitive system is what we consider the main contribution of cognitive control; it prepares the system to subsequently act according to instruction—it in a sense turns the system into a 'prepared reflex' (Hommel, 2000a; see also Bargh, 1989). Note that instruction wiring by itself ensures attention for the right dimension(s), for example key locations vs. light locations in Simulation 5. Indeed, as cognitive control is implemented as mere connectivity resulting from task instructions, there is – at least with respect to the simulated experiments – no need additional online control of the inner mechanisms.

Addressing the apparent crucial role of task goals in SRC, Ansorge and Wühr (2004) formulated the *response-discrimination hypothesis* that states that response representations are not automatically formed, but rather top-down controlled. Only spatial features that discriminate between alternative responses are represented and thus give rise to a Simon effect. This resonates well with the conclusions in a general review by Proctor and Vu (2006) that the Simon effect is not resulting from an automatic activation of a corresponding response by means of a hard-wired (e.g., Kornblum et al., 1990) or over-learned (e.g., Umiltà & Zorzi, 1997) route; rather the task defines S-R associations that mediate this responding. HiTEC is clearly consistent with this response-discrimination hypothesis and provides a rationale, in terms of internalization of an explicit task set using codes that are grounded in sensorimotor experience, for *how* and *why* these response representations are formed and top down modulated.

In general, existing models of SRC do not address internalizing task instructions or context explicitly. Stimulus codes and response codes are connected using two routes (e.g., Kornblum et al., 1990; Zorzi & Umiltà, 1995; De Jong, et al., 1994). A direct route connects the spatial stimulus codes to the corresponding spatial response codes, which is assumed to reflect an automatic process. The task instruction (e.g., "*when you hear a high tone, press the left key*") is implemented as a connection from the non-spatial stimulus code (e.g., 'high tone') to a spatial response code (e.g., 'left key'), following the task instruction. This is assumed to reflect a controlled process. Now, when a compatible stimulus is presented (e.g., a high tone presented on the left), both the direct connections and the controlled process connections contribute to a speedy activation of the correct response code. Conversely, when an incompatible stimulus is presented (e.g., a high tone presented on the right), the direct route activates the incorrect response. The controlled route, however, activates the response determined by the task instruction, which eventually wins the competition. As a result, processing incompatible stimuli results in longer reaction times than processing compatible stimuli. In sum, the stimulus–response compatibility effect arises from the interplay between the direct route, reflecting an automatic comparison between spatial stimulus and response codes, and the controlled route, reflecting the task instructions (see Chapter 4 for a detailed analysis of automaticity in these dual route models). Crucially, these models do not address how the direct route also depends on the task context, a result that is evidently demonstrated in the empirical study in this chapter as well as in the related work described above.

Moreover, existing dual process models have been built to simulate particular tasks, so that task instructions were hard-coded into the model. Although HiTEC also contains codes that are specific to the task to be simulated, the actual task instruction and practice trials prepare the model for the experimental trials. This allows a HiTEC model instance to take task instruction into account and is therefore able to flexibly interpret the same motor action as either 'left' or 'right' depending on this task instruction, and thereby account for the inversion of the Simon effect (example taken from Simulation 5). A crucial difference between dual process models and HiTEC, in this respect, is that HiTEC contains recurrent connections, allowing task codes to modulate both stimulus and response coding; whereas dual process models work strictly feedforward, requiring the modeler to fully design the resulting process in advance. This does not allow for 'run-time' differences between instruction conditions.

Interestingly, studies in neuroscience have demonstrated that the *same neural circuitry* is used for decision making in different tasks (e.g., Duncan & Owen, 2000). Presumably, by hearing and internalizing the task instruction, this circuitry can be tuned to the task at hand (i.e., implementing a *task set*, Monsell, 1996). In a similar vein, in HiTEC, the same perception–action model with the same generic structure, representations and processing principles enables the simulation of multiple experimental tasks when tuned by different task instructions.

The task modulated interaction between perception and action may also be related to psychologically meaningful interactions between levels: (1) activation of lower level codes is modulated by higher levels through their connections, effectively realizing what could be called 'attention' (i.e., task codes that by their connections enhance feature codes that enhance sensory codes; see particularly Simulation 5); and (2) competition at the sensory level supports competition at the feature level (i.e., their relative activation levels are continuously propagated) and at the task level; and vice-versa: competition between task-codes influences the selective enhancement of lower level codes (i.e., feature codes and thereby sensory codes) and thereby biasing their lateral competition.

HiTEC explicitly models the influence of task context on stimulus-to-response translation. It can be argued that different task instructions direct the model's 'attention' to different stimulus features (or response features). Cohen et al. (1990) have presented an influential PDP model of attentional control for the Stroop task. In their model, as in the aforementioned dual process models, activation propagates from stimulus codes towards response codes. In addition, this model contains two task codes. Depending on the task at hand, one of the task codes is also activated and biases processing, thus realizing task modulated processing from perception to action, as in HiTEC. In fact, HiTEC can be seen as an extension of the approach of Cohen et al. in that it considers how task representations modulate stimulus-response translation in a broader range of tasks and stimulus and response sets. In particular, the intentional-weighting principle built into TEC and HiTEC is perfectly consistent with the suggestion of Cohen et al. to explain task-specific modulation through the priming of task-relevant representations and pathways. At the same time, HiTEC goes beyond the Cohen et al. model by explaining where the respective representations are coming from and by accounting for a broader range of phenomena.

In an attempt to account for the role of task context Yamaguchi and Proctor (2012) propose a multidimensional vector model of SRC. This mathematical model addresses the issue of task context in the Simon task by means of an S-R vector space. In this model, stimulus features and response features are treated in similar fashion, which is in line with the HiTEC model. HiTEC, however, stresses biological plausibility by means of codes with activation dynamics that approximate biological neuron population, recurrent connections and within-layer lateral inhibition.

In the Ward model (Ward, 1999; see Chapter 2 for a more detailed comparison of is architecture with HiTEC), that also contains recurrent connections, units are selectively activated based on the task at hand. For example activation of 'red' code biases interactive processing such that the model enhances representations of 'red' stimuli. In contrast to HiTEC, however, the Ward model does not provide any means to internalize a number of translation 'rules' that effectively make up a forced choice paradigm. Also, in other models such as SLAM (Phaf et al., 1990) and the attentional response model by Cohen and Shoup (1997) – see Chapter 4 for a more elaborate comparison with HiTEC –, the task is hardwired as connections between stimulus feature codes and – through intermediate codes – response codes. Despite their recurrent connections, in contrast to the dual process models mentioned above, these models do not provide the means to internalize a task or explain SRC beyond the assumption that 'special links' exist between some stimulus - responses combinations and not between others (see Chapter 4 for a more detailed discussion).

Summarizing, existing models of perception and action typically do not allow for a straightforward internalization of a forced choice response task into a task set. Moreover, empirical evidence suggests and our simulations demonstrate that such a task set has critical influences on SRC effects. As most models of SRC do not explicitly address the task context they cannot account for these influences and thereby lack the ability to provide a rationale for the SRC effects to attempt to model.

To conclude, adaptive behavior is defined within a specific task context. Empirical evidence suggests and our simulations demonstrate that control is exerted by having the task instruction implemented as a task set that results both in strong task-driven translation pathways — so that the stimulus–response translation is controlled in distal terms — and in specific code overlap — so that automaticity effectively gets to fill in the details.

Chapter 6
General Discussion

This chapter is an integration and extension of major parts of the following articles:

Haazebroek, P., Raffone, A., & Hommel, B. *HiTEC: A Connectionist Model of the Interaction between Perception and Action Planning*. Manuscript submitted for publication.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2013). How task goals mediate the interplay between perception and action. *Frontiers in Psychology, 4:247*.

In contrast to traditional views of human information processing (e.g., Donders, 1868; Sternberg, 1969), abundant empirical evidence (e.g., Elsner & Hommel, 2001; Müsseler & Hommel, 1997; Simon & Rudell, 1967; Stroop, 1935; Stoet & Hommel, 2002) suggests that perception and action planning do not represent separable stages of a unidirectional processing sequence, but rather emerging properties of highly interactive processes. To capture these interactive characteristics of the human cognitive system, we have developed a connectionist model of the interaction between perception and action planning: HiTEC, based on the Theory of Event Coding (TEC, Hommel et al., 2001). The model is characterized by representations at multiple levels and by shared representations and processes. HiTEC extends and further specifies TEC's principles to account for a series of key experimental findings in a unitary theoretical framework and at a level of specificity that allows for computer simulation.

Research questions revisited

In order to gage the merits of the HiTEC model, we have addressed a number of specific research questions in the previous chapters.

How do neuron-like representations realize stimulus-response translation?

In HiTEC, the connectionist (Rumelhart et al., 1986) model presented in Chapter 2, neuron-like representations are *distributed* over *multiple levels* and processing involves both feedforward and *feedback* interaction between lower and higher level representations, in line with global cortical layering and connectivity (cf., Braitenberg & Schütz, 1991; Prinz, 2006). Through this bi-directional interaction, lower level representations code for sensory features (in line with DeYou & Van Essen, 1988) or motor responses, and higher levels modulate and coordinate their interaction in accord with the task context implemented as task set (in line with Duncan & Owen, 2000). In addition, in the spirit of neural circuitries that appear to code for both perception and action (e.g., mirror neurons in the premotor cortex, Keysers & Perrett, 2004; Rizzolati & Craighero, 2004), HiTEC contains common code representations (Prinz, 1990) that are used both for stimulus perception and response planning. These codes ensure that stimulus-response translation may also take features into account that are only implicitly specified in the task set.

Stimulus-response translation is initiated by presenting a stimulus. In HiTEC this is done by feeding external input to sensory codes. Responses are considered to execute when a motor code reaches the activation threshold. The connection between perception and action is realized by representations at multiple levels and interconnected by feedforward and feedback connections. The result is an interactive processing network that translates stimuli in responses by gradually propagating activation trough units in the model. Rather than a sequential stepwise process from sensory codes through intermediate representations to response codes, all representations at all levels cooperate and compete and together converge to a response outcome (in line with Duncan et al, 1997). Crucially, representations at higher levels modulate representations at lower levels. This allows both for direct interaction between perception and action representations and their modulation by the task context.

How do situation-specific meanings of motor actions emerge?

In order to control its actions in response to demands in the environment the cognitive system needs to know what actions are possible and what these actions 'mean'. Various empirical findings suggest that for a cognitive system this 'meaning' is not a fixed fact; it rather depends on the (perceptual) effects within the task context. Consequently, in order to select and execute an appropriate response to a stimulus a plausible cognitive model must first learn (i.e., from experience) what the effects of its motor actions are and how to interpret these effects in the task context.

