

A Model of Computation and Representation in the Brain

James S. Albus
Krasnow Institute for Advanced Studies¹
George Mason University
Fairfax, Virginia
james.albus@gmail.com

Short abstract

The brain is first and foremost a control system that builds an internal representation of the external world, and uses this representation to make decisions, set goals and priorities, formulate plans, and control behavior with intent to achieve its goals. The computational model proposed here assumes that this internal representation resides in arrays of cortical columns that, together with their underlying thalamic nuclei, are *Cortical Computational Units* (CCUs.) It is suggested that it may be possible to reverse engineer the human brain at the CCU level of fidelity using next-generation massively parallel computer hardware and software.

Abstract

The brain is first and foremost a control system that is capable of building an internal representation of the external world, and using this representation to make decisions, set goals and priorities, formulate plans, and control behavior with intent to achieve its goals. The internal representation is distributed throughout the brain in two forms: 1) firmware embedded in synaptic connections and axon-dendrite circuitry, and 2) dynamic state variables encoded in the firing rates of neurons in computational loops in the spinal cord, midbrain, subcortical nuclei, and arrays cortical columns. It assumes that clusters and arrays of neurons are capable of computing logical predicates, smooth arithmetic functions, and matrix transformations over a space defined by large input vectors and arrays. Feedback from output to input of these neural computational units enable them to function as finite-state-automata (fsa), Markov Decision Processes (MDP), or delay lines in

¹ The majority of this paper was written while the author was an employee of the National Institute of Standards and Technology. Therefore, this document is not subject to copyright.

processing signals and generating strings and grammars. Thus, clusters of neurons are capable of parsing and generating language, decomposing tasks, generating plans, and executing scripts. In the cortex, neurons are arranged in arrays of cortical columns that interact in tight loops with their underlying subcortical nuclei. It is hypothesized that these circuits compute sophisticated mathematical and logical functions that maintain and use complex abstract data structures. It is proposed that cortical hypercolumns together with their underlying thalamic nuclei can be modeled as a *Cortical Computational Unit* (CCU) consisting of a frame-like data structure (containing attributes and pointers) plus the computational processes and mechanisms required to maintain it and use it for perception cognition, and sensory-motor behavior. In sensory-processing areas of the brain, CCU processes enable focus of attention, segmentation, grouping, and classification. Pointers stored in CCU frames define relationships that link pixels and signals to objects and events in situations and episodes. CCU frame pointers also link objects and events to class prototypes and overlay them with meaning and emotional values. In behavior-generating areas of the brain, CCU processes make decisions, set goals and priorities, generate plans, and control behavior. In general, CCU pointers are used to define rules, grammars, procedures, plans, and behaviors. CCU pointers also define abstract data structures analogous to lists, frames, objects, classes, rules, plans, and semantic nets. It is suggested that it may be possible to reverse engineer the human brain at the CCU level of fidelity using next-generation massively parallel computer hardware and software.

Key Words: brain modeling, cognitive modeling, human neocortex, image processing, knowledge representation, perception, reverse engineering the brain, segmentation, signals to symbols

1. Introduction

Much is now known and progress is rapid in the neuroscience and brain modeling fields. A great deal is understood regarding how the brain functions. [48, 80, 31, 34, 49, 24, 25, 26, 37, 50] The structure and function of the visual system is perhaps best understood. [64] Much is known in the computer science and intelligent systems engineering fields about how to embed knowledge in computer systems. [54, 14, 65, 66, 34, 56, 45, 73] Researchers in robotics, automation, and intelligent control systems have learned how to build intelligent systems capable of performing complex operations in dynamic, real-world, uncertain, and sometimes hostile, environments. [5, 6, 57, 63] Reference model architectures and software development methodologies and environments have evolved over the past three decades that provide a systematic approach to engineering intelligent systems. [4, 58] Supercomputer hardware is approaching the estimated speed and memory capacity of the human brain, and computational power is increasing by an order of magnitude every five years. [77]

This paper is an attempt to integrate knowledge from all of these disciplines and more. It incorporates concepts from artificial intelligence (modern and GOFAI²), control theory, computer vision, robotics, neuroscience, and cognitive psychology, into a model of computation and representation in the human brain that is well suited for implementation on next generation computational hardware and software systems. By no means is this an exhaustive review of these many diverse fields. The papers cited are merely pointers into a massive and rapidly growing literature upon which the model of computation and representation presented here is based.

This model is the product of a biologically inspired engineering perspective at the intersection of neuroscience, computer science, and control theory. It is informed by over 40 years of experience in the design, engineering, and testing of a wide variety of intelligent machine systems that have successfully performed complex tasks in real-world environments. These include sensory-interactive robot manipulators [8], the NBS Automated Manufacturing Research Facility (AMRF) [79], control system architectures for automated coal mining systems, automation for Post Office general mail facilities and stamp distribution centers, intelligent controllers for multiple undersea vehicles, automation systems for next generation nuclear submarines, enhanced machine tool controllers for automotive and aircraft prototype machining and assembly cells, advanced controllers for commercial water jet cutting machines, and a number of Army Research Lab, DARPA, and Federal Highway Administration research programs for intelligent unmanned ground vehicles. [7, 2, 58, 81]

Each of these systems represents a milestone in the evolution of the RCS (Real-time Control System.) [3] RCS is a family of intelligent control systems that sense the environment, process sensory input to build and maintain an internal model of the world, and use that model to generate behavior in pursuit of operational goals. The original version of RCS was inspired by the Marr-Albus model of the cerebellum. [60, 12] It consisted of a hierarchy of computational units, each of which inputs a motor command from above, combines it with feedback from below, and outputs control commands to subordinate units to generate coordinated actions that drive the system toward the goal. [11] This line of research culminated 30 years later in the 4D/RCS reference model architecture developed for the Army Research Lab Demo III Experimental Unmanned Vehicle program. [19, 82] The 4D in 4D/RCS refers to the incorporation of the 4D approach to dynamic vision developed by Ernst Dickmanns and his students for control of autonomous vehicles.³ [27]

Unmanned vehicles based on 4D/RCS have been built and tested under rigorous experimental conditions. In 2002, vehicles with 4D/RCS controllers were judged by the U.S. Army to have achieved Technology Readiness Level VI (i.e., successful demonstration in a relevant environment.) As a result, the robot vehicle technology

² Good Old Fashioned Artificial Intelligence

³ The 4D approach was incorporated into the RCS architecture under an agreement between the U.S. Army Research Lab and the German Ministry of Defense.

was deemed ready for inclusion in the Army Future Combat System (FCS.) [22] An Autonomous Navigation System (ANS) based in 4D/RCS is currently being developed for command and control of all combat vehicles in the FCS, both manned and unmanned. [32] Despite recent cuts in the FCS budget, the ANS program has been continued.

Many of the features of the model of computation and representation presented here are derived from principles of the 4D/RCS reference model architecture. In order to explain these principles, Section 2 provides an introduction to 4D/RCS. Section 3 provides a mapping between the 4D/RCS architecture and the functionality of the human brain. Section 4 describes the computational architecture of cortical columns and the communications channels between them. Section 5 discusses the functional processes that populate the computational architecture in the brain. Section 6 presents the central hypothesis of this paper, i.e., the Cortical Computational Unit and how it is hypothesized to function in posterior and frontal cortices. Section 7 is a short discussion of learning and memory. Section 8 is a comparison of the CCU model with other models in the literature. Section 9 suggests how CCUs might be used for reverse engineering the brain. Section 10 is a summary.

2. Structure of the 4D/RCS Model

4D/RCS is a Reference Model Architecture for the design of intelligent machine systems. An architecture is the structure that identifies, defines, and organizes components, their relationships, and principles of design; the assignment of functions to subsystems and the specification of the interfaces between subsystems. A reference model architecture is an architecture in which the entities, events, relationships, and information units involved in interactions between and within subsystems and components are defined and modeled.

It is important to note that an architecture is not an algorithm or a procedure, or even a collection of algorithms. An architecture is a structure that can be populated by algorithms and procedures. An architecture is a framework that organizes algorithms and procedures so that they operate effectively and communicate efficiently. It is the structure that provides the synchronization in time and space between the computational elements. The 4D/RCS architecture is designed to accommodate a wide variety of algorithms and models, including those that employ symbolic, iconic, and connectionist mechanisms and representations. 4D/RCS provides the operating environment where many different algorithms can be housed and selectively invoked under favorable circumstances, and supplied with parameters so that they function effectively and efficiently.

At the highest level of abstraction, the 4D/RCS architecture is a classical sense-model-act paradigm shown in Figure 1. A Behavior Generation (BG) process accepts goals, makes decisions, creates plans, and controls actions to achieve or maintain its goals using information provided by an internal World Model (WM).

Goals can either be generated internally by a motivation process, or input from an external source. In a typical engineered intelligent system such as 4D/RCS, the goal comes from the outside. In a biological system it may be either generated internally, or imposed by an authority figure from the outside.

A collection of Sensory Processing (SP) functions operate on data from sensors to keep the World Model current and accurate. The World Model includes both a knowledge database and a set of world modeling processes that provide three basic functions:

- 1) Build and maintain the knowledge database,
- 2) Service requests for information from sensory processing and behavior generation processes
- 3) Create predictions based on knowledge stored in the knowledge database. Short-term predictions are generated for Sensory Processing to support focus of attention, segmentation, and recursive estimation. Long-term predictions are generated for Behavior Generation to support decision-making, planning, and control.

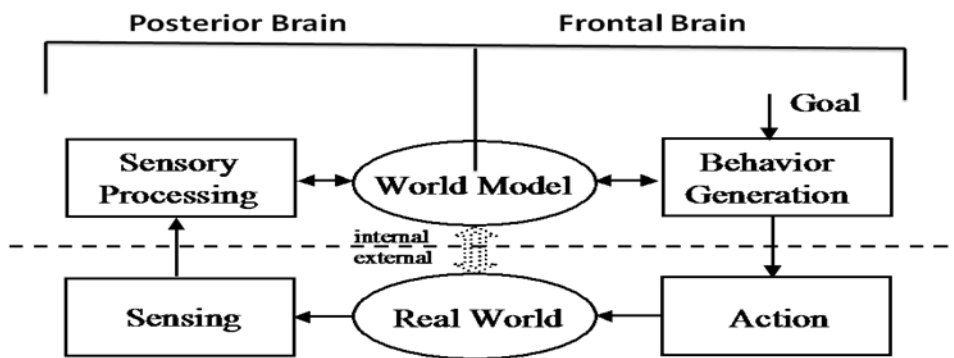


Figure 1. The fundamental structure of the RCS reference model architecture. Behavior Generation uses knowledge in a World Model to produce action designed to achieve the goal. Sensory processing uses data from sensors to build and maintain the World Model. A rough mapping of this diagram onto the brain is shown at the top.