Action control in HiTEC, discussed in Chapter 3, is based on the ideomotor principle (James, 1890; Lotze, 1852) which stresses both the acquisition of action-effect associations and the use of these associations in action planning (Hommel, 2009). Simulation 1 addressed how novel action-contingent perceivable effects are (spontaneously) associated to the motor actions that yield these effects (Elsner & Hommel, 2001). In HiTEC simulations, this is done during a phase of action-effect learning. Simulation 2 of a study by (Kunde et al., 2004) demonstrated how the (internal) consistency of these effects influences the representations of these effects. As action-effect learning depends on the activation of both motor codes and feature codes, the consistency of feature code activation has consequences for the resulting association strengths. And because these associations have a crucial role in planning actions in response to stimuli, subsequent stimulus-response translation is influenced by the strengths of these associations. As a result, responding to stimuli takes the contextual meaning (e.g., consistency among action effect features) of motor actions into account as it is represented in the acquired action-effect associations. Moreover, the strengths of these associations depend on the co-activation of motor codes and feature codes during learning. This co-activation not only depends on external stimulation but also on top down modulation (i.e., due to connectivity between sensory codes, feature codes and task codes). As a result, action-effect learning, and thus the 'meaning' of action, depends on the task set. This is the case in all simulations, including those in Chapter 3, but explicitly modeled in Simulations 5 (of Hommel, 1993) and 6 in Chapter 5. In these simulations, the task instruction was explicitly manipulated resulting in differences in task sets, and, therefore, differences in top down modulation of feature and sensory codes, also during action-effect learning. This resulted in different representations of the same motor actions and consequently in the emergence of situation-specific meaning of action.

How and why do parts of stimulus–response translation occur automatically?

Some parts of the translation from stimulus to response are considered to occur automatically as demonstrated by stimulus–response compatibility (SRC) effects (e.g., Hommel, 1993; MacLoad, 1991; Simon & Rudell, 1976). How and why these effects may occur is addressed in Chapter 4. As demonstrated in the simulations in this chapter, HiTEC is able to account for these effects. In fact, SRC is an *inevitable* consequence of HiTEC's structures and processing characteristics. First, in order to internalize task instructions into a task set, both stimuli and responses need to be represented on a distal level and associated through task codes. Secondly, actions are represented in terms of perceptual effects and therefore use the same distal codes as stimuli and, consequently, are grounded in the same perceptual world (Prinz, 1992). This means that code overlap is possible and – to the extent that stimuli and responses overlap in the external environment, such as spatial correspondence — very probable. Finally, HiTEC assumes integrated processing (cf. Duncan et al., 1997) which means that stimulus coding and response coding also overlap in time. Thus, the task set results in a pathway mediated by task codes and defined in terms of distal features, and in probable code overlap of these same distal features; as stimulus processing and response planning occur simultaneously, the cognitive system inevitably needs to combine task-driven and automatic feature code activation. As a result, code overlap between stimulus and response features results in either facilitation or interference effects.

How does the task context modulate stimulus-response translation?

How the task context may modulate stimulus-response translation is explicitly addressed in Chapter 5. In HiTEC, processing a task instruction is assumed to activate feature codes which are grounded in sensorimotor experience (Hommel et al., 2001). Implementing a – in principle abstract – task set wires these feature codes into a stimulus–to–response processing pathway using task codes. The fact that these feature codes also represent (prior) sensorimotor experience (i.e., by virtue of their connections to sensory codes) allows the task instruction to modulate subsequent sensorimotor processing (i.e., by top–down enhancing feature codes and therefore sensory codes), even on the automatic level of stimulus–response compatibility. Note that we constrain task internalization to setting connections between grounded feature codes and generic task codes only and do not allow additional codes or inputs to influence stimulus-response translation. The first simulation in this chapter, Simulation 5 of a study by (Hommel, 1993) demonstrated how the task context may modulate action control by means of (spatial) attention within the environment; this is realized by virtue of enhancement of either visual or haptic sensory dimensions projecting to a shared feature dimension; in similar vein, the empirical study presented in this chapter and Simulation 6 together showed how cognitive labeling of an ambiguous action may also influence response coding by modulating the enhancement of one or the other feature dimension. Together these simulations suggest that a task set in terms of connections is sufficient to top down modulate sensorimotor processing and that the theoretical notion of intentional weighting (Memelink & Hommel, 2013) may indeed operate on multiple levels of representation.

Key characteristics of HiTEC

The HiTEC connectionist model (described in more detail in Chapter 2) features a number of key characteristics. We now discuss each of these characteristics in more detail.

Multi-level architecture

In line with neuroscientific findings (e.g., DeYou & Van Essen, 1988) HiTEC is architected as a multi-level model, with lower level representations relating to sensory input and motor output and higher levels referring to generic cognitive control (cf. Duncan & Owen, 2000). Presumably the brain is architected using many more levels of representation, but HiTEC simulations of various empirical findings suggest that it is necessary to dissociate a *minimum of three levels*[8] *of representation*. These dissociations are illustrated in Figure 32. Panel A of the figure depicts the dissociation between feature codes and sensory codes. Although many feature codes in the HiTEC model instances presented in this thesis seem to merely duplicate their sensory code counterparts (e.g., 'high' pitch feature code and 'high' pitch sensory code in Simulation 1, illustrated in Figure 8), in some cases multiple sensory codes may project on shared feature codes, such as in Simulation 2, illustrated in Figure 12. Here multiple sensory dimensions, i.e., auditory *and* haptic intensities, are assumed to relate to a shared feature dimension. Indeed, sensory codes register proximal sensation, whereas distal feature codes represent cognitive interpretation of a (possibly multi-modal and even multi-sense) sensation—following the distinction introduced and elaborated by Heider (1926/1959, 1930/1959; see Hommel, 2009). Activating sensory codes means actually getting external

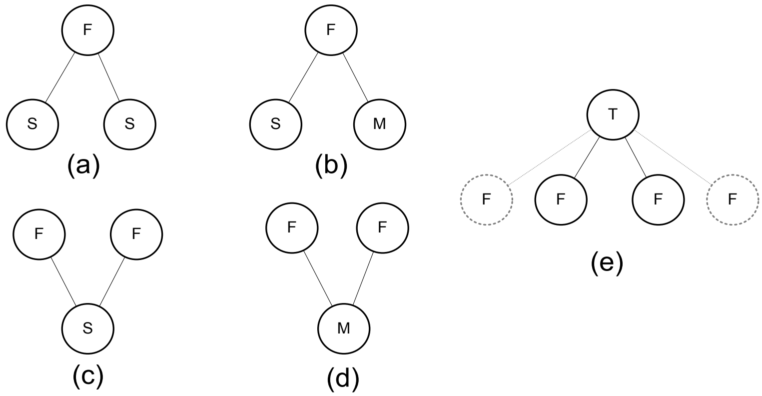


Figure 32. Levels of representation in HiTEC: (a) multiple sensory codes may relate to the same feature code, (b) feature codes relate to both sensory codes and motor codes, (c) sensory codes may relate to multiple feature codes, (d) motor codes may relate to multiple feature codes, and, (e) task codes selectively relate to feature codes

[8] Note that although we consider sensory codes and motor codes to be distinct systems we regard them similar in their level of representation. Hence task level, feature level and sensory / motor levels constitute three different levels of representation.

input from the senses, activating feature codes could reflect imagination (e.g., when following a task instruction), expectation and/or planning. Sensory codes are labeled for the reader's convenience but they reflect sensory input that is generally too specific — i.e., precise color hue, auditory pitch or exact haptic location — to retain in memory or to imagine, feature codes are considered(grounded) cognitive features and are available for planning, imagination et cetera.

Moreover, feature codes are assumed to relate to sensory codes and motor codes due to ideomotor learning. This relation may depend on various factors (e.g., consistency in sensory stimulation; top down modulation by the task set) as demonstrated in for example Simulation 2 and Simulation 5. This suggests dissociation between feature codes and motor codes as depicted in Figure 32b (Hommel et al., 2001). Also, the same sensory code may relate to multiple feature codes, such as in Simulation 6, illustrated in Figure 27. Here an ambiguous sensory sensation were found to relate to both left/right and forward/backward feature dimensions as suggested by the empirical study and demonstrated in the simulation. In similar vein, the same motor code may relate to multiple feature codes, as was the case in most simulations (for instance Simulation 1; here, motor codes were associated to both left-right location and high/low pitch features). Finally, the task set selectively relates stimulus features to response features resulting in top down modulation of stimulus perception and response planning, and their interaction. This is illustrated in Figure 32e. Crucially, this task set is implemented in terms of distal feature codes, presumably by means of verbal labels (Hommel & Elsner, 2009) rather than in sensory or motor codes. Indeed, the task instruction is likely to operate in terms of cognitive interpretations of sensory stimulation. For instance, the instruction "when you hear a high tone, press the left key" cannot relate to a specific sensory code, as it is, at the moment of task instruction, unclear what the actual pitch of the "high" tone would be.

Thus, together, these requirements result in *a minimum of three levels* of representation necessary to translate stimuli to responses according to a task context within the key paradigms selected for simulation: generic task codes control different task-directed translation pathways by selectively wiring to a-modal feature codes; these feature codes are activated both intentionally and automatically; sensory codes are activated by external input and are top down modulated by cognitively anticipating feature codes connected at the distal feature level.

Ideomotor learning

Another key characteristic of HiTEC, which it inherits from TEC, is ideomotor learning. Although this mechanism is currently implemented as an explicit phase preceding stimulus-response translation and modulated by prior set task instructions, it could be designed as to be part of an error-correction routine based on action anticipation (see Chapter 3). Importantly, in HiTEC, ideomotor learning allows representations that relate perception and action to be grounded in sensorimotor experience. This in turn, makes possible the emergence of situation-specific meaning of actions. Although many models and artificial cognitive systems are employed with predefined actions, it would seem that any autonomous cognitive system would need to learn from such experience. Naturally, the question arises as to what level these action effects should be represented. In HiTEC we have opted for a 'distal level' of representation, rather than directly associating motor output to sensory input. This allows for some variation in (presumably more noisy and less accurate) sensory stimulation, for combining information from multiple senses and modalities, for sensitivity to exogenous saliency and, importantly, for top down modulation originating from the current (task) context. The various empirical studies simulated in this thesis show that the meaning of action depends on these factors and the simulations show that ideomotor learning provides the means to cope with exactly that.

Common codes

Central to HiTEC and TEC alike, is the assumption of common codes (Prinz, 1990). In addition to various findings in neuroscience that strengthen the plausibility of such representations (Keysers & Perrett, 2004; Rizzolati & Craighero, 2004), common codes constitute representations with an intuitive grounding in regularities in sensorimotor experience. Implementing a task set in terms of these common feature codes further allows the cognitive system to control its stimulus-response processing while allowing stimulus features and response features that overlap to be translated implicitly. This implicit translation explains the facilitation and impairment of performance observed in experimental SRC paradigms. In 'real life', however, it allows the cognitive system to control its behavior using minimal task set specifications and leave the details to automatic mechanisms (Hommel, 2009). The idea of this implicit translation resonates well with the notion of *affordance*. Gibson (1979) defined affordances as the action possibilities that an object offers to a perceiver/actor. He further proposed that these action possibilities can be directly perceived (see also Neisser, 1992), that is without mediating (symbolic) representations. Hence, action systems can readily make use of this information. On a similar note, rather than setting out to represent every minute detail of the outside world (and then reason about it using hard computation) the world itself can serve as "its own best model" (Brooks, 1991, p. 139). Indeed, the outside world is always up to date and contains every detail there is to be known. The use of common codes representing both object features and action features on a distal level may enable lower level closed-loop mechanisms to sample the world while acting. Such mechanisms may eliminate the need for higher level representations for detailing out specifics of action control (as more elaborately discussed in Chapter 3; see also Hommel, 2009; c.f., findings by Prablanc & Pélisson, 1990).