The fact that 4D/RCS is a control system means that real-time issues of bandwidth, latency, timing, speed, and accuracy are important for system performance. Delay or noise in a feedback control loop limits accuracy and responsiveness, and can cause instability. Recursive filtering and predicting are used to overcome noise and enable preemptive action. Model based estimation techniques such as Kalman filtering are used to enhance performance. [27]

Behavior Generation uses knowledge in the World Model to make decisions, set priorities, generate plans, and control actions that produce results in the external world. If the World Model can predict the future state of the world, Behavior Generation can generate tactics, plans, and actions that anticipate events and prevent undesirable situations from occurring.

The flow of information between the World Model and Behavior Generation is bi-directional. While the World Model provides Behavior Generation with information regarding the state of the external world, Behavior Generation provides the World Model with information about the state of the task. This enables the World Model to represent what task is in progress, and what commands are currently being generated at each echelon in the BG hierarchy. Behavior Generation also informs the World Model about plans for possible future actions. The World Modeling processes can then simulate the probable results of these possible future actions, and a Value Judgment (VJ) process computes an estimate of cost, benefit, and risk. This enables Behavior Generation to choose among alternative future courses of action. This three-way conversation between Behavior Generation, World Modeling, and Value Judgment is a planning loop.

The flow of information between the World Model and Sensory Processing is also bi-directional. While Sensory Processing keeps the World Model updated, the World Model provides context and predictions that assist Sensory Processing in the interpretation of sensory data. The World Model also provides knowledge of what is important, for focusing attention. The World Model provides predictions of what kinds of objects and events to expect, where and when to expect them to occur, and what attributes and behaviors to expect them to exhibit. This information is used for segmentation, grouping, tracking, and classification of targets; and for processing of temporal sequences. Value Judgment computes confidence in classifications and assigns worth to perceived objects, events, and situations.

2.1 First Level of Detail

A first level of detail in the 4D/RCS reference model is shown in Figure 2. Behavior is generated by Behavior Generation (BG) processes (Planners and Executors) supported by Task Knowledge. Task knowledge is procedural knowledge, i.e., skills, abilities, and knowledge of how to act to achieve task goals under various conditions. Task knowledge includes requirements for tools and resources, lists of objects that are important to task performance, and expected behavior of objects subjected to planned actions. In 4D/RCS, task knowledge can be represented in the form of task frames, recipes, schema, state graphs, procedures, programs, rules, or flow charts. Task knowledge is used by Planners and Executors to make decisions, generate plans, and control behavior. Some task knowledge may be acquired by learning how to do things, either from a teacher or from experience. Other task knowledge is embedded in BG software in the form of algorithms and data structures containing implicit and explicit *a priori* information about the world and the intelligent system. Still other task knowledge is embedded in the circuitry which communicates messages between functional modules in the control system.

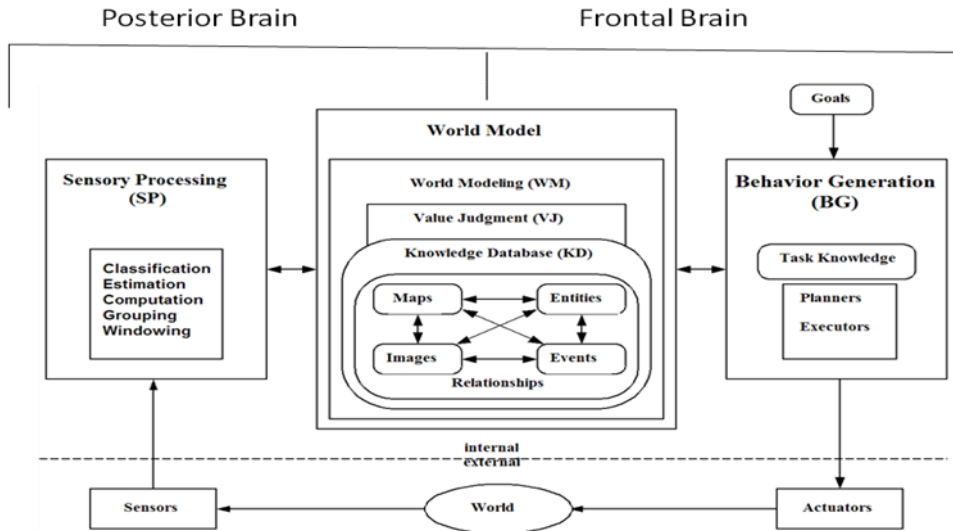


Figure 2. A first level of detail in the basic structure of intelligent systems. Processes are represented by boxes with square corners. Data structures are represented by boxes with rounded corners. Data flow between processes and pointers within KD are represented by arrows. A rough mapping of this diagram onto the brain is shown at the top.

Perception is enabled by Sensory Processing (SP) processes that operate on sensory input to window (i.e., focus attention), segment and group entities and events, compute group attributes, do recursive estimation, and perform classification operations. The World Model is composed of a Knowledge Database (KD), a set of World Modeling processes, and a set of Value Judgment processes.

The rough mappings onto the brain shown in Figures 1 and 2 are not intended to be neuomorphic. They do, however, suggest that those portions of the WM that are tightly coupled with perception are in the posterior brain, and those portions tightly deeply involved goal selection, task decomposition, and planning are located in the frontal brain. The posterior and frontal portions of the WM are interconnected by massive tracts of fibers such as the longitudinal and arcuate fasciculi that connect the frontal and posterior brain. [86]

2.2 World Modeling (WM) processes and the Knowledge Database (KD)

In 4D/RCS, knowledge is represented in both iconic and symbolic forms. Iconic forms include images and maps. These are two-dimensional projections of the three-dimensional structure of the external world. Images are in the coordinate frame of the sensors. Maps are in horizontal coordinate frames that are ideal for planning behavior. For example, an overhead map view is useful for planning foot placement in rough terrain. For pursuing prey or attacking an enemy, an image in egosphere coordinates is more useful. In 4D/RCS, these different coordinate frames are simultaneously maintained and updated in real-time.

In the brain, images exist in the retina and the visual cortex, while maps of the surface of the body exist in the somatosensory cortex. Iconic representations also exist in the posterior parietal cortex, the hippocampus, and possibly other regions. Images and maps may contain non-uniform resolution. Sensory processing in the retina produces an image with high resolution color pixels in the center, and lower resolution gray-scale images in the periphery. In the 4D/RCS world model, map resolution is high at the egocenter, and decreases exponentially with range. In 4D/RCS, maps are typically overlaid with symbols that describe features such as roads, bridges, streams, and terrain contour lines. Military maps may contain control lines between sectors of responsibility.

Iconic information may be represented by two-dimensional arrays of pixels. Each pixel is a picture element corresponding to a piece of real estate on a sensory surface. Each pixel has an address corresponding to its location in the array. Each pixel generates a set of signals that can be represented by a set of variables that define: 1) an attribute vector (intensity, color, temperature, pressure, etc.), 2) a state vector (range and velocity), and 3) pointers that define relationships and class membership.

Symbolic forms include entities, events, situations, episodes, goals, tasks, and plans. Entities represent spatial patterns or segmented regions of space. Events represent temporal patterns or segmented intervals of time. In 4D/RCS, entities and events can be represented by abstract data structures such as LISP frames, C structs, and C++ or Java objects and classes. Relationships are represented by pointers. Entities and events in the Knowledge Database (KD) can be linked by pointers to represent places, situations, and episodes. Parent-child relationships and class membership relationships can be represented by “belongs-to” or “has-part” pointers. Situations and places can be represented by graph structures and semantic nets. Episodes are strings of situations that occur over extended periods of time. Episodes can be described by linguistic structures such as words, phrases, sentences, and stories. Spatial, temporal, mathematical, social, and causal relationships can be represented by abstract data structures and pointers that link them together in networks that assign context and meaning.

Knowledge in the KD can be provided a priori, and updated in real-time by Sensory Processing. Knowledge can also be provided by subject matter experts, or acquired via learning from experience. In the lower levels of the 4D/RCS hierarchy, knowledge in the KD is updated every few milliseconds. At higher levels, trends and historical records may span minutes, hours, days, weeks, or years.

2.3 Sensory Processing (SP) processes

Direct contact with the external world is provided by input from sensors. In 4D/RCS systems, input from sensors comes from cameras, LADAR, radar, acoustic sensors, accelerometers, microphones, vehicle internal sensors, radio communications, and GPS receivers. In the brain visual images are input from

arrays of photodectors in the retina. Tactile feelings are input from arrays of touch, pressure, vibration, temperature, and pain sensors in the skin. Sounds are input from arrays of acoustic sensors in the ears. A sense of gravity and inertial space is derived from sensors in the vestibular system. A sense of body position and movement is derived from groups of proprioceptive sensors that measure position, velocity, and tension in the muscles and joints. Smell and taste are provided by sensors in the nose and tongue.

Signals from sensors enter the sensory processing system in iconic form. Spatial and temporal patterns of signals from sensors are transformed into symbolic form through the operations of segmentation and grouping. Pixels in images are segmented and grouped into spatial patterns, or entities, such as edges, surfaces, objects, groups, situations, and places. Time varying signals from sensors are segmented and grouped into temporal patterns such as events and episodes. For example acoustic signals may be segmented and grouped into clicks, grunts, phonemes, words, sentences, songs, stories, and episodes. Patterns of signals from inertial and odometry sensors can be grouped into body position, orientation, and velocity. In 4D/RCS, patterns of bits on radio channels are segmented and grouped into messages that convey commands, priorities, and rules of engagement. Patterns of bits on GPS receivers provide map position coordinates.

Grouping and segmentation operations can generate pointers that link iconic to symbolic representations, and vice versa. Pointers can link pixels in images and maps to symbolic frames representing entities, events, and classes. Forward pointers linking iconic to symbolic representations provide the basis for symbolic reasoning. This enables an intelligent system to perceive the world not as a collection of pixels and signals, but as a montage of objects, events, situations, and episodes. Meaning occurs when signals from sensors are grouped into entities and events that are linked to Value Judgment significance and worth variables. Back-pointers link symbolic frames to images and maps that can be projected back onto the sensory data stream, and from there back onto the world. Back-pointers provide the basis for symbol grounding. This enables the intelligent system to project context and meaning onto sensory experiences, and hence onto the external environment. Two-way links between iconic and symbolic forms provide the basis for scene and speech understanding, abductive inferencing, and symbol grounding.

The KD includes knowledge stored in three forms: 1) long-term memory, 2) short-term memory, and 3) immediate experience. The 4D/RCS model assumes that long-term memory is symbolic and stored in non-volatile media. Short-term memory is symbolic and stored in volatile media such as finite state automata or recirculating delay lines. Immediate experience is iconic and stored in dynamic registers and arrays within active processes such as recursive estimation loops, adaptive resonance circuits, and phase-lock loops.

2.4 Value Judgment (VJ) Processes

The VJ processes provide a variety of evaluation functions required for intelligent decision-making, planning and control. VJ processes provide criteria for the selection of goals and assignment of priorities. VJ processes evaluate the cost, benefit, and risk of planned behaviors. VJ processes provide the criteria for selecting modes of behavior such as aggressive vs. cautious, or fight vs. flee.

VJ processes also assign worth and emotional significance to entities, events, situations, and episodes, and provide the basis for deciding whether something is worth storing in long-term memory. VJ processes provide criteria for assessment of what is good or bad, attractive or repulsive, beautiful or ugly. VJ processes determine what is to be feared or hoped for, loved or hated, important or insignificant. VJ processes also compute levels of confidence for perceptual hypotheses. In the human brain, VJ processes in the limbic system provide values that give rise to a sense of good and evil, love and hate, hope and fear, duty, justice, and morality.

2.5 Behavior Generation (BG) processes

A fundamental feature of the RCS architecture is its hierarchical structure. The BG hierarchy consists of echelons of nodes containing intelligent agents. A WM hierarchy supplies an internal representation of the external world, and a SP hierarchy extracts information about the world from the sensory data stream.

The 4D/RCS organizational hierarchy developed for the Army Research Lab Demo III program is shown in Figure 3. At each echelon in the BG hierarchy, RCS nodes decompose task commands from a higher echelon node into subtasks for one or more subordinate nodes in a lower echelon. At each level in the SP hierarchy, RCS nodes focus attention, segment and group patterns of sensory data from lower levels, and compute attributes and classifications of those patterns for higher levels. At each level and each echelon, WM processes use SP results to maintain KD data with range and resolution needed to support BG planning and control functions in each node. VJ processes provide evaluations to support decision-making, planning, control, memory, and focus of attention in BG, WM, and SP processes in each node.

Each 4D/RCS node has a well-defined role, set of responsibilities, span of control, range of interest, and resolution of detail in space and time. These nodes can be configured to model any management style, and can be reconfigured at any time in response to changing task priorities and availability of resources.

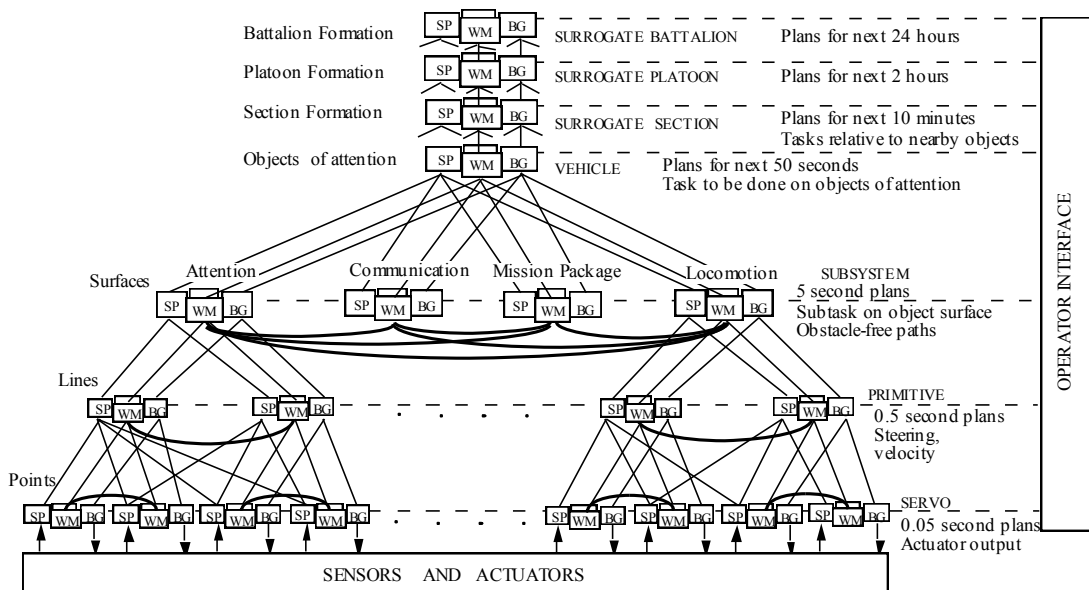


Figure 3. The 4D/RCS reference model architecture developed for the Army Research Laboratory Demo III experimental unmanned ground vehicle program. This example is for an autonomous vehicle in a scout platoon. [5]

Figure 3 is a reference model architecture for a single scout vehicle in a section of a platoon attached to a battalion. A similar reference model might apply to a single human being embedded in a social structure consisting of an immediate family, an extended family, and a tribe. Processing nodes are organized such that the BG processes form a chain of command. There are horizontal communication pathways within nodes, and information in the knowledge database is shared between WM processes in nodes above, below, and at the same level within the same subtree. On the right in Figure 3, are examples of the functional characteristics of the BG processes at each echelon. On the left, are examples of the scale of maps generated by SP-WM processes and populated by the WM in the KD at each level. VJ processes are hidden behind WM processes in the diagram. A control loop may be closed at every node. Numerical values in the figure are representative examples only. Actual numbers depend on parameters of specific vehicle dynamics. It should be noted that there are not necessarily the same number of SP levels as BG echelons. This is discussed in more detail in [4].

In 4D/RCS, an operator interface provides access to every node. This allows for teaching, debugging, performance monitoring, and operator control at any node in the 4D/RCS architecture

The BG process in each node has a well-defined and limited set of task skills. Goals are represented as desired state vectors. Task knowledge can be represented by frames, and plans by directed graphs. Each echelon in the BG hierarchy relies

on the echelon above to define goals and priorities, and provide long-range plans. Each node relies on the echelon below to carry out the details of assigned tasks. Within each node, the KD provides a model of the external world at a range and resolution that is appropriate for behavioral decision-making and planning activities that are the responsibility of that node. This hierarchical distribution of roles and responsibilities provides a way to manage computational complexity as systems scale up to real-world tasks and environments.

Note that in Figure 3 there are surrogate nodes for the Section, Platoon, and Battalion echelons. These enable any individual vehicle to assume the role of a section, platoon, or battalion commander. Surrogate nodes also provide each vehicle with higher echelon plans, models, goals, rules of engagement, and priorities during those periods of time when the vehicle is not in direct communications with its supervisor. Surrogate nodes in every individual vehicle enable it to cooperate effectively with others, and act appropriately in teams, even when contact with supervisor nodes is interrupted, as often happens in the field.

In mapping Figure 3 onto the brain, the Servo echelon roughly corresponds to the spinal motor centers. The Primitive echelon roughly corresponds to the midbrain and cerebellum. The Subsystem echelon corresponds to the primary sensory-motor cortices. The higher levels of Vehicle, Section, Platoon, etc. correspond to the higher levels and echelons in the frontal, posterior, and limbic cortices.

As noted in Figures 1 and 2, the WM modules are split between those parts that support BG decision making, planning, and control; and those parts that support SP focus of attention, segmentation, grouping, estimation of group attributes, and classification.

Figure 4 is a more detailed view of the first five echelons in the chain of command containing the Autonomous Mobility Subsystem in the 4D/RCS architecture developed for the Demo III program. On the right of Figure 4, the Behavior Generation hierarchy (consisting of Planners and Executors) decomposes high echelon mission commands into low echelon actuator commands. The text inside the Planner at each echelon indicates the planning horizon at that echelon.

The Executor at each echelon executes the plan generated by its Planner. Meanwhile, the Planner in each echelon is replanning based on an updated world state. Each planner has a world model simulator that is appropriate for the problems encountered within the node at its echelon. The Planners and Executors operate asynchronously. At each echelon, the Planner generates a new plan and the Executor outputs new commands to subordinates on the order of ten times within each planning horizon. At each lower echelon, the planning horizons shrink by a factor of about ten. The relative timing relationships between echelons are designed to facilitate control stability and smooth transitions among hierarchical control echelons. The timing numbers in Figure 4 are illustrative only. The actual rates may be situation dependent.

It should be noted, however, that timing is a central feature. 4D/RCS is a real-time intelligent control system designed for operations in real battle environments with human-robot teams of warriors. As opposed to most AI systems designed for operating in simulated environments or simple laboratory experiments, the entire family of RCS controllers has been designed for implementation in real physical systems with state-of-the-art performance.

In the center of Figure 4, each map has a range and resolution that is appropriate for path planning at its echelon. At each echelon, there are symbolic data structures and segmented images with labeled regions that describe entities, events, and situations that are relevant to decisions that must be made at that echelon. On the left is a sensory processing hierarchy that extracts information from the sensory data stream that is needed to keep the world model knowledge database current and accurate.

At the bottom of Figure 4 are actuators that act on the world and sensors that measure phenomena in the world. The Demo III vehicles were designed to accommodate a variety of sensors that include a LADAR, stereo CCD cameras, stereo FLIRs, a color CCD, vegetation penetrating radar, GPS (Global Positioning System), an inertial navigation package, actuator feedback sensors, and a variety of internal sensors for measuring parameters such as engine temperature, speed, vibration, oil pressure, and fuel level. The vehicles also may carry a Reconnaissance, Surveillance, and Target Acquisition (RSTA) package that includes long-range cameras and FLIRs, a laser range finder, and an acoustic package. Actuators control steering, brakes, throttle, sensor suite pan/tilt units, and various mission package operating modes.

The bottom (Servo) echelon in Figure 4 has no map representation. The Servo echelon deals with actuator dynamics and reacts to sensory feedback from actuator sensors. The Primitive echelon map has range of 5 m with resolution of 4 cm. This enables the vehicle to make small path corrections to avoid bumps and ruts during the 500 ms planning horizon of the Primitive echelon. The Primitive echelon also uses accelerometer data to control vehicle dynamics during high speed driving. The Subsystem echelon map has range of 50 m with resolution of 40 cm. This map is used to plan about 5 s into the future to find a path that avoids obstacles and provides a smooth and efficient ride. The Vehicle echelon map has a range of 500 m with resolution of 4 m. This map is used to plan paths about 1 min into the future taking into account terrain features such as roads, bushes, gullies, or tree lines. The Section echelon map has a range of 2 km with resolution of about 30 m. This map is used to plan about 10 min into the future to accomplish tactical behaviors. Higher echelon maps (not shown in Figure 4) can be used to plan platoon, company, and battalion missions lasting about 1 h, 5 h, and 24 h respectively. These are derived from military maps and intelligence provided by the digital battlefield database.