One could argue, however, that the common feature codes in HiTEC need not be actually common per se. For example, Oriet, Stevanovski and Joliceur (2001) claim that a two-way interaction between separate perceptual features and action features would suffice to yield similar results. This, however, would require additional, and in fact duplicated, codes (e.g., instead of a common 'left' feature code, the model would contain two interconnected (sub)codes: a 'left' perceptual feature code and a 'left' action feature code) as well as additional computational mechanisms without clear computational benefits or theoretical necessity. In contrast, common codes provide a parsimonious solution and a clear concept of how actions are grounded in the perceptual world (Hommel, 2009; 2013).

Interactive processing

Processing in HiTEC is interactive: stimulus-response translation is carried out by including all related codes in a global competition mechanism (cf. Duncan et al., 1997). However, one could question the usefulness of such interactive processing as some of the existing SRC models do not seem to need this and employ strictly feedforward processing (and in the case of Kornblum et al., 1990 even divide processing into perceptual and response stages). Indeed, one could argue that at some point (1) sensory codes need only to register stimulus input, that (2) motor codes need to be activate in order to execute a response, and, thus, (3) that activation needs only to propagate from these sensory codes to these motor codes without an apparent need for interactivity.

In this thesis, however, we argue that interactivity is not as much a direct requirement for such stimulus-response translation on its own, rather a logical consequence of constraints regarding crucial aspects that precede such translation: (1) not allowing additional task specific input biases during stimulus-response translation requires the task set to be internalized using connections between feature codes and task codes only. This means that modulation of feature codes (i.e., attending to one feature over the other, in accord with the task set) is to be done using these connections. This is realized by making these connections bi-directional. (2) feature codes are grounded by means of their connections to sensory codes. It is to be expected that these connections reflect long term experience and do not change in their weights that easily. However, short-term enhancement of specific sensory dimensions is necessary for situation-specific response coding as demonstrated in Simulation 5. In this simulation, without modulation of haptic vs. visual sensory inputs both would have influenced the left-right feature dimension equally, reducing the model's ability to internalize context-specific action-effect regularities, and, eliminating the Simon effect altogether. Bi-directional connections between feature codes and sensory codes realize such modulation as they allow feature codes to bias sensory code activation without affecting their grounding. And (3) codes within the same layer compete for activation as a consequence of lateral inhibition following PDP (e.g., Cohen et al., 1992). Together these aspects yield integrated competiton (cf. Duncan et al., 1997). Moreover, other empirical findings (e.g., Müsseler & Hommel, 1997; Stoet & Hommel, 2002) explicitly demonstrate direct influence of action perception on stimulus perception, requiring such interactivity. Finally, interactive processing seems very much in line with basic general neuroscientific findings (Braitenberg & Schütz, 1991; Prinz, 2006) that show vast numbers of recurrent connections in the brain.

Assuming direct interactivity between codes from different representational levels opposes serial stage theories and has consequences for what is often referred to as *modularity*. The strong modularity hypothesis (e.g., Fodor, 1983; Pinker, 1997) states that the cognitive system is composed of information encapsulating modules that receive input from only a few other modules and produce all-or-none output in discrete stages. In contrast, in PDP models (Rumelhart et al., 1986), such as HiTEC, representations are distributed and processing is continuous rather than being staged. Moreover, information within PDP components is available to other components rather than encapsulated. At the same time, as discussed in Chapter 2, we do aim at constraining HiTEC's representations and connectivity to basic observations in neuroscience: lower cortical area's code for specific sensory modalities and dimensions, mid-level areas combine information from lower levels and high-level areas are task-generic and involved in practically any decision task. Moreover, representations from different levels are connected not only with feedforward but also with vast amounts of feedback connections. By adhering to these constraints, it could be argued that the HiTEC model can indeed be decomposed in different structures/modules. Each of these modules may even be considered to represent specific aspects of perception, cognition or action. However, as processing in HiTEC is fully interactive it is misleading to assign such labels to HiTEC's structural components: indeed, perception emerges from interaction between all structures, and so do cognition and action. In similar vein, one could to try to discretize the global process in HiTEC; that is, divide the time course of stimulus-response translation into stages such as 'registering the stimulus', 'choosing between task mappings', 'planning the response' and map these stages onto the 'mainly active representations' (e.g., sensory codes and stimulus feature codes are mainly active right after stimulus presentation and would, in such discretization carry out the 'registration of the stimulus'). However, first, as clearly demonstrated in the various simulations in this thesis, these 'stages' would overlap in time; and, second, during each stage representations other levels (e.g., task level, motor level) are involved enhancing or suppressing activation in the structures that would be considered to carry out the respective stage; and even though one would regard their activation only as moderate, their influence on the activation levels of other representations (e.g., at the feature level) is actually be decisive at crucial moments of the task (e.g., top done enhancement of visual over haptic locations in the light condition in Simulation 5).

Extending HiTEC

Although we have demonstrated that the current version of HiTEC can simulate a variety of perception-action experiments, its principles, processes, and properties are still far from sufficient to account for all known phenomena in this domain. Most notably, HiTEC yet lacks the ability to bind features—an ability that was emphasized in the original TEC (Hommel et al., 2001). When an agent is presented with two visual objects, say one blue and one red, both 'Blue' and 'Red' sensory codes will be activated concurrently and the present model has no means to code or keep track which color belongs to which object (the classic 'binding problem': Treisman, 1996). Given that several empirical phenomena in the domain of perception-action interactions are likely to reflect feature-binding processes (Müsseler & Hommel, 1997; see Hommel, 2004, for an overview), extending HiTEC to include a binding mechanism seems essential. Closely related is the impact of episodic memory on perception and action planning (Waszak, Hommel, & Allport, 2003). Not only the current task set but also recent experiences with particular stimuli and actions under a particular task set can play a large role in the later interpretation of stimuli and responses and the efficiency with which their processing can be controlled (Nuxoll & Laird, 2004).

In addition, action planning in HiTEC is currently greatly simplified as we only allow one rather simple action to be planned at a time. A more realistic model of action control would include the planning, programming, and coordination of more complex, hierarchical actions (Logan & Crump, 2010) and action sequences (Tubau, Hommel, & López-Moliner, 2007). As stressed in Chapter 2, a particular strong addition would be the inclusion of conflict monitoring and performance feedback along the lines of (Botvinick et al., 2001). The experience of response conflict and/or of negative feedback might strengthen the activation state of goal codes and their impact on stimulus-response processing, which would tend to prevent errors in the future (van Steenbergen, Band, & Hommel, 2009).

Moreover, the current computational implementation of HiTEC as a connectionist model introduces notable simplifications as such a model may adhere to only a few basic constraints from neuroscience. Clearly the 100 billion neurons of the human brain allow for much more intricate interactions than HiTEC is able to model. For example, the current connectionist approach focuses on simple causal task mappings between stimuli and responses. Although this was sufficient for the paradigms simulated for this thesis, extending representational power for higher order logical reasoning may enable simulating more complex experiments.

In addition, in the current version of HiTEC we have focused on the interaction between perception and action and how this is mediated by the task context, both in terms of a direct pathway of activation propagation and in terms of connecting feature codes that relate to both stimuli and responses. In current simulations, however, feature codes are assumed have evolved from prior sensorimotor experience, presumably by detecting regularities in sensorimotor patterns. An obvious extension of our current work would be to investigate the actual grounding mechanism of feature codes by addressing how and when these regularities are detected and result in the creation of new (or possibly the adjustments of existing) feature codes. One option would be to employ self-organizing feature maps (e.g., Cos-Aguilera, Canamero, & Hayes, 2004; Kohonen, 1982) allowing to create and adjust inter-related clusters of possibly high variance, multidimensional sensory input. Along the lines we have suggested for action-effect learning, creating or adjusting such sensory feature networks may also be triggered by the detection of discrepancies between expectations and perception, and modulated by top down processes in similar spirit as action-effect learning and stimulus-response translation processes discussed in this thesis.

Finally, HiTEC is very much in line with theories of embodied cognition (e.g., Barsalou, 1999; Glenberg, 1997; Wilson, 2002). Numerous findings (e.g., Pecher & Zwaan, 2005) support the idea that perceptual and motor systems are involved in even abstract cognitive processing. This suggests that cognition interacts with perception and action and that these systems share the same representations and processes. Hence, as HiTEC explicitly models stimulus-response translation in a way consistent with these theories, it may prove quite interesting to actually *embody* HiTEC. That is, to endow an embodied cognitive system (i.e., a robot) with a HiTEC-like control architecture and put both theory and model to the test of real world interaction. Interestingly, in line with Brooks (1986), stimulus-response translation in HiTEC involves multiple interacting perception-action pathways at different levels. Moreover, ate each of these levels perception and action interact resulting in experience based action-effect learning, mid-level common features that may automatically overlap and high level wiring of task codes and feature codes implementing a cognitive task.

To conclude, I hope to have shed some light on the link between perception and action. Although HiTEC may seem to be a simple model considering its architecture, its recurrent connections yield interesting dynamics. The HiTEC model is intended as a proof of principle, showing how common representations may mediate the interaction between - what could be called - perception and action in a stage-less way, addressing issues of situation-specific action knowledge, selective automaticity and task demands, all relevant for effective behavior.

So, how do we interact with our environment? How do we grasp our cups, shift gears or press buttons on the TV remote? We do not seem to first perceive, then think and finally act. No, we are much smarter than that! We configure our perception-action system in advance, tuned to the task at hand. Then, we let perception and action interact as they see fit, and control this interaction on an abstract level only, saving scarce cognitive resources for other demanding mental activities, such as reading a thesis.

References

Ach, N. (1910). *Über den Willensakt und das Temperament*. Leipzig: Quelle & Meyer.

Anderson, J. R. (1992). Automaticity and the ACT theory. *The American Journal of Psychology, 105*, 165-180.

Anderson, J. R. (1993). *Rules of the mind*. Hillsdale, NJ: Erlbaum.

Ansorge, U., & Wühr, P. (2004). A response-discrimination account of the Simon effect. J*ournal of Experimental Psychology: Human Perception and Performance, 30*, 365–377.

Bargh J. A. (1989). Conditional automaticity: varieties of automatic influence in social perception and cognition. In: J.S. Uleman & J.A. Bargh (Eds.), *Unintended thought* (pp. 3–51). New York: Guilford Press.

Bargh, J. A., & Gollwitzer, P. (1994). Environmental control of goal-directed action: Automatic and strategic contingencies between situations and behavior. *Nebraska Symposium on Motivation, 41*, 71–124.

Barsalou, L. W. (1999). Perceptual symbol systems. *Behavior and Brain Sciences, 22*, 577–660.

Barsalou, L. W. (2008). Grounded cognition. *Annual Review of Psychology, 59*, 617–645.

Baseski, E., Pugeault, N., Kalkan, S., Bodenhagen, L, Piater, J.H., Krüger, N. (2010). Using multi-modal 3D contours and their relations for vision and robotics. J*ournal of Visual Communication and Image Representation, 2*, 850-864.

Beckers, T., De Houwer, J., & Eelen, P. (2002). Automatic integration of non-perceptual action effect features: The case of the associative affective Simon effect. *Psychological Research, 66*, 166–173.