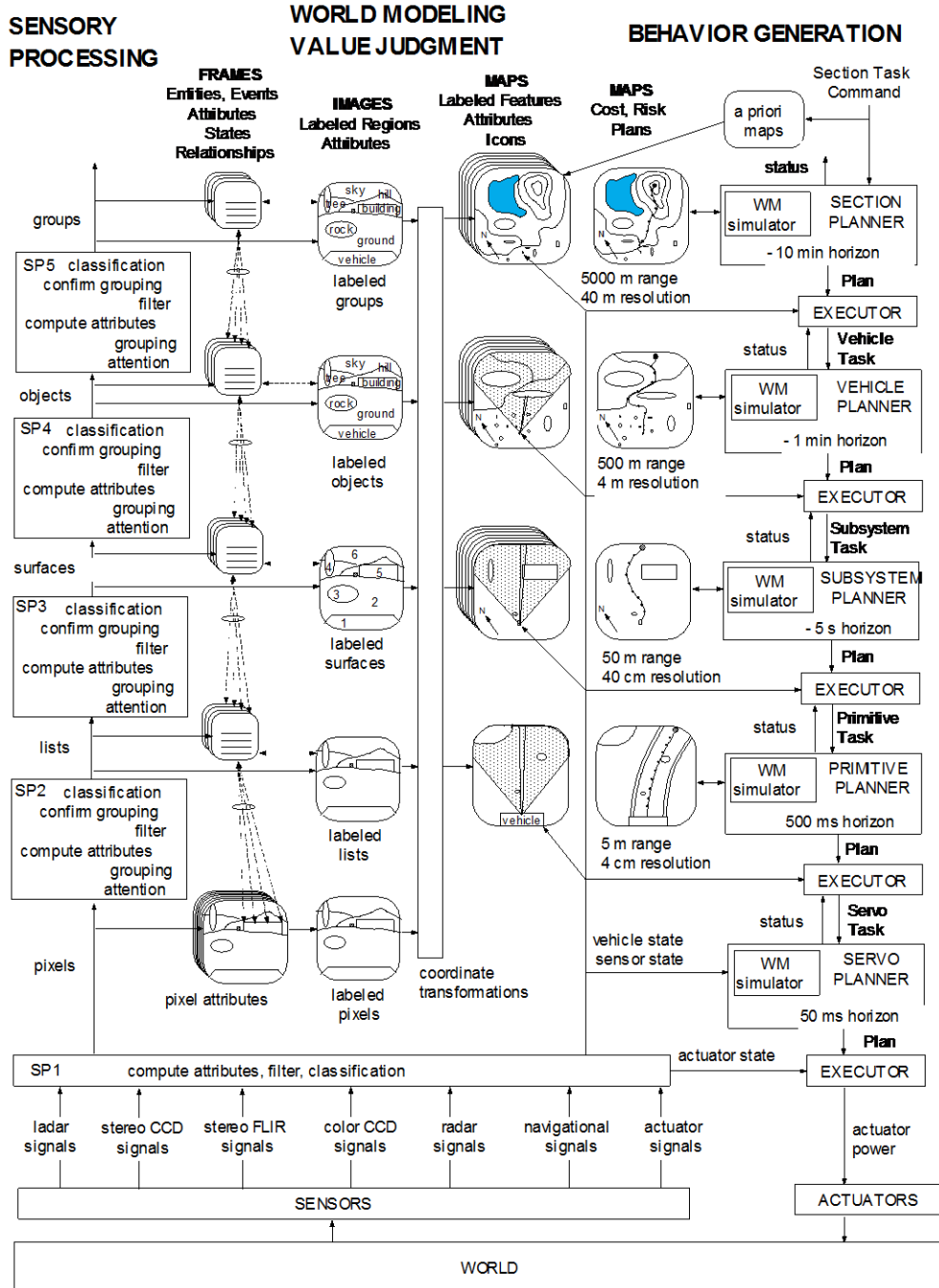


Figure 4. Five echelons of the 4D/RCS architecture for Demo III. On the right are Planner and Executor modules. In the center are maps for representing terrain features, road, bridges, vehicles, friendly/enemy positions, and the cost and risk of traversing various regions. On the left are Sensory Processing functions, symbolic representations of entities and events, and segmented images with labeled regions. The coordinate transforms in the middle use range information to assign labeled regions in the entity image hierarchy on the left to locations on planning maps on the right. [4. 5]

At all echelons, 4D/RCS planners are designed to generate new plans well before current plans become obsolete. Thus, action always takes place in the context of a recent plan, and feedback through the executors closes reactive control loops using recently selected control parameters. To meet the demands of dynamic battlefield environments, the 4D/RCS architecture specifies that replanning should occur within about one-tenth of the planning horizon at each echelon.

Executors are designed to react to sensory feedback even faster than the replanning interval. The Executors monitor feedback from the lower echelons on every control cycle. Whenever an Executor senses an error between its output CommandGoal and the predicted state (status from the subordinate BG Planner) at the GoalTime, it may react by modifying the commanded action so as to cope with that error. This closes a feedback loop through the Executor at that echelon within a specified reaction latency.

The BG hierarchy is not fixed. It can be reconfigured at any time so that subsystems within vehicles can be replaced, or vehicles can be reassigned to different chains of command whenever required by the assigned mission.

Each node of the BG hierarchy is supplied with that portion of the world model that is relevant to its responsibilities; at a level of resolution in space and time that is appropriate to its control functions. In this world model, arrays of signals are linked to objects and events with attributes and state, that are represented in coordinate frames that are optimized for perception, cognition, and intelligent behavior.

Figure 4 is included here to illustrate the kinds of computation and representation that have proven experimentally successful for the real world task of driving an unmanned ground vehicle to a designated location through hilly wooded terrain filled with obstacles and hazards using both off-road and on-road driving behaviors.

The CCU model presented below in Section 6 assumes that even more sophisticated computations and richer representations must occur in the brain to enable human levels of performance on similar tasks.

2.5 Interaction between Sensory Processing and World Modeling

The role of Sensory Processing and World Modeling is to build and maintain a distributed internal model of the external world, with range and resolution that is appropriate for Behavior Generating processes at every echelon of the BG hierarchy. Perception is accomplished by interactions between SP and WM processes. At each level in the SP hierarchy, patterns in the sensory input are segmented from the background and grouped into entities and events. For each entity or event, pointers are established that define relationships to other entities and

events at higher, lower, and the same hierarchical levels, and to the regions in space and time that contain them.

The diagram in Figure 5 shows how bottom-up SP processes are influenced by top-down information from *a priori* KD knowledge and BG representations of tasks and goals at a single hierarchical level. Comparable interactions occur at every level in the SP hierarchy.

At the bottom left of Figure 5, subentity images enter a SP level to be processed. At the lowest level, a subentity image is simply an array of pixels from a camera or a retina, or a sensory array on the skin or cochlea. At higher levels, a subentity image is the output from a lower level SP/WM processes. The SP process of windowing operates to attenuate regions of the image that are without behavioral significance, and focus SP and WM resources on regions that are important to achieving behavioral goals. The SP process of segmentation separates subentities that belong to entities of interest from those that do not. The SP grouping process clusters subentities into entities based on gestalt hypotheses (e.g., proximity, similarity, good continuation, symmetry, etc.) The grouping process labels the segmented subentities with the name (or address) of the entity to which they belong. At various levels in the SP-WM hierarchy, grouping yields entity images of edges, boundaries, surfaces, objects, or groups of objects. The result of each grouping operation is a hypothesized entity image wherein each pixel in the entity image is labeled with the name of the group to which it belongs. Each grouping operation generates pointers from subentities to entities, and vice versa. This establishes links between labeled regions in the iconic representation, and named entity or event frames in the symbolic representation. This is discussed in more detail in Sections 6.3 and 6.4.

Once segmentation and grouping have been achieved, SP and WM computation processes can then compute the value of entity attributes (e.g., size, shape, color, and texture) and state (e.g., position, orientation, and motion) for each segmented region of interest in the entity image. Next, SP/WM recursive estimation processes generate predicted entity attributes to be compared with observed entity attributes. When predicted attributes match observed attributes over time, confidence in the gestalt grouping hypothesis is increased. When the confidence rises above threshold, the grouping hypothesis is confirmed.

During the recursive estimation process, small differences between predictions and observations are used to update the model. Large differences may cause the level of confidence in the grouping hypothesis to fall below threshold. When this happens, the hypothesis is rejected, and another gestalt hypothesis must be generated. If a suitable hypothesis cannot be found, then the observed region in the image is declared novel, and worthy of inclusion on the list of entities of attention as a region of interest.

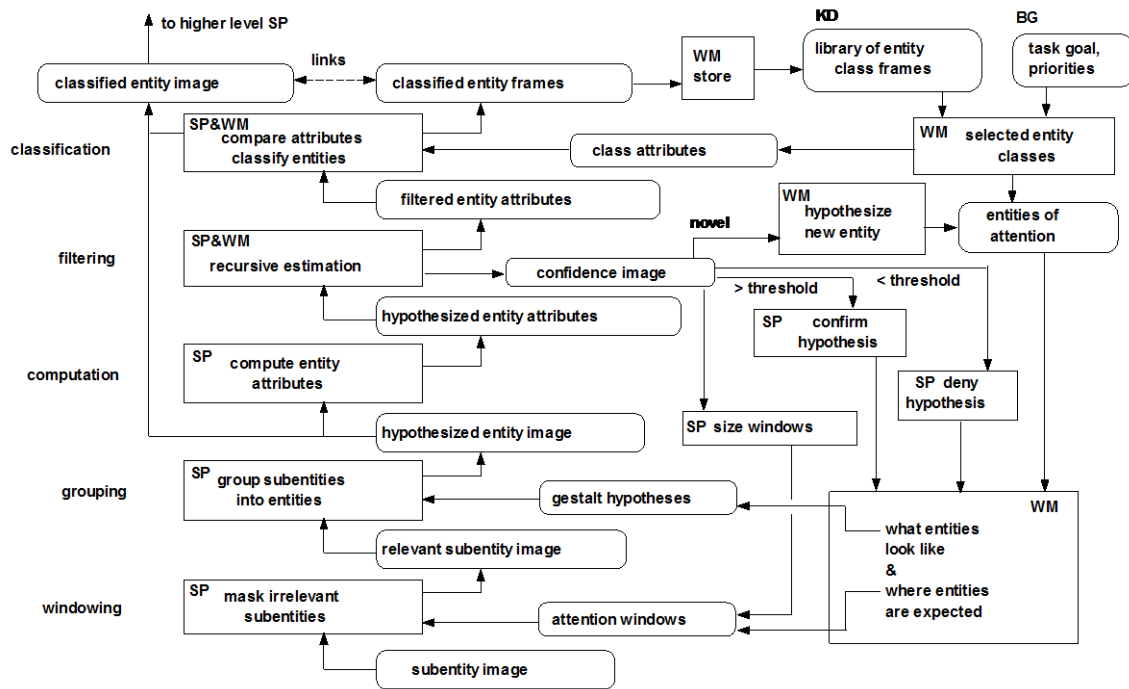


Figure 5. A data flow diagram of interactions between SP and WM processes in a typical SP-WM level. Functional operations are shown in boxes with square corners. Data structures are shown in boxes with rounded corners. [4, 5]

Once the grouping hypothesis is confirmed, the list of attributes in the confirmed entity frame can be compared with the attributes of stored entity class prototypes in the KD. When a match is detected, the entity is assigned to the matching class. This establishes a class pointer from the entity frame to the name (or address) of the class prototype frame. Each pixel in the entity image can then inherit additional class attributes through its link with the entity frame. Finally, a VJ process determines whether or not the classified entity is of sufficient importance to be stored in long-term memory. If so, then a WM process will enter the classified entity frame into long-term memory in the KD.