Botvinick, M. M., Braver, T. S., Barch, D. M., Carter, C. S., & Cohen, J. D. (2001). Conflict monitoring and cognitive control. *Psychological Review, 108*, 624–652.

Botvinick, M. M., Buxbaum, L. J., Bylsma, L., & Jax, S. A. (2009). Toward an integrated account of object and action selection: A computational analysis and empirical findings from reaching-to-grasp and tool use. *Neuropsychologia, 47*, 671 - 683.

Braitenberg, V., & Schüz, A. (1991). *Anatomy of the cortex: Statistics and geometry*. Heidelberg, Germany: Springer.

Brooks, R. A. (1986). A robust layered control system for a mobile robot. *Journal of Robotics and Automation, 2*, 14-23.

Brooks, R. A. (1991). Intelligence without representation. *Artificial Intelligence, 47*, 139-159.

Butz, M. V. & Pezzulo, G. (2008). Benefits of Anticipations in Cognitive Agents. In G. Pezzulo et al. (Eds.): *The Challenge of Anticipation, A Unifying Framework for the Analysis and Design of Artificial Cognitive Systems,* pp. 45-62.

Byrne, M. D. (2003). Cognitive architecture. In J. Jacko & A. Sears (Eds.), *The human-computer interaction handbook: Fundamentals, evolving technologies and emerging applications* (pp. 97–117). Mahwah, NJ: Lawrence Erlbaum.

Card, S. K., Newell, A. & Moran, T. P. (1983). *The psychology of human-computer interaction.* Hillsdale, N.J: Erlbaum Associates.

Cohen, J. D., Dunbar, K., & McClelland, J. L. (1990). On the control of automatic processes: A parallel distributed processing account of the Stroop effect. *Psychological Review, 97*, 332–361.

Cohen, J. D., Servan-Schreiber, D., & McClelland, J. L. (1992). A parallel distributed processing approach to automaticity. *American Journal of Psychology, 105*, 239–269.

Cohen, A., & Shoup, R. (1997). Perceptual dimensional constraints in response selection processes. *Cognitive Psychology, 32*, 128-181.

Cos-Aguilera, I., Canamero, L., & Hayes, G. (2004). Using a SOFM to learn object affordances. *Workshop of Physical Agents* (WAF).

Cowey, A. (1985). Aspects of cortical organization related to selective attention and selective impairments of visual perception: A tutorial review. In M. I. Poster & O. S. M. Marin (Eds.), *Attention and performance XI* (pp. 41-62). Hillsdale, NJ: Erlbaum.

Deco, G., & Rolls, E. T. (2004). A neurodynamical cortical model of visual attention and invariant object recognition. *Vision Research, 44*, 621-642.

Dehaene, S., Sergent, C., & Changeux, J.-P. (2003). A neuronal network model linking subjective reports and objective physiological data during conscious perception. *Proceedings of the National Academy of Sciences of the United States of America, 100*, 8520–8525.

De Jong, R., Liang, C.-C., & Lauber, E. (1994). Conditional and unconditional automaticity: A dual-process model of effects of spatial stimulus-response correspondence. *Journal of Experimental Psychology: Human Perception and Performance, 20*, 731–750.

Detry, R., & Piater, J. (2011). Continuous surface-point distributions for 3D object pose estimation and recognition. *Lecture Notes in Computer Science, 6494*, 572–585.

DeYoe, E. A., & Van Essen, D. C. (1988). Concurrent processing streams in monkey visual cortex. *Trends in Neuroscience, 11*, 219–226.

Donders, F. C. (1868). Over de snelheid van psychische processen. Onderzoekingen, gedaan in het physiologisch laboratorium der Utrechtsche hoogeschool, 2. reeks, 2, 92–120.

Duncan, J., Humphreys, G., & Ward, R. (1997). Competitive brain activity in visual attention. *Current Opinion in Neurobiology, 7*, 255–261.

Duncan, J., & Owen, A. M. (2000). Common regions of the human frontal lobe recruited by diverse cognitive demands. *Trends in Neurosciences, 23*, 475–483.

Eenshuistra, R. M., Weidema, M. A., & Hommel, B. (2004). Development of the acquisition and control of action-effect associations. *Acta psychologica, 115*, 185–209.

Elsner, B., & Hommel, B. (2001). Effect anticipation and action control. *Journal of Experimental Psychology: Human Perception and Performance, 27*, 229–240.

Elsner, B., Hommel, B., Mentschel, C., Drzezga, A., Prinz, W., Conrad, B., et al. (2002). Linking actions and their perceivable consequences in the human brain. *Neuroimage, 17*, 364–372.

Eriksen B.A., & Eriksen C.W. (1974). Effects of noise letters upon the identification of a target letter in a non-search task. *Perception and Psychophysics, 16*, 143–149

Estes, Z., Verges, M., & Barsalou, L. W. (2008). Head up, foot down: object words orient attention to the objects' typical location. *Psychological Science, 19*, 93–97.

Fagioli, S., Hommel, B., & Schubotz, R. I. (2007). Intentional control of attention: Action planning primes action-related stimulus dimensions. *Psychological Research, 71*, 22–29.

Fischer, M. H., Castel, A. D., Dodd, M. D., & Pratt, J. (2003). Perceiving numbers causes spatial shifts of attention. *Nature Neuroscience, 6*, 555–556.

Fodor, J. (1983). *The modularity of mind*. Cambridge, MA: MIT Press

Gibson, J. J. (1979). *The Ecological Approach to Visual Perception*. Houghton Mifflin.

Glenberg, A.M. (1997). What memory is for. *Behavioral and Brain Science, 20*, 1–55.

Glenberg A.M., & Kaschak M.P. (2002). Grounding language in action. *Psychonomic Bulletin & Review, 9*, 558–565.

Glover, S. (2004). Separate visual representations in the planning and control of action. *Behavioral and Brain Sciences, 27*, 3–24.

Grossberg, S., & Grunewald, A. (1997). Cortical Synchronization and Perceptual Framing. *Journal of Cognitive Neuroscience, 9*, 117–132.

Grossberg, S., & Somers, D. (1991). Synchronized oscillations during cooperative feature linking in a cortical model of visual perception. *Neural Networks, 4*, 453–466.

Guiard, Y. (1983). The lateral coding of rotations: a study of the Simon effect with wheel-rotation responses. *Journal of Motor Behavior, 15*, 331–342.

Haazebroek, P., & Hommel, B. (2009a). Anticipative control of voluntary action: Towards a computational model. *Lecture Notes in Artificial Intelligence, 5499*, 31-47.

Haazebroek, P., & Hommel, B. (2009b). Towards a computational model of perception and action in human computer interaction. *Lecture Notes in Computer Science, 5620*, 247-256.

Haazebroek, P., Raffone, A., & Hommel, B. HiTEC: *A Connectionist Model of the Interaction between Perception and Action Planning*. Manuscript submitted for publication.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2009). Towards a computational account of context mediated affective stimulus-response translation. *Proceedings of the 31st Annual Conference of the Cognitive Science Society*. Austin, TX: Cognitive Science Society.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2011a). A computational model of perception and action for cognitive robotics. *Cognitive Processing, 12,* 355-365

Haazebroek, P., van Dantzig, S., & Hommel, B. (2011b). Interaction between Task Oriented and Affective Information Processing in Cognitive Robotics. *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, 59*, 34-41.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2013). How task goals mediate the interplay between perception and action. *Frontiers in Psychology, 4:247*.

Harless, E. (1861). Der Apparat des Willens. *Zeitschrift fuer Philosophie und philosophische Kritik, 38*, 50–73.

Hasbroucq, T., & Guiard, Y. (1991). Stimulus-response compatibility and the Simon effect: Toward a conceptual clarification. *Journal of Experimental Psychology: Human Perception and Performance, 17*, 246–266.

Heider, F. (1926/1959). Thing and medium. *Psychological Issues, 1959, Monograph 3* (original work published 1926).

Heider, F. (1930/1959). The function of the perceptual system. *Psychological Issues, 1959, Monograph*, 371–394. (original work published 1930).

Hommel, B. (1993). Inverting the Simon effect by intention: Determinants of direction and extent of effects of irrelevant spatial information. *Psychological Research, 55*, 270–279.

Hommel, B. (1996). No prevalence of right-left over above-below spatial codes. *Perception & Psychophysics, 58*, 102–110.

Hommel, B. (1997). Interactions between stimulus-stimulus congruence and stimulus-response compatibility. *Psychological Research, 59*, 248–260.

Hommel, B. (2000a). The prepared reflex: Automaticity and control in stimulus-response translation. In S. Monsell & J. Driver (Eds.), *Control of cognitive processes: Attention and performance* (Vol. XVIII, pp. 247–273). Cambridge: MIT Press.

Hommel, B. (2000b). Intentional control of automatic stimulus-response translation. In Y. Rossetti & A. Revonsuo (Eds.), *Interaction between dissociable conscious and nonconscious processes* (pp. 223–244). Amsterdam: John Benjamins Publishing Company.

Hommel, B. (2004). Event files: Feature binding in and across perception and action. *Trends in Cognitive Sciences, 8*, 494–500.

Hommel, B. (2009). Action control according to TEC (theory of event coding). P*sychological Research, 73*, 512–526.

Hommel, B. (2010). Grounding attention in action control: The intentional control of selection. In B.J. Bruya (ed.), *Effortless attention: A new perspective in the cognitive science of attention and action* (pp. 121-140). Cambridge, MA: MIT Press.

Hommel, B. (2011). The Simon task as tool and heuristic. *Acta Psychologica, 136*, 189–202.

Hommel, B. (2013). Ideomotor action control: On the perceptual grounding of voluntary actions and agents. In W. Prinz, M. Beisert & A. Herwig (Eds.), *Action science: Foundations of an emerging discipline* (pp. 113-136). Cambridge, MA: MIT Press

Hommel, B., & Elsner, B. (2009). Acquisition, representation, and control of action. In E. Morsella, J. A. Bargh, & P. M. Gollwitzer (Eds.), *Oxford handbook of human action* (pp. 371–398). New York: Oxford University Press.

Hommel, B., Müsseler, J., Aschersleben, G., & Prinz, W. (2001). The theory of event coding (TEC): A framework for perception and action planning. *Behavioral and Brain Sciences, 24*, 849–937.

Hommel, B., & Prinz, W. (1997). Theoretical issues in stimulus-response compatibility: An editor's introduction. In B. Hommel & W. Prinz (Eds.), *Theoretical issues in stimulus–response compatibility* (pp. 3–8). Amsterdam: North-Holland.

Hebb, D.O. (1949). *The Organization of Behavior*. New York: Wiley & Sons.

James, W. (1890). *The principles of psychology* (Vol. 2). New York: Dover Publications.

Kaschak, M. P., Madden, C. J., Therriault, D. J., Yaxley, R. H., Aveyard, M. E., Blanchard, A. A., & Zwaan, R. A. (2005). Perception of motion affects language processing. *Cognition, 94*, B79–B89.

Keele, S. W. (1968). Movement control in skilled motor performance. *Psychological Bulletin, 70*, 387–403.

Keysers, C., & Perrett, D. I. (2004). Demystifying social cognition: A Hebbian perspective. *Trends in Cognitive Sciences, 8*, 501–507.

Kieras, D. E., & Meyer, D. E. (1997). An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Human-Computer Interaction, 12*, 391—438.