Top-down information from BG processes about task goals and priorities enter Figure 5 at the top-right. This top-down information enables a WM process to select a set of entity classes that are important to the task from a library of prototype entity class frames that resides in long-term memory. This set of important entities is prioritized and combined with bottom-up information about novel regions of interest. The result is a list of entities or regions of attention. This list is used by WM processes at the bottom right of Figure 5 to generate expectations and predictions regarding where these entities and regions of attention should be expected to appear in the image, and what they should be expected to look like.

What entities are expected to look like is defined by the attributes in the prototype entity class frames. *What* information provides guidance to the heuristic selection of gestalt hypotheses that will be used to control the grouping of subentities into entities. *Where* entities can be expected to appear in the image can be computed from the state-variables in the entity class frames. *Where* information provides guidance to the heuristic processes that define windows of attention to be used to control sensory pointing, tracking, and segmentation operations.

In summary, the 4D/RCS reference model architecture specifies an intelligent control system that:

1. Gathers input from sensors and transforms it into a rich, dynamic, meaningful, real-time internal representation of the external world. That representation consists of images and maps, entities and events, situations and episodes, with pointers that link all these together in relationships that are overlaid with cost-risk-benefit values needed for generating successful behavior.
2. Uses this internal representation to make decisions, set goals and priorities, formulate plans, and control behavior with intent to achieve goals.
3. Partitions both sensory processing and behavior generation into many hierarchical levels with characteristic range and resolution in time and space at each level.
4. Uses feedback at all BG echelons to detect and correct errors so as to achieve goals despite uncertainties and unexpected perturbations.
5. Uses model based predictions at all echelons to anticipate future events, and to take action to maximize the probability of success in achieving goals and objectives.

According to the classification scheme of Weng [88], 4D/RCS is a Type-6 Markov Decision Process architecture. That is, it consists of hierarchical stacked arrays of Markov decision processors that implement sensory processing, world modeling, behavior generation, and value judgment functions. As a reference model architecture, 4D/RCS does not specify the algorithms for *how* those functions are implemented – only *what* they do, and in what form they store and communicate information. Thus, 4D/RCS nodes may be implemented by neurons, neural nets, mathematical or logical functions, computational procedures, or computer programs. Although it possesses the “sense–model–act” structure of a GOFAI architecture, 4D/RCS is much more. It is a hierarchical control system that contains both iconic and symbolic representations at multiple levels of resolution in space and time, with links between pixels and objects that provide symbol grounding in real-time.

More detail about the 4D/RCS reference model architecture is available in [4, 5, 58].

3. Modeling Computation and Representation in the Brain

The model of computation and representation presented here combines many of the engineering principles derived from 4D/RCS with knowledge of the brain derived from neuroscience, behavioral science, and cognitive psychology. It uses the language of computer science and control theory to describe functionality in the brain. It uses mathematical and computational concepts and data structures such as state-variables, vectors, arrays, objects, classes, addresses, pointers, and functions to describe activity in the neuropile. The goal is to bridge the gap between neuroscience, computer science, and control theory.

Our hypothesized model provides a conceptual framework in which:

- a) to transform data from sensors into perceived objects, events, and relationships in the environment
- b) to reason about perceived objects, events, and relationships in the current situation as part of a historical episode;
- c) to make decisions for action within the context of a set of goals and priorities;
- d) to generate a hierarchy of plans to achieve those goals with minimum cost and maximum benefit;
- e) to command the muscles to generate coordinated forces and motions that produce external behavior that is goal directed and effective.

The top of Figure 1 shows a hypothesized mapping of the 4D/RCS model onto the brain at the highest level of abstraction. Sensory Processing and those parts of the World Model that service Sensory Processing are located in the posterior brain, while Behavior Generation and those parts of the World Model that service Behavior Generation are located in the frontal brain. The posterior and frontal parts of the World Model are connected in the brain by massive fiber bundles such as the longitudinal and arcuate fasciculi that run fore and aft under the cortex. [86] The frontal World Model is assumed to include the basal ganglia and some regions of the cerebellum. These communication pathways suggest how the physical structure of the brain can instantiate a distributed but integrated internal representation of both the external world, and the internal state of the body, with a self at the center of an egocentric coordinate frame.

3.1 Basic Assumptions

Our model makes a number of basic assumptions about the structure and function of the brain. These are:

1. The brain is first and foremost a control system designed by natural selection to achieve the top level goals of survival and propagation of the gene pool.

2. The neural circuitry in the brain is much more computationally sophisticated than is considered biologically plausible by most neuroscientists.
3. The brain maintains a dynamic internal representation of the external world that is much richer and more meaningful than the sensory input.
4. The conscious self has access only to the internal representation – not the external world. The internal representation is what the conscious self perceives to be external reality.
5. The internal representation is what the self uses to make decisions, select goals, set priorities, generate plans, and command actions intended to achieve its goals.
6. The self perceives itself at the center of its own universe, with the external world represented in an egocentric polar coordinate frame.
7. Perception is a set of processes that gather input from many different sensors and transform it into a unified but distributed internal representation of the external world.

The egosphere is a polar coordinate frame analogous to the celestial sphere used in astronomy. We can define a number of egospheres with different origins and different orientations.. A sensor egosphere can be defined by the position and orientation of the sensory array. A head egosphere may be defined by the position and orientation of the skull. The north polar axis of the head egosphere is the top of the head, zero heading is defined by the tip of the nose, and the origin is at eye level on the midline of the brain. We can define a self egosphere that is congruent with the head egosphere. An inertial egosphere can be defined by the gravity acceleration vector and a distant point on the horizon. A world egosphere can be defined by local geodetic coordinates.

The surface of the egosphere can be tessellated into pixels. Each pixel represents a solid angle projected onto the egosphere. Each pixel has a location on the egosphere and can be described by an attribute vector. Elements of the attribute vector can include brightness, color, spatial and temporal derivative, and range to the region in the 3D world onto which that pixel projects. The 2D structure of the egosphere maps directly onto the 2D structure of the retina, the thalamus, the superior colliculus, and the visual cortex. A 2D array of range attribute values is often referred to as a 2.5D image. The representation of range in the 2.5D image dramatically reduces the size of the data structure needed to represent 3D space. The volume of 3D space is infinite, but the area on the surface of the egosphere is finite, and equal to 4π steradians, or roughly 40,000 square-degrees.

It is assumed that the brain has evolved biological processes that are functionally equivalent to most, if not all, of the forms of computation and representation that have been discovered and used by mathematicians, computer scientists, and engineers for representing physical reality. Specifically, it is assumed that processes in the brain provide the biological equivalent of state-variables, vectors, arrays, frames, abstract data structures, data management techniques, and communication mechanisms. Some of these have yet to be demonstrated in the

neuropile, but all have been proven theoretically possible. The computational equivalence between general purpose computers and biological brains has long been established at the theoretical level. [87, 61] This paper will attempt to make some of these equivalences more specific in a form that could be implemented in current, and near term future, computer software and hardware.

3.2 Overall Structure of the Brain

It is assumed here that form follows function in the brain. Among the most obvious features of both the brain and body is that they are symmetrical about the medial plane. The left side of the brain represents the right hemisphere of egospace and controls the right side of the body. The right side of the brain represents the left hemisphere of egospace and controls the left side of the body. The two sides of the brain communicate through the corpus callosum, the anterior commissure, and many sub cortical structures in the midbrain and spinal cord. The two sides of the neocortex form a symmetrical pair of large two-dimensional arrays – one for each hemisphere of the egosphere.

The brain is partitioned back to front into posterior regions that are primarily concerned with sensing and perception, and frontal regions that are primarily concerned with behavior. This partition exists at all levels, from bottom to top. At the bottom in the spinal cord, sensory neurons enter the posterior roots, and motor neurons exit the anterior roots. This partition continues up the centerline of the cord through the midbrain. At the top of the brain in the cortex, the partition travels along the central sulcus in the cortex.⁴ Two-way communications between posterior and frontal regions are accomplished at every level by a multitude of fibers that run fore and aft within the cord, the midbrain, and beneath the cortex.

3.3 Hierarchical Organization

It has long been recognized that the entire central nervous system is a hierarchy of computational modules, starting with the sensory-motor centers in the spinal cord, to nuclei in the midbrain and basal ganglia, and finally to the sensory-motor regions of the cortex. [47, 69] At the bottom in the spinal cord, SP and BG are tightly connected forming reflex control loops. In the midbrain, tactile and proprioception signals are combined with inertial inputs from the vestibular sensors. It is now become clear that multiple regions in the cortex are also organized in hierarchical layers in both the frontal and posterior areas as illustrated in Figure 6.

⁴ An exception is the cerebellum, which hangs off of the back of the midbrain. The cerebellum is here assumed to be a special purpose computer for the kinematic and dynamic transformations that are required for coordinated muscle control. It receives motor commands from above, proprioceptive feedback from below, and inertial data from the vestibular system. It outputs signals that coordinate precise high speed movements. The cerebellum provides a massive communication pathway between sensing and acting at the midbrain level. Evolutionary newer parts of the cerebellum provide these same types of computational capabilities to the cortex to support recursive estimation (in posterior cortex) and planning (in frontal cortex.)

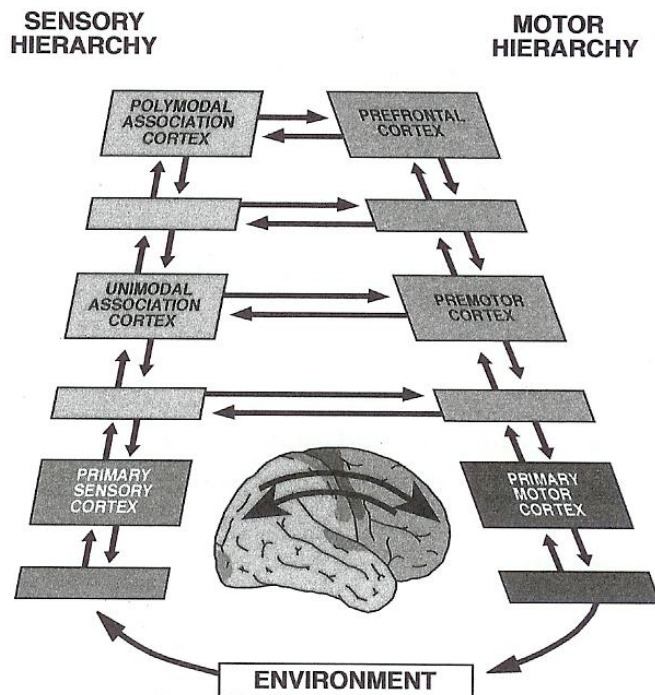


Figure 6. Hierarchical structure in the cortex. A sensory processing hierarchy in the posterior cortex, and a motor hierarchy in the frontal cortex, with many two way connections between them. [31]

In the first level in the cortical hierarchy, somatosensory and motor maps are closely linked along the banks of the central sulcus. Higher levels in the somatosensory hierarchy are located further back where they merge with higher levels in the cortical hierarchy dedicated to visual space. Higher levels in the cortical hierarchy dedicated to hearing are located off to the side in the temporal lobe where they merge with higher levels in the cortical hierarchy dedicated to visual object classification and recognition.

Similar hierarchies exist in frontal cortex. The lowest echelon in the BG cortical hierarchy is the primary motor cortex which produces desired sequences of motions of small muscle groups in limbs and digits. Higher echelons in the premotor cortex, supplemental motor cortex, frontal eye fields, and prefrontal cortex are dedicated to higher levels of planning for simple behaviors, tactical behaviors, focus of attention, and strategic behaviors respectively.

Although the brain is fundamentally a hierarchy, it is by no means a pyramid.⁵ The number of neurons at the top of the brain's hierarchy far exceeds the number at the bottom. The hierarchy in the brain is also not a tree, it is a layered graph. There are massive horizontal communication pathways within hierarchical levels, as well as

⁵ It is shaped more like an inverted trapezoid.

ascending and descending pathways between levels, some of which bypass levels. Many of these pathways form loops, many of which are nested within longer loops.

In the frontal cortex, the cortical surface is partitioned into regions that make decisions, decompose tasks, and generate plans at multiple hierarchical echelons, and in multiple coordinate frames. Outputs from higher echelons become inputs to lower echelons. The prefrontal cortex is where high-level decisions are made, long-term plans are generated, and behavioral priorities are established.

Plans generated in prefrontal cortex flow down the behavior generating hierarchy causing computational units in the frontal eye fields to direct the eyes toward regions of interest on the egosphere, and causing units in the premotor, supplemental motor, and primary motor cortices to issue commands to the midbrain and spinal motor centers to move the legs and feet to reach desired locations, and maneuver the arms, hands, and fingers to touch, feel, and manipulate objects of interest. At each echelon, behavioral commands specify goals and select procedures that make plans consisting of sequences of behavioral commands to lower-level echelons. At the bottom, commands to muscle fibers cause the body to move and act on the environment, and to search for and manipulate objects in pursuit of high-level goals. Status information flows up the behavior generating hierarchy so that higher echelon behavioral centers can monitor the progress of behavior as it evolves, and can modify plans when necessary to meet unexpected contingencies. At all levels, information flows horizontally in both directions between sensory processing (SP) and behavior generating (BG) hierarchies.

In the posterior cortex, the cortical surface is partitioned into regions that represent the sensory egospace multiple times, at multiple levels in the SP hierarchy, and in multiple coordinate frames. These regions are interconnected such that, outputs from lower level regions become inputs to the upper level regions, sometimes from one level to the next, and sometimes bypassing one or more levels. Our model assumes that sensory signals flowing up the SP hierarchy are windowed and segmented into patterns, grouped into entities and events, classified, and linked to other sensory modalities in situations and episodes. Sensory priorities flow down the SP hierarchy from high-level SP centers to lower level SP centers to focus attention on objects that are important to current goals. Sensory priorities cause computational units in the SP hierarchy to select appropriate processing algorithms for windowing, segmentation, grouping, filtering, and classification of sensory signals.

In short, information flows both up and down hierarchies in both frontal and posterior cortex. Information also flows horizontally, both within local neighborhoods, and over long distances between sensory and motor hierarchies. [31]

Of course, Figure 6 is only a cartoon of a first approximation. There are many modules and communication pathways not shown in this diagram. For example,

there are many links to the limbic system at all levels. But the main thrust of Figure 6 is accurate. Everything is not connected to everything else. There are many clearly identifiable subdivisions in the cortex that perform specific functions. These are arranged in hierarchical layers that are cross-coupled in circuits with multiple loops that perform extremely complex computational functions.

The cortex in posterior brain is not only partitioned into a hierarchy of levels, but into separate hierarchies that process input from different sensory modalities. The lower levels of the visual processing hierarchy are located in the occipital cortex. The primary somatosensory processing hierarchy is located in the anterior parietal cortex. The acoustic processing hierarchy is located in the superior temporal cortex. Outputs from these three unimodal hierarchies are combined in two bimodal association areas:

- a) visual and somatosensory representations of egospace are combined in the posterior parietal cortex. This information represents where objects are and how they are moving. This information is primarily used for controlling behavioral tasks.
- b) visual entities are merged with acoustic events in the inferior and anterior temporal cortex to form associations between visually observed objects (including written words) and acoustically observed sounds (including spoken words.) This information identifies what objects are and what classes they belong to.

Output from the two bimodal association areas are merged in the region surrounding the junction of the occipital, temporal, and parietal cortices (a.k.a. Wernike's area.) [62] Output from this region travels to Broca's area where speech is generated. It also goes to the hippocampus which controls whether short-term memories are consolidated into long-term memory. It also travels to the amygdala and other parts of the limbic system that assign emotional values (e.g., worth, importance, cost, risk, and benefit) to objects, events, situations, and episodes. Output from all levels of the sensory processing hierarchies travels to various areas in the frontal cortex where behavior is generated.

Finally, there are many loops between the cortex and thalamus, as well as between cortical columns within the same neighborhood at the same level, and between different levels within the same receptive field or field of influence. In the frontal cortex, these loops pass through the basal ganglia before entering the thalamus on the way back to the cortex. These frontal loops are believed to sequence behavioral activity. In both frontal and posterior cortex, some loops also pass through the cerebellum and various modules in the limbic system.

From this computational architecture emerges the richness of human experience in vision, hearing, feeling, perception, cognition, reasoning, decision-making, planning, and control of behavior that not only reacts to current sensory input, but reasons about the past, anticipates the future, sets long range goals, makes plans,

develops tactics, and performs actions designed to achieve the needs and desires and high level goals of the self.

Our model assumes that the overall result of the sensory processing hierarchy is to generate and maintain a rich, dynamic, and colorful internal spatial-temporal model of the world that is overlaid with many levels of meaning and emotional value. Top-to-bottom, high level symbolic representations with syntax and semantics are linked all the way down the hierarchy from objects and relationships in situations and episodes at the top to individual signals and pixels in sensory images at the bottom. Bottom-to-top, low level pixel representations are segmented into patterns, objects, and events with attributes and state that are linked all the way up the hierarchy to relationships and classes with attributes that can be inherited and behaviors that can be predicted. Signal from sensor arrays are dynamically linked to objects, agents, relationships, situations, and episodes that have spatial and temporal continuity. Entities are perceived to move about the environment in ways that are subject to laws of physics, i.e., their attributes and state can be tracked and predicted. Agents are classified into categories, and are assigned attributes inherited from their class prototypes.

We argue that our internal representation must be at least as rich, colorful, and dynamic as the world that we consciously see, hear, feel, smell, and taste, otherwise we could not experience these qualities. However, conscious experience is only a subset of the world model representation that enables the brain to generate and control successful behavior. Underlying conscious experience is an unconscious representation of the world in terms of pixels, signals, attribute-vectors, state-variables, and pointers that reside in computational modules that are interconnected by a network of communication pathways that include feedback loops and feedforward predictions. Working together, these computational modules and representational structures generate the fundamental processes of perception: i.e., focus of attention, segmentation and grouping, computation of group attributes, classification, and establishment of relationships. These processes window and mask incoming signals, detect spatial and temporal patterns, compute properties of those patterns, assign patterns to classes, and link patterns to other patterns in large tapestries of situations and episodes. Many, perhaps even most, of these processes are unconscious. Only their results rise to the level of conscious experience.

4. Structure in the Cortex

The cortex is a massively parallel structure. The human neocortex is a thin sheet of computational units about 2000 cm^2 (2.2 ft^2) in area and about 3 mm thick. [86] There are noticeable differences in cell types and structural details between different regions in the cortex. These differences enabled Brodmann and others to make charts of cortical regions. To a significant degree these different regions perform different functions and/or service different modalities. However, these anatomical differences are relatively minor in comparison to the many ways in

which the cortex is uniform throughout its extent. Nearly everywhere it consists of six layers of neurons arranged in columns oriented perpendicular to the surface of the cortex. The six layers of the cortex contain different kinds of neurons, axons, dendrites, and synapses that are specialized in form and function.

4.1 Cortical Columns

The cortex is tessellated into columns of neurons. There are two types of cortical columns:

- 1) microcolumns contain 100 to 250 neurons. These are only 30 to 50 μ in diameter (scarcely more than one neuron wide) and about 3000 μ long (i.e., the full thickness of the cortex from layer 6 to layer 1)
- 2) hypercolumns (a.k.a. columns) contain 100+ microcolumns in a bundle. These are on the order of 500 μ in diameter and also about 3000 μ long.

A typical hypercolumn occupies about 0.2 mm² of cortical real estate. Thus, there are about a million hypercolumns in the cortex, and more than 100 million microcolumns. Each of the hypercolumns in the cortex is serviced by underlying thalamic nuclei that are connected to the cortex through looping communication pathways.

It has long been suspected that these cortical columns perform some kind of computational function in the cortex. [68, 69, 44, 15] The degree of uniformity of structure in the cortex and the corticothalamic loops suggests that fundamental computational mechanisms may be similar throughout the cortex despite the differences in functional processes that are performed in different regions.

4.2. Communications within the Brain

Communications within and between regions in the brain take place via two fundamentally different types of neurons: drivers and modulators. [80]

4.2.1 Drivers

Driver neurons are often called relay neurons because they receive topologically organized data from input drivers and transmit topologically organized data to target drivers. Driver inputs are characterized by relatively small receptive fields, and their output axons cover small fields of influence. Axons from driver neurons typically terminate in dense proximal synaptic connections on a few target neurons in a limited neighborhood. The size of these neighborhoods defines the receptive fields of the target neurons.

In the visual and somatosensory systems, axons from driver neurons convey information in the form of images or maps (i.e., arrays) in sensory egospace. Driver neurons in both the thalamus and cortex define maps of egospace, and these maps are repeated many times at different levels in both sensory and motor hierarchies.

In the acoustic processing system, driver neurons convey information roughly in the form of spectrographs. These are essentially maps of spectral energy vs. time.

Driver signals flow up the sensory processing hierarchy in the form of arrays of attribute and state vectors. Driver neurons and their axons are topographically mapped from arrays of sensors in the retina, the skin, and the cochlea onto arrays of driver neurons in the thalamus and then in the cortex. There are a number of these maps (or arrays) that are registered with the sensory egosphere in the posterior cortex and in underlying thalamic nuclei. These maps are connected in a hierarchy of sensory processing levels. Within each level, there are multiple map representations of attributes of entities and events. For example, Van Essen and his colleagues [29] have identified at least 32 separate representations of the visual field in the macaque monkey cortex. These are arranged in 12 hierarchical levels with many two-way modulator communication pathways both within and between levels. There are corresponding multiple map representations in the thalamus. [48]

Driver output from each SP level travels upward to higher SP levels and horizontally to corresponding echelons of the BG hierarchy in the frontal cortex. At each hierarchical level, receptive fields grow larger and overlap each other, but topology is preserved until near the top where receptive fields cover the entire egosphere. Output from the top of the SP hierarchy travels to the hippocampus where decisions are made as to which entities and events are worthy of storage in long-term memory.