Kirlik, A. (2007). Conceptual and Technical Issues in Extending Computational Cognitive Modeling to Aviation. *Proceedings of Human-Computer Interaction International*, 872–881.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics, 43*, 59–69.

Kornblum, S., Hasbroucq, T., & Osman, A. (1990). Dimensional overlap : Cognitive Basis for stimulus–response compatibility – A model and taxonomy. *Psychological Review, 97*, 253–270.

Kornblum, S., Stevens, G. T., Whipple, A., & Requin, J. (1999). The Effects of Irrelevant Stimuli 1: The time course of stimulus–stimulus and stimulus–response consistency Effects with Stroop-like stimuli, Simon-like tasks, and their factorial combinations. *Journal of Experimental Psychology: Human Perception and Performance, 25*, 688–714.

Kraft, D., Baseski, E., Popovic, M., Batog, A.M., Kjær-Nielsen, A., Krüger, N. et al. (2008). Exploration and Planning in a Three-Level Cognitive Architecture. *Proceedings of the International Conference on Cognitive Systems (CogSys 2008)*, Karlsruhe.

Kray, J., Eenshuistra, R.M., Kerstner, H., Weidema, M., & Hommel, B. (2006). Language and action control: the acquisition of action goals in early childhood. *Psychological Science, 17*, 737–741.

Kunde, W., Koch, I., & Hoffmann, J. (2004). Anticipated action effects affect the selection, initiation, and execution of actions. *Quarterly Journal of Experimental Psychology, 57*, 87–106.

Lakens, D. (2012). Polarity correspondence in metaphor congruency effects: structural overlap predicts categorization times for bi-polar concepts presented in vertical space. *Journal of Experimental Psychology: Learning, Memory and Cognition, 38*, 726–723.

Lamme, V. A. F., & Roelfsema, P. R. (2000). The distinct modes of vision offered by feedforward and recurrent processing. *Trends in Neurosciences, 23*, 571–579.

LeDoux, J. (1996). The emotional brain: The mysterious underpinnings of emotional life. New York: Touchstone.

Logan, G. D. (1988). Toward an instance theory of automatization. *Psychological Review, 95*, 492–527.

Logan, G. D., & Bundesen, C. (2004). Very clever homunculus: compound stimulus strategies for the explicit task-cuing procedure. *Psychonomic Bulletin & Review, 11*, 832–840.

Logan, G. D., & Crump, M.J.C. (2010). Cognitive illusions of authorship reveal hierarchical error detection in skilled typists. *Science, 330*, 683–686.

Lotze, R. H. (1852). *Medicinische Psychologie oder die Physiologie der Seele*. Leipzig: Weidmann'sche Buchhandlung.

MacLeod, C. M. (1991). Half a century of research on the Stroop effect: An integrative review. *Psychological Bulletin, 109*, 163–203.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. T*he Bulletin of Mathematical Biophysics, 5*, 115–133.

Meier, B. P., and Robinson, M. D. (2004). Why the sunny side is up: Associations between affect and vertical position. *Psycholigcal Science, 15*, 243–247.

Meier, B. P., Hauser, D. J., Robinson, M. D., Friesen, C. K., & Schjeldahl, K. (2007). What's "up" with god? vertical space as a representation of the divine. J*ournal of Personality and Social Psychology, 93*, 699–710.

Melcher, T., Weidema, M., Eenshuistra, R. M., Hommel, B., & Gruber, O. (2008). The neural substrate of the ideomotor principle: an event-related fMRI analysis. *NeuroImage, 39*, 1274–1288.

Memelink, J., & Hommel, B. (2005). Attention, instruction, and response representation. *European Journal of Cognitive Psychology, 17*, 674–685.

Memelink, J., & Hommel, B. (2013). Intentional weighting: A basic principle in cognitive control. *Psychological Research, 77*, 249–259.

Meteyard, L., Bahrami, B., & Vigliocco, G. (2007). Motion Detection and Motion Words: Language Affects Low-Level Visual Perception. *Psychological Science, 18*, 1007–1013.

Meteyard, L., Zokaei, N., Bahrami, B., & Vigliocco, G. (2008). Visual motion interferes with lexical decision on motion words. *Current Biology, 18*, 732–733.

Miller, E. K., & Cohen, J. D. (2001). An integrative theory of prefrontal cortex function. *Annual Review of Neuroscience, 24*, 167–202.

Milner, A. D. & Goodale, M. A. (1995). *The visual brain in action*. Oxford University Press.

Minsky, M. L., & Papert, S. A. (1969). *Perceptrons*. Cambridge, MA: MIT Press.

Monsell, S. (1996). Control of mental processes. In V. Bruce (Ed.), *Unsolved Mysteries of the Mind: Tutorial Essays in Cognition* (pp. 93–148). Erlbaum.

Moors, A., & De Houwer, J. (2006). Automaticity: a theoretical and conceptual analysis. *Psychological Bulletin, 132*, 297–326.

Murre, J. M. J., Phaf, R. H., & Wolters, G. (1992). CALM: categorizing and learning module, *Neural Networks, 5*, 55–82.

Müsseler, J., & Hommel, B. (1997). Blindness to response-compatible stimuli. *Journal of Experimental Psychology: Human Perception and Performance, 23*, 861–872.

Neisser, U. (1967). *Cognitive psychology*. New York: Appleton-Century-Crofts.

Neisser, U. (1992). Distinct systems for “where” and “what”: Reconciling the ecological and representational views of perception. *Paper presented at the Fourth Annual Convention of the American Psychological Society*, San Diego, CA.

Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.

Nicoletti, R., & Umiltà, C. (1984). Right–left prevalence in spatial compatibility. *Perception and Psychophysics, 35*, 333–343.

Nicoletti, R., & Umiltà, C. (1985). Responding with hand and foot: The right/left prevalence in spatial compatibility is still present. *Perception & Psychophysics, 38*, 211–216.

Nicoletti, R., Umiltà, C., Tressoldi, E. P., & Marzi, C. A. (1988). Why are left–right spatial codes easier to form than above–below ones? *Perception & Psychophysics, 43*, 287–292.

Nilsson, N.J. (1984), *Shakey the Robot*, SRI. Technical Note no. 323, SRI, Menlo Park, California.

Norman, D. (1988). *The Design of Everyday Things*. New York: Basic Books.

Nuxoll, A., & Laird, J. E. (2004). A cognitive model of episodic memory integrated with a general cognitive architecture. *Proceedings of the International Conference on Cognitive Modeling*.

O’Regan, J. K., & Noë, A. (2001). A sensorimotor account of vision and visual consciousness. *Behavioral and brain sciences, 24*, 939–972.

Oriet, C., Stevanovski, B., & Jolicoeur, P. (2001). Theory of event coding: Interesting, but underspecified. *Behavioral and Brain Sciences, 24*, 897–898.

Pecher, D., & Boot, I. (2011). Numbers in space: Differences between concrete and abstract situations. F*rontiers in Cognition, 2:121*.

Pecher, D., & Zwaan, R.A. (2005). *Grounding Cognition: The Role of Perception and Action in Memory, Language, and Thinking*. Cambridge: University Press.

Phaf, R. H., Van der Heijden, A. H. C., & Hudson, P. T. (1990). SLAM: a connectionist model for attention in visual selection tasks. *Cognitive Psychology, 22*, 273–341.

Pinker, S. (1997). How the mind works. New York: Norton.

Pitts, W., & McCulloch, W. S. (1947). How we know universals the perception of auditory and visual forms. *The bulletin of mathematical biophysics, 9*, 127–147.

Prablanc, C., & Pélisson, D. (1990). Gaze saccade orienting and hand pointing are locked to their goal by quick internal loops. In: *Attention and Performance. XIII. Motor Representation and Control*, edited by M. Jeannerod. Hillsdale, NJ: Erlbaum, 653-676.

Prinz, J. (2006). Is the mind really modular? In R. Stainten (Ed.), *Contemporary debates in cognitive science*, New York: Blackwell, 22–36.

Prinz, W. (1990). A common coding approach to perception and action. In O. Neumann & W. Prinz (Eds.), *Relationships between perception and action* (pp. 167–201). Berlin: Springer.

Prinz, W. (1992). Why don't we perceive our brain states? *European Journal of Cognitive Psychology, 4*, 1–20.

Prinz, W., & Hommel, B. (2002). *Common mechanisms in perception and action: Attention and performance XIV*. Oxford: Oxford Press.

Proctor, R. W. (2010). Playing the Simon game: Use of the Simon task for investigating human information processing. *Acta Psychologica, 136*, 182–188.

Proctor, R. W., & Cho, Y. S. (2006). Polarity correspondence: A general principle for performance of speeded binary classification tasks. *Psychological Bulletin, 132*, 416–442.

Proctor, R. W., & Lu, C. H. (1999). Processing irrelevant location information: Practice and transfer effects in choice-reaction tasks. *Memory and Cognition, 27,* 63–77.

Proctor, R. W., & Vu, K.-P. L. (2006). S*timulus-Response Compatibility Principles: Data, Theory, and Application*. Boca Raton, FL: CRC Press.

Proctor, R. W., Vu, K.-P. L., & Nicoletti, R. (2003). Does right–left prevalence occur for the Simon effect? *Perception and Psychophysics, 65*, 1318–1329.

Raffone, A., and Pantani, M. (2010). A global workspace model for phenomenal and access consciousness. *Consciousness and Cognition, 19*, 580–596.

Riggio, L., Gawryszewski, L. D. G., & Umiltà, C. (1986). What is crossed in crossed-hand effects? *Acta Psychologica, 62*, 89–100.

Rizzolatti, G., & Craighero, L. (2004). The mirror-neuron system. *Annual review of neuroscience, 27*, 169–92.

Rosenblatt, F. (1961). Principles of neurodynamics. perceptrons and the theory of brain mechanisms, *Cornell Aeronautical Laboratory Report VG 1196 G-8*, Buffalo, NY.

Rubichi, S., Gherri, E., Nicoletti, R., & Umiltà, C. (2005). Modulation of the vertical Simon effect in two-dimensional tasks: The effect of learning. *European Journal of Cognitive Psychology, 17*, 686–694.

Rumelhart, D. E., Hinton, G. E., & McClelland, J. L. (1986). A General Framework for Parallel Distributed Processing. In D. E. Rumelhart & J. L. McClelland (Eds.), P*arallel Distributed Processing: Explorations in the Microstructure of Cognition* (pp. 45–76). Cambridge, MA: MIT Press.

Santiago, J., Ouellet, M., Román, A., & Valenzuela, J. (2012). Attentional factors in conceptual congruency. *Cognitive Science, 36*, 1051-1077.

Schmidt, R. A. (1975). A schema theory of discrete motor skill learning. *Psychological Review, 82*, 225-260.

Schubert, T. W. (2005). Your highness: vertical positions as perceptual symbols of power. *Journal of Personality and Social Psychology, 89*, 1–21.

Schütz-Bosbach, S., & Prinz, W. (2007). Prospective coding in event representation. *Cognitive processing, 8*, 93–102.

Šetić, M., & Domijan, D. (2007). The influence of vertical spatial orientation on property verification. *Language and Cognitive Processes, 22*, 297–312.

Shanahan, M. P. (1998). Reinventing Shakey. *Working Notes of the 1998 AAAI Fall Symposium on Cognitive Robotics*, 12–135.