Topology and registration is obviously important to the computational processes in the brain because the neural circuitry goes to considerable lengths to preserve them, even though in many cases this requires that the maps be inverted or reflected between levels in the hierarchy in order to preserve the topological relationships.

4.2.2 Representation of Driver Information

Each driver neuron conveys a signal that represents the value of an attribute that can be represented by a time-dependent real scalar variable

$$s_k(t)$$

where $s_k(t)$ = value of the attribute represented by the k^{th} neuron at time t
 $t = 0$ is the present instant
 $t < 0$ is in the past
 $t > 0$ is in the future

For example, a driver axon in the optic nerve may convey a signal that represents the intensity of excitation of a particular rod or cone in the retina. Another driver axon may convey a signal that represents the spatial or temporal derivative of intensity or color within the same or another pixel on the retina.

Driver neurons are typically organized as arrays (i.e., images or maps) over a surface such as the retina, the skin, or the cochlea. A pixel is a resolution element of an image. For example, the surface of the retina is tessellated into an array of pixels such that each pixel represents a tiny patch of real estate on the retina. The area enclosed by a retinal pixel typically contains more than one type of sensor. In general, a pixel contains a set of sensory neurons and processing neurons that generate signals that are attributes of the pixel. Thus, each retinal pixel can be represented by a vector consisting of signals from red, green, and blue cones, intensity rods, and spatial and temporal gradients of color and intensity. An array of pixels can be represented as a time dependent array of vectors of the form:

$$\mathbf{S}(u, v, j, t)$$

where

u = horizontal index of the pixel in the array

v = vertical index of the pixel in the array

j = index of the element in the pixel vector

t = time

Elements in a pixel vector may represent attributes or state-variables.

Similarly, the surface of the body is tessellated into an array of tactile pixels. A tactile pixel represents a patch of real estate on the body surface. A single tactile pixel on the skin may host several different types of sensors (e.g., that measure pressure, touch, vibration, temperature, and pain.) We can then represent the entire body surface, or the entire retinal image, as a two-dimensional array of time dependent attribute vectors

$\mathbf{S}_m(u, v, j, t)$, where m is the modality of the sensory input.

The size of the pixels and the extent of the image array are defined by the density of the sensors over the surface of the retina or the skin. Pixel size may vary within the array. For visual image arrays, the size of the pixels is small in the fovea, and large in the periphery. For tactile image arrays, pixel size is small in the lips, tongue, and finger tips, and larger elsewhere.

The image arrays from vision, somatosensory, and auditory sensory fields of regard are repeated at multiple levels in the sensory processing hierarchy. Arrays of drivers at any level in the sensory processing hierarchy can be represented as a time dependent matrix of the form:

$$\mathbf{S}_m(i, u, v, j, t)$$

where

i = level of the array in the hierarchy

m = modality of the sensory array

A temporal sequence of driver arrays enables a representation of motion and temporal continuity. Temporal sequences are referenced to the present at $t = 0$. At each level in the hierarchy, temporal histories may extend back in time from $t = 0$ to a short-term memory horizon, while predicted future expectations may extend forward from $t = 0$ to a planning horizon.

4.2.3 Modulators

Modulator signals control the computational processes that operate on driver signals. Modulator neurons have relatively large receptive fields and their axons typically cover large fields of influence with low density and distal synaptic connections on driver neurons. Modulator neurons typically do not preserve topological order, or at best do so only generally.

Modulator inputs were traditionally thought to simply suppress or enhance general levels of activity of drivers in various regions of the brain. This classical view of modulators is undoubtedly correct for those modulator inputs that are concerned with general levels of alertness, e.g., those originating in midbrain reticular activating centers.

However, it seems likely that modulator inputs originating in the cortex play a more sophisticated role. For example, modulators seem particularly well suited for broadcasting state-variables or parameters to an entire array of processing units. This can enable modulators to effectively select algorithms and set priorities for array processing. Thus, it is hypothesized that cortical modulators select and control a variety of image operations such as region growing, zooming, scrolling, and coordinate transformation.

In the model presented here, it is hypothesized that modulator neurons can also act as address pointers. In the brain, every input to a neuron, or an array of neurons, produces an output. Thus, any input is effectively an address, and the output is effectively the contents of that address. Furthermore, the output of almost every neuron or array becomes input to another neuron or array. Thus, input to the first array of neurons can generate an output that includes both data and pointers to a second array of neurons where both data and still other pointers are stored. This is a hypothesis that enables a wide range of computational and representational possibilities.

For example, our model hypothesizes that modulators access information of the following kinds:

- a) **parameters that filter, enhance, or mask regions** in the driver data arrays – this information enables focus of attention and segmentation.
- b) **filtered estimates of pixel, entity, or event attributes** – this information enables the storage and retrieval of an internal model of the external world.

- c) **predicted values of pixel, entity, or event attributes** – this information enables recursive estimation and predictive filtering on the internal world model.
- d) **attributes of entity or event class prototypes** – this enables classification of entities and events based on similarities between observed attributes and class prototype attributes.
- e) **procedures that operate on driver arrays** – this enables modulators to select procedures for processing of sensory information.
- f) **parameters that set decision criteria or modulate functional operations** – this enables modulators to manipulate algorithms that generate and control behavior.
- g) **pointers that define relationships** within and between entities and events – this enables modulators to make and break relationships that define situations, episodes, plans, and behaviors; that assign entities and events to classes; that link them to emotional values; that embed them in syntactic structures; and imbue them with semantic meaning.

The hypothesis that neurons can generate the neural equivalent of pointers is central to the model of computation and representation proposed here. Grouping of pixels and signals into patterns (e.g., entities and events) requires the neural equivalent of pointers. Representation of relationships between entities and events in situations and episodes requires the neural equivalent of pointers. Representation of rules, plans, behaviors, beliefs, and linguistic structure requires the neural equivalent of pointers. Therefore, we make the engineering hypothesis that the functional equivalent of pointers exists in the brain. Justification for this hypothesis is contained in the remainder of this section and the next.

It has been observed that in the cortico-thalamic loop between V1 and the lateral geniculate nucleus, the number of modulator axons exceeds the number of driver neurons by a ratio of 20:1. [34, 80] This is consistent with the hypothesis that driver axons convey data in the form of vectors or arrays of attribute values, while modulator axons convey information in the form of broadcast variables and address pointers that access stored information. Jones [48] uses the terms “contents” and “context” in the place of “data” and “broadcast variables and pointers” respectively, but the meaning is the same.

Our model assumes that the brain contains many different types of pointers that are defined by different sets of modulator axons. For example, *has-part* and *belongs-to* pointers define relationships between entities and events at higher and lower levels of the segmentation and grouping hierarchy. Bottom-to-top, they link pixels to objects and signals to events, and vice versa. *is-a-member-of* pointers link entity and event frames to class prototypes. Other types of pointers can link entities and events in to each other in graph structures that describe situations and episodes. In general, we hypothesize that pointers created by modulators are able to represent many forms of symbolic knowledge, including semantic nets, procedures, rules, grammars, tasks, skills, plans, and behaviors. We hypothesize that loops between

sets of modulator neurons in computational modules at two different locations in the brain, enable the establishment of forward and backward pointers between those locations. This is discussed in more detail in section 6.3.2 and illustrated in Figure 11.

Some modulators run between regions representing maps or images at higher and lower levels in the sensory hierarchy. We assume that descending modulator axons provide top-down perceptual assumptions and priorities needed to window regions of attention and select gestalt grouping hypotheses at each level in the sensory hierarchy. We assume that ascending modulator axons provide information regarding regions of egospace where observations fail to match predictions, and hence merit attention.

Locally, interneurons generate modulator connections that run both vertically within microcolumns, and horizontally between microcolumns within the same hypercolumn. Horizontal connections also exist between neighboring hypercolumns in the same sensory array. These modulators seem well suited to provide lateral inhibition to sharpen distinctions, and to support “winner-take-all” segmentation and grouping processes.

In summary, our model assumes that drivers communicate specific data (e.g., attributes) that are topologically organized as images or maps, while modulators broadcast state-variables and communicate addresses (e.g. pointers) that control computational processes and define contextual relationships. A detailed discussion of the neuroscience evidence for driver and modulator neurons is provided in Sherman and Guillery [80].

4.2.4 Cortical Inputs and Outputs

As mentioned above, the cortex contains six layers of neurons. Cortical layers 1, 2, 3, and 4 receive inputs. Inputs to layers 1, 2, and 3a are mostly from modulators. Inputs to layer 3b and 4 are primarily from drivers in topologically structured receptive fields. Cortical layers 3, 5, and 6 of the cortex produce outputs. Outputs from layers 3 and 6 are modulators. Outputs from layer 5 are drivers.

In the posterior cortex, driver inputs to layers 3a and 4 of the cortex convey attribute images from sensor arrays relayed through the thalamus. Driver outputs from layer 5 convey attribute images to higher order thalamic nuclei that relay them to higher levels in the cortical processing hierarchy. Some layer 5 driver axons branch and send signals out of the sensory hierarchy to corresponding and lower echelons in the behavior generation hierarchy.

More detail regarding cortical inputs and outputs is contained in Figures 10, 11, and 13 and associated text.

5. Neural Computation

It is well established that the neuron is the elemental computational device in the brain. Each neuron receives a vector of input variables, and produces a scalar output variable. The neuron consists of a set of postsynaptic synapses, a set of dendrites that may branch many times, a cell body, an axon that may branch many times, and a set of axon terminals that release neurotransmitters that act on postsynaptic synapses on another neurons (and in some cases on itself.) In short, a neuron has a set of inputs, it performs a function over those inputs, and it generates an output that travels over its axon to many places.

The synapse provides input to the neuron. Each synapse is effectively an electronic gate that is actuated by the release of neurotransmitter chemicals from a presynaptic axon terminal. There are many different kinds of synapses that respond differentially to a wide variety of neurotransmitters, each of whose effect on the postsynaptic neuron may differ in sign, magnitude, and temporal profile. There are many sophisticated chemical and mechanical activities that transpire within each synapse.

Neurons compute by integrating the influence of all their synaptic inputs. In most cases, this integration is far from a simple linear summation, as is often assumed in artificial neural net models. The contribution of each synaptic input depends on the firing rate of the presynaptic axon, the type of presynaptic neuron, the chemical form of the neurotransmitter, the transfer function of each synapse, and the relative location of each synapse on the dendritic tree or cell body of the receiving neuron.

A typical neuron has a dendritic tree that branches many times. Each dendritic branch contains many (sometimes hundreds or even thousands) of synapses from different sources that excite or inhibit each particular branch. In some cases, the dendritic branches are active. In such cases, each fork in the dendritic tree may perform a nonlinear operation on signals arriving from its two incoming branches. Dendrites may also contain glomeruli. These are tiny computational elements that are peripheral to the neuron. They perform a non-linear function on inputs from a small set of presynaptic inputs and forward the results via a dendrite to the parent neuron.

Finally, in the cell body, the neuron integrates all of its inputs. At each instant of time, depending on its integrated input and its current state, the neuron will either fire or not fire. If it fires, it produces an action potential that propagates down its axon.