Simon, J. R., & Rudell, A. P. (1967). Auditory S-R compatibility: the effect of an irrelevant cue on information processing. *The Journal of Applied psychology, 51*, 300–304.

Sternberg, S. (1969) The discovery of processing stages: Extensions of Donders' method. In W. G. Koster (Ed.), Attention and performance II. *Acta Psychologica, 30*, 276–315.

Stoet, G., & Hommel, B. (2002). Interaction between feature binding in perception and action. In W. Prinz & B. Hommel (Eds)., *Common mechanisms in perception and action: Attention and performance XIV*. Oxford: Oxford Press.

Stock, A., & Stock, C. (2004). A short history of ideo-motor action. *Psychological Research, 68*, 176–188.

Stroop, J. R. (1935). Studies of interference in serial verbal reactions. J*ournal of Experimental Psychology, 28*, 643–662.

Tagliabue, M., Zorzi, M., Umiltà, C., & Bassignani, F. (2000). The role of LTM links and STM links in the Simon effect. *Journal of Experimental Psychology: Human Perception and Performance, 26*, 648–670.

Taylor, W.K. (1956). Electrical Simulation of Some Nervous System Functional Activities. In C. Cherry (Ed.) *Information Theory* (pp.314-28). London: Butterworths.

Taylor, L. J., & Zwaan, R. A. (2009). Action in cognition: The case of language. *Language and Cognition, 1*, 45–58.

Tononi, G., Sporns, O., & Edelman, G. M. (1992). Reentry and the problem of integrating multiple cortical areas: Simulation of dynamic integration in the visual system. *Cerebral Cortex, 2*, 310–335.

Treisman, A. (1996). The binding problem. *Current Opinion in Neurobiology, 6*, 171–178.

Tubau, E., Hommel, B., & López-Moliner, J. (2007). Modes of executive control in sequence learning: From stimulus-based to plan-based control. *Journal of Experimental Psychology: General, 136*, 43–63.

Umiltà, C., & Zorzi, M. (1997). Commentary on Barber and O'Leary: Learning and attention in S-R compatibility. In B. Hommel & W. Prinz (Eds.), *Theoretical issues in stimulus response compatibility* (pp. 173-178). Amsterdam: North-Holland.

Valle-Inclán, F., & Redondo, M. (1998). On the automaticity of ipsilateral response activation in the Simon effect. *Psychophysiology, 35*, 366–371.

van Dantzig, S., Raffone, A., & Hommel, B. (2011). Acquiring contextualized concepts: A connectionist approach. *Cognitive Science, 35*, 1162-1189.

van Steenbergen, H., Band, G. P. H., & Hommel, B. (2009). Reward counteracts conflict adaptation: Evidence for a role of affect in executive control. *Psychological Science, 20*, 1473–1477.

Wallace, R. A. (1971). S-R compatibility and the idea of a response code. *Journal of Experimental Psychology, 88*, 354-360.

Wang, D.-Y., D., Proctor, R. W., & Pick, D. F. (2007). Coding controlled and triggered cursor movements as action effects: Influences on the auditory Simon effect for wheel-rotation responses. *Journal of Experimental Psychology: Human Perception, 33*, 657–669.

Ward, R. (1999). Interactions between perception and action systems: a model for selective action. In G. W. Humphreys, J. Duncan, & A. Treisman (Eds.), *Attention, space and action: studies in cognitive neuroscience*. Oxford: Oxford University Press.

Ward, R. (2002). Coordination and integration in perception and action. In W. Prinz & B. Hommel (Eds)., *Common mechanisms in perception and action: Attention and performance XIV*. Oxford: Oxford Press.

Waszak, F., Hommel, B., & Allport, A. (2003). Task-switching and long-term priming: Role of episodic stimulus-task bindings in task-shift costs. *Cognitive Psychology, 46*, 361–413.

Watson, J. B. (1913). Psychology as the behaviorist views it. *Psychological Review, 20*, 158–177.

Wickens, J., Hyland, B., & Anson, G. (1994). Cortical cell assemblies: A possible mechanism for motor programs. *Journal of Motor Behavior, 26*, 66–82.

Wilson, M. (2002). Six Views of Embodied Cognition. *Psychonomic Bulletin & Review, 9*, 625–636.

Wilson, H., & Cowan, J. (1972). Excitatory and inhibitory interactions in localized populations of model neurons. *Biophysics Journal, 12*, 1–24.

Wolpert, D. M., & Ghahramani, Z. (2000). Computational principles of movement neuroscience. *Nature Neuroscience, 3*, 1212–1217.

Wykowska, A., Schubö, A., & Hommel, B. (2009). How you move is what you see: Action planning biases selection in visual search. *Journal of Experimental Psychology: Human Perception and Performance, 35*, 1755–1769.

Yamaguchi, M., & Proctor, R. W. (2011). The Simon task with multi-component responses: Two loci of response-effect compatibility. *Psychological Research, 75*, 214–226.

Yamaguchi, M., & Proctor, R. W. (2012). Multidimensional vector model of stimulus-response compatibility. *Psychological Review, 119*, 272–303.

Young, M. P. (1995). Open questions about the neural mechanisms of visual pattern recognition. In M. S. Gazzaniga (Ed.), *The Cogntive Neurosciences*. Cambridge, MA: Bradford.

Zanolie, K., van Dantzig, S., Boot, I., Wijnen, J., Schubert, T. W., Giessner, S. R., & Pecher, D. (2012). Mighty Metaphors: Behavioral and ERP Evidence That Power Shifts Attention on a Vertical Dimension. *Brain & Cognition, 78*, 50–58.

Zorzi, M., & Umiltà, C. (1995). A computational model of the Simon effect. *Psychological Research, 58*, 193–205.

Zwaan, R. A., & Taylor, L. J. (2006). Seeing, acting, understanding: motor resonance in language comprehension. *Journal of Experimental Psychology: General, 135*, 1–11.

Appendix

HiTEC incorporates realistic neuronal integration/decay properties, non-linear response (output) functions of excitatory and inhibitory neural units, as well realistic voltage-dependency of feedback connections and associative Hebbian learning. Incorporating these realistic neural properties implies various parameters in the model. The default values for all parameters are listed in Table 4.

Table 4. Default parameter values for HiTEC simulations

Category	Parameter	Default value
External inputs	External input sensory codes	0.5[a]
	External input motor codes (during learning)	0.3
Sensory weights	Sensory – feature forward	0.4[b]
	Sensory – feature backward	3.0
Stimulus features	Feature –task forward	1.3[c]
	Feature – task backward	0.2
Response features	Task – feature forward (location, intensity, etc.)	1.3
	Task – feature forward (other)	0.9[d]
	Task – feature backward	0.2
Inhibition	Excitatory to inhibitory paired unit	1.25
	Inhibitory to other excitatory codes within layer	-0.75
Code parameters	d_a decay parameter	0.1[e]
	qa sigmoid parameter in response function	0.9
	na sigmoid parameter in response function	4
	γ_{exc} scale parameter	0.9[f]
	γ_{inh} scale parameter	0.9[g]
Noise	Mean	0.025
	Standard deviation	0.015
Code thresholds	Voltage threshold (VT)	0.5
	Learning threshold (LT)	0.6[h]
	Response threshold for motor code selection	0.6
Learned weights	Learning rate (LR)	0.1
	Weight decay (d_w)	0.0005
	Weight scale factor (φ)	0.8[i]
General parameters	Action effects trials	10 trials per motor code
	Action effect duration (= cycles of weight learning)	50 cycles

Exceptions (see Table 4)

a. 0.6 in Simulation 6
b. 0.45 in Simulation 4 for the sensory to feature connections (visual to word); 0.3 for 'forward' and 'backward' in Simulation 6
c. 1.5 in Simulation 6
d. 0.55 in Simulation 2 and 0.6 in Simulation 3
e. 0.2 for sensory codes (all simulations)
f. 0.8 for sensory codes in Simulations 1 to 5; all codes 1.0 in Simulation 6
g. 0.8 for sensory codes in Simulations 1 to 5; all codes 1.0 in Simulation 6
h. 0.5 for Simulations 1 and 2
i. 1.0 in Simulation 6

Note that we have attempted to eliminate various scaling parameters in the model instance sued for Simulation 6. That is, γexc , γinh and have been set to 1.0. In order to retain the model dynamics, some of the other parameters were adjusted accordingly: external input and task-feature (other) weights.

Samenvatting

Hoe werkt onze interactie met de directe omgeving? Betrekkelijk moeiteloos openen wij deuren, reiken we voor een kop koffie en gebruiken wij een variatie aan gereedschappen en apparaten in ons dagelijks leven. Maar hoe lukt het ons eigenlijk om al onze acties zó aan te sturen dat we rekening houden met allerlei specifieke omstandigheden? We draaien onze hand immers zodanig dat we onze vingers gemakkelijk om de deurklink kunnen vouwen en we pakken een kop koffie op zo'n manier op dat we er ook nog uit kunnen drinken.

We zijn in staat tot dergelijke handelingen door eigenschappen, *features,* van waargenomen *stimuli* als het ware om te zetten in *features* van de handelingen, *actie responsen,* die wij uitvoeren. Het oortje van mijn kop koffie voor mij op tafel bevindt zich nu aan de rechterzijkant, dus ik kantel mijn rechterhand en vouw mijn vingers zodanig dat de afstand tussen de vingertoppen overeenstemt met de grootte van het oortje, terwijl ik mijn hand naar het kopje verplaats. Dit doe ik echter niet heel bewust, ondertussen denk ik na over deze tekst, maar als ik helemaal niet oplet stoot ik misschien wel het kopje omver. Het doorgronden van de processen onderliggend aan deze *stimulus-respons omzetting* is niet alleen van belang om het mentale functioneren van de mens beter te begrijpen, maar ook nuttig voor bijvoorbeeld het ontwerp van robots. Robots dienen immers ook voortdurend een vertaalslag te maken van waargenomen objecten naar zorgvuldig gecoördineerde acties.

De vraag is nu *hoe* waargenomen stimuli worden omgezet in uitvoerbare actie responsen. Traditionele theorieën (e.g., Donders, 1968; Sternberg, 1969) en ook meer specifieke computermodellen van menselijke informatieverwerking (e.g., Anderson, 1993; Card, Moran & Newell, 1983; Kieras & Meyer, 1997) veronderstellen dat deze omzetting in losse, opeenvolgende stadia plaatsvindt: eerst wordt een stimulus waargenomen, geanalyseerd en geselecteerd om op te reageren. Vervolgens wordt er een passende actie respons gekozen en uiteindelijk uitgevoerd. Met andere woorden, deze theorieën veronderstellen dat eerst perceptuele, dan cognitieve en tot slot actiegerelateerde processen plaatsvinden.

Hoewel deze theorieën en modellen intuïtief overkomen en zij veel fenomenen in de cognitieve psychologie hebben kunnen verklaren, hebben verschillende experimenten aangetoond dat sommige aspecten van stimulus-respons omzetting niet bewust werken. Eigenschappen van waargenomen stimulus objecten (zoals locatie, orientatie en grootte) kúnnen acties direct beinvloeden, buiten onze controle om. Dit wordt geïllustreerd door zogenaamde *stimulus-respons compatibiliteit* fenomenen, zoals het Simon effect (Simon & Rudell, 1967). In een typische Simon taak in een onderzoekslaboratorium worden verschillende stimuli één voor één op een beeldscherm getoond aan de proefpersoon. Deze stimuli zijn bijvoorbeeld rode of blauwe stippen. Bovendien worden ze soms links en soms rechts op het scherm weergegeven. Proefpersonen dienen alleen te letten op de kleur. Afhankelijk van de kleur dienen ze op een linker of rechter knop te drukken. Alhoewel de locatie van de stimulus er helemaal niet toe doet voor deze taak, heeft deze locatie desondanks wél invloed op de reactietijd en accuraatheid. Proefpersonen reageren sneller en maken minder fouten als de stimulus zich aan dezélfde kant (*compatibele* conditie) bevindt als de te geven respons dan wanneer de stimulus zich aan de tegengestelde kant bevindt (*incompatibele* conditie). Deze inmiddels vaak gerepliceerde bevinding suggereert dat er een *directe interactie* *is tussen stimulus waarneming en actie planning*. In meer recente modellen van stimulus-respons omzetting (e.g., Kornblum, Hasbroucq, & Osman, 1990; Zorzi & Umilta, 1995) wordt dit veelal opgelost door naast de 'gewone' perceptie-cognitie-actie route een éxtra route van perceptie naar actie te veronderstellen. Alhoewel dergelijke *dual route* modellen mogelijk maken om verschillende stimulus-respons compatibiliteit fenomenen te simuleren, blijft echter onduidelijk hoe en waarom sommige stimuluseigenschappen wél direct invloed hebben op actie planning, en andere eigenschappen niet.

Een alternatieve theorie, die zich expliciet richt op de directe interactie tussen perceptie en actie én op de representaties die deze interactie mogelijk kunnen maken, is de *Theory of Event Coding* (TEC, Hommel, Müsseler, Aschersleben & Prinz, 2001). Deze theorie stelt dat het brein stimulus features en actie features representeert gebruikmakend van een *gemeenschappelijke verzameling feature codes*. Hierbij verwijzen feature codes naar de 'distale' eigenschappen van objecten in de omgeving, zoals de globale vorm, grootte, afstand en locatie. Hierdoor zullen een tast-sensatie bij de linkerhand en de waarneming van een visuele stimulus ergens in het linker gezichtsveld beiden *dezelfde* distale feature code 'links' activeren. Van acties wordt verondersteld dat zij worden uitgevoerd door het activeren van *motor codes*. Het uitvoeren van een actie leidt tot een waarneembaar effect, een verandering in de omgeving, dat in feite weer een nieuwe stimulus vormt. Het waarnemen van dit effect leidt tot activatie van feature codes. Deze actieve feature codes worden vervolgens geassociceerd met de reeds actieve motor codes. Op basis van deze *associaties* kan vervolgens een actie doelgericht worden gepland en gecoördineerd. Door de feature codes horend bij een 'gewenst actie effect' te (her)activeren, zorgen deze associaties er namelijk voor dat de juiste motor codes worden geactiveerd en de bijhorende actie wordt uitgevoerd. Tot slot stelt TEC dat de taak context van belang is voor het representeren van stimuli en responsen. Wanneer, bijvoorbeeld, de taak is gegeven om een object op te pakken, dan worden díe feature codes versterkt in de waarneming die relevant zijn voor oppakken, zoals de vorm, grootte en locatie van het object, terwijl feature codes die minder relevant zijn voor deze taak, zoals de kleur of toonhoogte, verzwakt worden.

Dit proefschrift richt zich op de biologische en computationele plausibiliteit van deze gemeenschappelijke feature codes voor waarneming en actie planning. Hiertoe heb ik *HiTEC* ontwikkeld, een *connectionistisch* model gebaseerd op TEC. Connectionistsiche modellen houden, in tegenstelling tot de typische sequentiele modellen van informatieverwerking, ook rekening met de globale eigenschappen van het menselijk brein. Het brein bevat immers geen complexe, centrale processor, zoals bij een computer, die de informatie binnenkrijgt van de zintuigen, dan stapsgewijs alle berekeningen uitvoert en het resultaat tot slot naar de spieren stuurt. Integendeel, het brein bevat miljarden betrekkelijk simpele rekeneenheden, neuronen, die in grote netwerken met elkaar verbonden zijn. Via deze verbindingen krijgen zij signaaltjes binnen van hun buren en sturen zij weer signaaltjes door naar andere buren. Door het enorme aantal neuronen en de complexiteit van hun onderlinge verbindingen, en door de dynamiek waarmee nieuwe neuronen en verbindingen worden aangemaakt en bestaande verbindingen worden aangepast, is de werking van het brein niet eenvoudig te doorgronden.

In het connectionistische perspectief wordt een poging gedaan om – met de nodige simplificaties – een netwerkaanpak op informatieverwerking te onderzoeken. Hierbij worden netwerken gebouwd van eenvoudige elementen, *units*, die elk een zekere mate van *activatie* hebben. Zo'n netwerk wordt beschouwd als een *dynamisch* systeem waarbij signaaltjes binnenkomen bij een aantal units en deze dientengevolge gaandeweg een hogere activatie krijgen. Tegelijk verspreidt deze activatie zich naar de andere units in het netwerk via de onderlinge verbindingen en stabiliseert het netwerk zich na verloop van tijd. Van zo'n connectionistisch netwerk wordt gesteld dat het een 'taak uitvoert' door signalen (input) aan te bieden aan sommige units, te wachten tot het netwerk de activatie heeft verspreid en dan te bepalen wat de activatie van andere units (output) is. De verwerking van informatie in zo'n netwerk geschiedt niet in losse rekenstapjes, zoals bij de traditionele, sequentiele modellen, maar door het heen en weer verspreiden van activatie tussen de units. Er wordt dan ook gesteld dat informatieverwerking hier *gedistribueerd* is, vergelijkbaar met de interacties tussen de neuronen in het menselijk brein.

Ons doel voor HiTEC was om een duidelijk alternatief te formuleren voor de sequentiele modellen van stimulus-respons omzetting, en om tot een minimaal framework te komen waarin de interactie tussen perceptie, cognitie en actie processen leidt tot taakgericht gedrag. HiTEC bouwt verder op de principes van TEC om een verklaring te bieden voor een reeks aan experimentele bevindingen, in een eenduidig framework en op een niveau van specificiteit die computersimulatie mogelijk maakt.

Hoofdstuk 2 van dit proefschrift beschrijft het HiTEC model en de computationele uitwerking als connectionistisch netwerk. Hierbij staat de vraag centraal hoe een netwerk van neuron-achtige representaties stimulus-respons omzetting kan realiseren. In HiTEC zijn de representaties gedistribueerd over *meerdere niveaus*. Lagere niveaus corresponderen met sensorische input en motor output (respectievelijk *sensory codes* en *motor codes*), hogere niveaus met meer globale representaties (*feature codes* en *task codes*). Stimulus-respons omzetting vindt plaats door activatie door het model te verspreiden. Feature codes in HiTEC verwijzen naar de gemeenschappelijke representaties in TEC. Deze codes worden gebruikt voor zowel stimulus waarneming als actie planning. Bij stimulus-respons omzetting werken alle representaties *op alle niveaus tegelijk* samen. Dit heeft tot gevolg dat stimulus-respons omzetting volledig interactief is en niet sequentieel. Hogere niveau representaties versterken en verzwakken daarbij de gevoeligheid van representaties op lagere niveaus. Dit maakt mogelijk dat er directe interactie mogelijk is tussen perceptie en actie én dat deze interactie kan worden beïnvloed door de taak context. Het HiTEC model is gebruikt in alle simulaties zoals beschreven in Hoofdstukken 3 tot en met 5. In deze simulaties ligt de nadruk op aspecten die een uitdaging vormen voor de bestaande sequentiele modellen van stimulus-respons omzetting.

In Hoofdstuk 3 wordt ingegaan op de *situatie-specifieke betekenissen van motor acties*. Om adequaat te kunnen reageren op de omgeving dient het cognitieve systeem te weten wat voor acties er zoal mogelijk zijn en wat deze acties 'betekenen'. Verschillende empirische bevindingen suggereren dat deze betekenis niet een vaststaand feit is, maar afhangt van de (waarneembare) effecten van een actie binnen de taak context. Dit heeft tot gevolg dat als het systeem een passende actie repons moet selecteren en uitvoeren, het systeem eerst moet leren (op basis van ervaring) wat de effecten van zijn eigen motor acties zijn en hoe deze effecten binnen de taak context geïnterpreteerd dienen te worden. In bestaande modellen wordt deze betekenis veelal in het model ingebouwd, wat het voor deze modellen lastig maakt empirische bevindingen te verklaren die de flexibiliteit van deze betekenis demonstreren.

Simulatie 1 in dit hoofdstuk toont hoe in HiTEC nieuwe actie effecten spontaan worden geassocieerd met motor acties tijdens een fase van actie-effect leren. In Simulatie 2 wordt dit mechanisme gebruikt om een empirische studie van Kunde en collega's (Kunde, Koch, & Hoffmann, 2004) te simuleren. In deze simulatie wordt gedemonstreerd hoe de consistentie van verschillende actie effecten de interne representaties van deze effecten beïnvloedt. Aangezien actie-effect leren weer afhankelijk is van deze representaties, heeft de varierende consistentie consequenties voor de geleerde associaties tussen feature codes en motor codes. Deze associaties spelen vervolgens weer een cruciale rol in het plannen van acties. De simulatie laat zien dat het omzetten van stimuli naar respons vervolgens inderdaad beïnvloed wordt overeenkomstig de resultaten van de oorspronkelijke empirische studie. Dit demonstreert dat het mechanisme zoals voorgesteld in HiTEC in staat is om context-afhankelijke betekenis aan motor acties te verbinden en deze betekenis mee te nemen bij het plannen van deze acties als respons op aangeboden stimuli.

In Hoofdstuk 4 wordt verder ingegaan op *hoe en waarom sommige sommige aspecten van stimulus-respons omzetting automatisch geschieden*. Bestaande modellen van stimulus-respons compatibiliteit veronderstellen een extra route die het mogelijk maakt om deze fenomenen, zoals het eerder genoemde Simon effect (Simon & Rudell, 1976), te simuleren. Het blijft hierbij echter onduidelijk hoe en waarom sommige stimulus eigenschappen wél direct invloed hebben op actie controle, en andere eigenschappen niet. Simulaties in dit hoofdstuk, van het Simon effect en het Stroop effect (Stroop, 1935), demonstreren hoe HiTEC wél een duidelijke verklaring biedt voor deze effecten. In HiTEC zijn deze effecten in feite een onvermijdelijke consequentie van de representaties op verschillende niveaus, en hun onderlinge verbindingen. Ten eerste, om een taak te kunnen uitvoeren dienen zowel stimuli als responsen te worden gerepresenteerd op het distale representatieniveau en te worden verbonden met task codes. Ten tweede, acties die worden uitgevoerd in de taak context worden gerepresenteerd in termen van hun waarneembare effecten. Deze effecten bevinden zich in dezelfde perceptuele omgeving als de waarneembare stimuli. Dit betekent dat overlap in feature codes tussen stimulus features en actie features mogelijk en zeer waarschijnlijk is. Tot slot veronderstelt HiTEC dat stimulus perceptie en actie planning tegelijk plaatsvinden. Met andere woorden, de taakinstructie definieert een *task set* die wordt geïmplementeerd als een verwerkingspad van taak codes en specifieke feature codes. De omgeving leidt tot de mogelijke overlap van deze feature codes voor stimuli en responses. Aangezien stimulus percepie en actie planning vervolgens tegelijk plaatsvinden, dient het cognitieve systeem onvermijdelijk het expliciete verwerkingspad te combineren met feature code activatie. Code overlap tussen stimulus en respons features resulteert tot slot in facilitatie (compatibele conditie) of interferentie (incompatibele conditie) effecten.

Hoofdstuk 5 behandelt de vraag *hoe de taak context stimulus-respons omzetting beinvloedt*. In dit hoofdstuk worden zowel twee simulaties besproken als een nieuwe empirische studie waarbij gebruik is gemaakt van een Wii Balanceboard. De eerste simulatie in dit hoofdstuk, van een empirische studie van Hommel (1993), demonstreert hoe de taak context actie planning kan beïnvloeden door middel van aandacht voor verschillende elementen binnen de fysieke omgeving. In verschillende condities (vergelijkbaar met het origienele experiment) ontvangt het model verschillende taak instructies. De ene taak instructie benadrukt de visuele dimensie, de andere instructie de tast dimensie. Sensorische input van beide dimensies worden vervolgens op een gemeenchappelijke links-rechts *feature dimensie* gerepresenteerd. Zo heeft de taak instructie invloed op hoe een actie effect dat uit zowel een visuele als een tastcomponent bestaat intern wordt gerepresenteerd en daarmee op hoe actie-effect associaties worden geleerd. De gemeenschappelijke links-rechts feature dimensie wordt vervolgens ook gebruikt voor stimulus perceptie bij stimulus-respons omzetting waardoor er compatibiliteitseffecten optreden. Dit (automatische) compatibiliteitseffect is daarmee afhankelijk van de taakinstructie die een *weging van verschillende sensorische dimensies* bewerkstelligt.

De empirische studie en tweede simulatie in dit hoofdstuk laten op vergelijkbare wijze de invloed van taakinstructie op directe perceptie-actie interactie zien. In deze studie echter, wordt gedemonstreerd hoe alleen de taakomschrijving van verder ambigue acties invloed kan hebben op hoe de effecten van deze acties worden gerepresenteerd op het feature code representatieniveau, in dit geval voor-achter of links-rechts. Afhankelijk van de taak context worden actie effecten dus verschillend gerepresenteerd en de associaties tussen feature codes en motor codes dus ook verschillend geleerd. Bij de stimulus-respons omzetting van stimuli die ook een voor-achter of links-rechts component hebben tonen de verschillende instructiegroepen dan ook verschillen in hun compatibiliteitseffecten. In de HiTEC simulatie wordt duidelijk dat dit verklaard kan worden door *weging van verschillende feature dimensies* ten gevolge van de verschillen in taakinstructie.

Samen laten deze simulaties en de gedragsstudie zien dat de taakinstructie, geïnternaliseerd als verbindingen tussen feature codes en task codes, voldoende is om *top down* de verwerking van stimulus features en planning van actie features en hun automatische interactie te beïnvloeden. De weging van features lijkt dus te opereren op *verschillende niveaus van representatie*.

Tot slot worden in Hoofdstuk 6 de algemene conclusies beschreven. Vergeleken met de sequentiele modellen biedt HiTEC een meer representationele verklaring van gecontroleerde én automatische interactie tussen stimulus features en respons features. Hierbij wordt aangenomen dat er geen afzonderlijke deelstapjes zijn in deze processen, maar dat *representaties van verschillende nvieaus, tegelijk* en *met voortdurende onderlinge beïnvloeding* deze interactie realiseren. De netwerkaanpak in HiTEC past in de traditie van connectionische modellen waarin informatieverwerking gedisribueerd is over meerdere units. In veel van de gangbare modellen wordt echter wel een graduele maar strikte *feedforward* (eenrichtingsverkeer) architectuur aangehouden, terwijl er bij HiTEC sprake is van zowel *top down* als *bottum up* interactie tussen verschillende niveaus van representatie.

Bovendien wijzen de simulaties in dit proefschrift erop dat voor stimulus-respons omzetting met neuron-achtige units *teminste drie niveaus van representatie* nodig zijn. Zo vereisen sommige simulaties dat meerdere sensory codes verbonden zijn met dezelfde feature code, en andere simulaties dat dezelfde motor code geassocieerd kan worden met meerdere feature codes en dat dezelfde sensory code verbonden is met meerdere feature codes. De verbindingen van task codes met feature codes dienen eveneens flexibel te zijn om zo hetzelfde cognitieve systeem verschillende taken te kunnen laten uitvoeren.

Tot slot wordt benadrukt dat het connectionistische HiTEC model nog erg beperkt en gesimplificeerd is. De werking van de miljarden neuronen in het menselijk brein kent talloze bronnen van complexiteit, zoals de invloed van neurotransmitters. Evenzo zijn de verbindingen tussen neuronen vele malen complexer dan gesuggereerd in het huidige HiTEC model. Naast deze computationele simplificatie kent HiTEC ook beperkingen ten aanzien van de psychologische fenomenen die het model kan verklaren. Zo is het huidige model niet goed in staat meerdere objecten tegelijk waar te nemen en te representeren of om meerdere acties vooruit te plannen. Dit zijn interessante mogelijke uitbreidingen van het model die weer verschilende nieuwe uitdagingen met zich mee brengen.

Het proefschrift besluit met de conclusie dat HiTEC is voorgesteld als een *proof of principle*, gericht op hoe gemeenschappelijke representaties de interactie tussen perceptie en actie medieren. Dit heeft geresulteerd in een model dat zonder expliciete deelstappen stimuli in responsen kan omzetten en daarbij een expliciete behandeling mogelijk maakt van verschillende issues – situatie-specifieke betekenis van acties, selectieve automaticiteit en de invloed van taak context – die allen relevant zijn voor effectief gedrag.

Dus, hoe openen wij deuren en pakken wij kopjes koffie op? En hoe zouden robots dit ook kunnen aanpakken? Resultaten van dit proefschrift suggereren dat wij niet eerst waarnemen, dan denken en dan pas acties uitvoeren; integendeel, wij stellen eerst ons perceptie-actie systeem in op basis van de taak die wij dienen uit te voeren, en we coördineren deze omzetting vervolgens op louter abstract niveau. Hierdoor hoeven we ons niet continu en bewust te bekommeren om elk klein detail en is er genoeg denkkracht over om bijvoorbeeld dit proefschrift te lezen.

Acknowledgements

Een proefschrift geldt als een individueel eindproduct, een proeve van bekwaamheid. Maar het proces, de weg ernaartoe, is een complex geheel van interacties met tal van mensen zonder wie het eindproduct niet zou bestaan.

Allereerst Walter en Fons, jullie hebben mij enthousiast gemaakt voor wetenschap. Gwen, jouw bijzonder prettige begeleiding bij mijn scriptie heeft mij overgehaald dit avontuur tegemoet te treden en Joyca, jij was een enorm welkome aanvulling op het Leidse PACO team. Saskia, zonder jou had ik de stap naar spannend empirisch onderzoek niet gezet en jouw vrolijke aanwezigheid, sterke betrokkenheid en pragmatische aanpak waren alles wat ik me een kamer- en projectgenoot kon wensen!

Antonino, without you I would not have succeeded in the simulations. I would probably still be staring at the list of parameters and paper-and-pencil sketches of what the model was supposed to do. Thank you for inviting me to Rome, it was a wonderful experience!

AIO-collega's, Michiel, André, Merel, Nelleke, Steph, Marieke, Rinus, Henk, Stephen, Mikael, Roy, Sharon, Anne, Kalinka, Jun, Kerwin, Edward, Floor, Anne-Wil, Claire en anderen, die eenzelfde proces (hebben) doorlopen, van al onze gesprekken heb ik erg veel energie en inspiratie gekregen. Anne, Kalinka, Floor, met jullie heb ik in het bijzonder de ups en downs van de PHD rollercoaster gedeeld en zeer waardevolle herinneringen opgedaan. Joost, dank voor al je reflecties en de vruchtbare samenwerking; Frank, een bijpraatsessie met jou brengt altijd nieuwe ideeën! en Randy, je was een fantastische kracht in het Wiilab en je hebt mij geïnspireerd tot het empirische werk in dit proefschrift.

Dit proefschrift zou ook zeker niet zijn gelukt zonder secretariele ondersteuning. Albertien, Atie, Marianne en Eva, heel erg bedankt voor al jullie praktische hulp en luisterend oor, voor de sociale stroomlijning in onze sectie, en voor de reminders voor zaken die bij mij soms bleven steken.

Ook vanuit mijn vrienden heb ik veel steun mogen ontvangen tijdens dit proces. Armin, de werkweken in Friesland en vooral in Florence waren effectief en minstens zo leuk! Maarten, onze pauzes brachten de nodige afstand en bezinning. Jeroen, Tim en Martijn, de VTG breaks trokken die mij even uit de papers. Martijn, thanx, niemand wilde zó aanhoudend weten hoe mijn model nu precies werkte, als jij. Leander, bedankt voor de jaarlijkse goede gesprekken op een plein in Berlijn! Annelies, Erica, Linda, Louise, George, Hylke, dank voor jullie moral support door de jaren heen. Maaike, dankjewel voor al je geduld en begrip tijdens de vele uren die ik aan mijn laptop heb moeten slijten.

Daniël, broer!, ik heb je hulp en inspanning zeer gewaardeerd in je rollen als gastspreker, ondernemer en cameraman bij mijn vakken en als reisgenoot bij mijn conferentiebezoek. Jouw energie en gedrevenheid inspireren mij om ook in mijn projecten door te zetten.

En tot slot mijn ouders. Pa, Mama, ik weet dat ik niet altijd duidelijk heb kunnen maken waar ik precies mee bezig was, maar evengoed is álles wat er is gelukt vooral gelukt dankzij jullie. Bedankt voor jullie onvoorwaardelijke steun.

Curriculum Vitae

Pascal Haazebroek was born in The Hague on March 23rd, 1980. He attended Gymnasium Haganum in The Hague from 1992 to 1998. After completing secondary education he started studying Computer Science at Leiden University in 1998 and started working at Fortis Bank as ICT specialist. In 1999, he also started studying Psychology at Leiden University. In 2000, he started working as a teaching assistant for various courses at the Computer Science department. In 2004, he graduated in Computer Science with an investigation of automated usability evaluation and datamining of user interactions on websites. In 2005, he worked at 2C Communication Consultancy as a usability consultant and graduated in Cognitive Psychology on the neurodynamic simulation of visual search. He also started a PHD project at Computer Science, and continued to pursue a PHD at the Cognitive Psychology department in 2006. Here, he participated in PACO-PLUS, a European project on Cognitive Robotics involving research institutes throughout Europe. During his PHD he developed a number of courses, such as Human Computer Interaction, in close collaboration with various companies. In January 2013, he started a position as docent at the Leiden Institute of Psychology teaching various courses in bachelor, international bachelor, and master. Pascal aspires a career as both lecturer and researcher focusing on both theoretical and applied topics in psychology.