The information carried by an axon is represented by patterns of action potentials, or spikes. Spike duration is about 1 ms and the minimum interspike interval is also about 1 ms, so the highest frequency of firing is about 500 Hz. Thus, the information computed by a single neuron can be represented as a time-dependent scalar variable with a frequency bandwidth of about 500 Hz. The output of a set of

neurons can be represented as a time-dependent vector. The output of an array of neurons can represent a time-varying image or map. A temporal sequence of vectors or images enables the representation of motion and temporal continuity.

The computational function performed by a single neuron can be written as:

$$p_k(t) = h_k(\mathbf{S}(j, t))$$

where

k = identifier (i.e., name or address) of the neuron

$p_k(t)$ = time varying output variable from the k -th neuron

h_k = computational function performed by the k -th neuron

$\mathbf{S}(j, t)$ = time-varying vector of input variables

j = index of input variable in the input vector

The function h_k is a single-valued function computed over the input vector $\mathbf{S}(j, t)$.

5.1 Groups of Neurons

Neurons in the brain are clustered in nuclei or group. Input to each group arrives via a set of axons. Within each group, there are typically interneurons that communicate mostly, or only, within the group. Output from these groups travel over sets of axons to other groups. For a group of neurons, both input and output can be represented as a time-dependent vector. This can be written as

$$\mathbf{P}_n(k, t) = \mathbf{H}_n(\mathbf{S}(j, t))$$

where

n = identifier (i.e., name or address) of the group

$\mathbf{P}_n(k, t)$ = time-varying output vector from n -th nuclei

\mathbf{H}_n = functional transformation performed by the n -th nuclei

$\mathbf{S}(j, t)$ = time-varying input vector

j = index of input variable in input vector

k = index of output variable in output vector

The function \mathbf{H}_n is a vector or array transformation from \mathbf{S} space to \mathbf{P} space. If \mathbf{S} is an address, \mathbf{P} is the contents of that address. If \mathbf{P} becomes an address for another \mathbf{H}_q in the q -th nuclei, then \mathbf{P} is a pointer (i.e., an indirect address.) Thus, we hypothesize that neuronal clusters are capable of many types of arithmetic and logical matrix operations.

5.2 Computation of arithmetic and logical functions

Groups of neurons effectively compute by table look-up. For every input, there is an output. The input vector is functionally equivalent to an address, and the output vector is functionally equivalent to the contents of the address. The input vector

(typically acting through a set of interneurons) activates a set of synapses on the output neurons where the output is stored.

A well known example of how groups of neurons can compute arithmetic and logical functions can be found in the cerebellar cortex. [60, 12, 11, 10, 9, 8] In the cerebellum, commands from the cortex and feedback from proprioceptive sensors merge together as mossy fiber inputs to interneurons in the granular layer of the cerebellar cortex. Mossy fibers have all the characteristics of modulator inputs. They synapse in a layer of granule cells that are roughly 100 times more numerous than mossy fibers. The granule cell layer thus transforms mossy fiber input vectors into granule cell vectors (i.e., parallel fiber vectors) in a 100 times higher dimensional vector space. Gain control feedback through Golgi cells assures that only about 1% of parallel fibers are active for any mossy fiber input. Thus, each mossy fiber input vector is transformed into a sparse parallel fiber vector that excites a sparse set (~1%) of synaptic weights on Purkinje cells. This is analogous to an electronic computer address decoder that takes an input vector in the form of an N-bit binary address and transforms it into a 2^N dimensional sparse vector with a single non-zero element. The mossy fiber address is effectively decoded into a sparse parallel fiber vector that accesses a sparse set of Purkinji synapses where the contents of the address are stored.

For each mossy fiber input, each Purkinje neuron generates an output. This can be represented as a single valued function $p_k(t) = h_k(\mathbf{S}(j, t))$. A set of Purkinje cells generates a time-dependent vector $\mathbf{P}_n(k, t) = H_n(\mathbf{S}(j, t))$. Thus, the cerebellar cortex generates a vector transformation from input space to output space.

Because mossy fibers are broadly tuned to input variables, and a number of parallel fibers are active for any mossy fiber input vector, the table look-up function generalizes over local neighborhoods of input space. The size of these neighborhoods of generalization can be dynamically controlled by the Golgi negative feedback variable. For more details, see [12, 10, 9].

The functional relationship between the input \mathbf{S} and the output \mathbf{P} in the cerebellum can be modified (i.e., learned) via climbing fibers that convey error signals to Purkinje cells. When the \mathbf{P} vector differs from the desired $\underline{\mathbf{P}}$ vector for a particular mossy fiber input, the climbing fibers punish those Purkinje spines that contributed to the error. This form of error correction causes the cerebellum to converge rapidly to the desired control function necessary to produce the desired behavioral movement for each command-feedback condition.

Thus, a group of neurons such as a section of cerebellar cortex can compute a vector transformation from input space to output space. For example, we hypothesize that a computational module such as the cerebellum might perform a kinematic forward or inverse Jacobian transformation from wrist coordinates to joint coordinates, or from retinal coordinates to inertial coordinates, or from self

coordinates to world coordinates, or object coordinates. And these transformations can be modified and tuned through error correction learning.

It is widely believed that the phylogenetically older regions of the cerebellum provide kinematic and dynamic transformations required for dynamic coordinated control of muscles. It can be hypothesized that the newer regions of the cerebellum provide kinematic and dynamic transformations required for planning future behavior. These newer regions of the cerebellum appear to be part of the planning loops that include the motor cortex, striatum, pallidum, and thalamus. Other portions of the neocerebellum may be integrated into recursive estimation loops that involve the occipital, parietal, and temporal cortices and the thalamus.

The computational structure of the cerebellum illustrates four important principles of neural computation:

1. Input to neural modules can address memory locations with global specificity but local generalization.
2. Interneurons can act as address decoders that transform input addresses into high dimensional sparse vectors that access specific sets of memory locations.
3. Sparse access vectors provide global specificity that enables neural modules to compute vector transformations and other non-linear mathematical and logical functions by table look-up.
4. Sparse access vectors, combined with broad tuning on input variables, provide generalization over local regions of address space.

5.3 Arrays of neurons.

In the cortex, neurons are arranged in 2D arrays of computational modules consisting of cortical columns and their associated subcortical nuclei. Input to and output from an array of computational modules can be represented as a time-dependent array of vectors. This can be written as:

$$\mathbf{P}_a(r, s, k, t) = \mathbf{H}_a(\mathbf{S}(u, v, j, t))$$

where

$\mathbf{P}_a(r, s, k, t)$ is an output array of vectors

(r, s) = horizontal and vertical indices in the output array

k = index in the output vector at location (r, s) at time t

and

$\mathbf{S}(u, v, j, t)$ is in input array of vectors where

(u, v) = horizontal and vertical indices in the input array

j = index in the input vector at location (u, v) at time t

\mathbf{H}_a is a matrix transformation from \mathbf{S} to \mathbf{P}_a

a = identifier of the computational module that performs the \mathbf{H}_a transformation

Arrays of cortical computational units are often layered in hierarchies. This can be written as:

$$\mathbf{P}_a(p, r, s, k, t) = \mathbf{H}_a(\mathbf{S}(i, u, v, j, t))$$

where

a = identifier of the computational unit

i = index of the hierarchical level from which the input originates

p = index of the hierarchical level in which the computational array resides

It is well known that the visual cortex consists of many arrays of cortical columns that are topological maps of the visual field of regard. These maps are distorted due to the non-uniform resolution of pixels over the field of view, but they are topographically uniform in that neighboring columns in the visual cortex represent neighboring pixels in the retina.

It is also well known that these arrays are layered in hierarchical levels, albeit with many horizontal connections within levels, and many communication pathways that skip levels.

5.4 Loops

The existence of loops in the nervous system is widely acknowledged, but seldom appreciated. When a neuron or a computational unit contains a loop from the output back to the input, it becomes endowed with a whole new level of capabilities – and it inherits a new set of problems. New problems include the potential for oscillation, saturation, and other forms of instability. New capabilities include the ability to generate frequencies and rhythms, and to perform temporal analysis, such as time differential and/or integral operations on input vectors and arrays. This enables groups of neurons to generate real-time control solutions to own body movements and environmental dynamics.

Loops between sets of modulator neurons in computational modules at two different locations in the brain, enable the establishment of forward and backward pointers between those two locations. An illustration of how this might be accomplished is illustrated in Figure 11.

Loops provide feedback paths for state-variables. This enables a neural module to act as a finite-state automaton (fsa). A fsa can generate sequential strings, lists, graphs, and grammars. A fsa can perform algorithms, store and execute rules, generate plans, and control programs. A fsa can provide short-term memory in the form of a delay line or queue. Turing [87] proved that a fsa with memory is equivalent to a general purpose computer. Arrays of fsa's with memory and communications can therefore perform as a massively parallel network of general purpose computers.

A hierarchy of fsa's can generate goal-seeking sensory-interactive behaviors. Many robot control systems use fsa's as computational mechanisms. [8, 3, 20, 21, 58]

It is hypothesized here that hierarchies of arrays of neural computational modules acting as fsa's enable the posterior brain to parse temporal sequences and perceive temporal continuity in a world filled with moving objects, evolving situations, and extended episodes. It is also hypothesized that hierarchies of arrays of fsa's enable the frontal brain to generate complex behaviors such as hunting, foraging, telling stories, playing musical instruments, and performing routine daily tasks of going to work or school, avoiding danger, and interacting with others in social situations.

7.5 A Hierarchy of Receptive Fields

Connections between driver neurons at different hierarchical levels are not point-to-point, but region-to-point, or point-to-region. Each driver neuron at level(i) in the sensory processing hierarchy receives input from a local group of driver neurons from computational modules at level(i - 1); and each driver at level(i) sends its output to a local group of drivers in computational modules at level(i+1). This produces the phenomena of receptive fields where, at each level of the hierarchy, the receptive field of a computational unit consists of a group of computational units in a compact neighborhood from the next lower level. The receptive field is the region from which a computational unit receives driver input from a lower level. These receptive fields are defined by anatomy, but we assume that driver signals within receptive fields can be windowed or masked dynamically by modulator inputs. [48]

Figure 7 illustrates an array of computational units at level(i), one of which has a 10 x 10 receptive field of computational units at level(i-1).

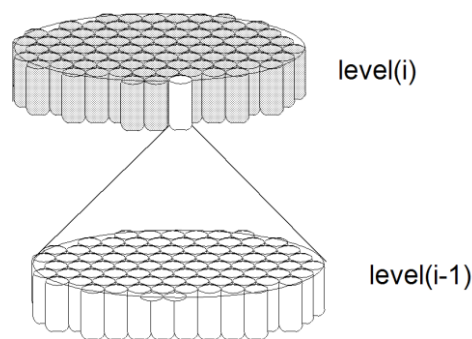


Figure 7. An array of computational units at level(i), one of which has a receptive field at level(i-1).

Figure 8 shows three levels in a hierarchy of computational units and their receptive fields.

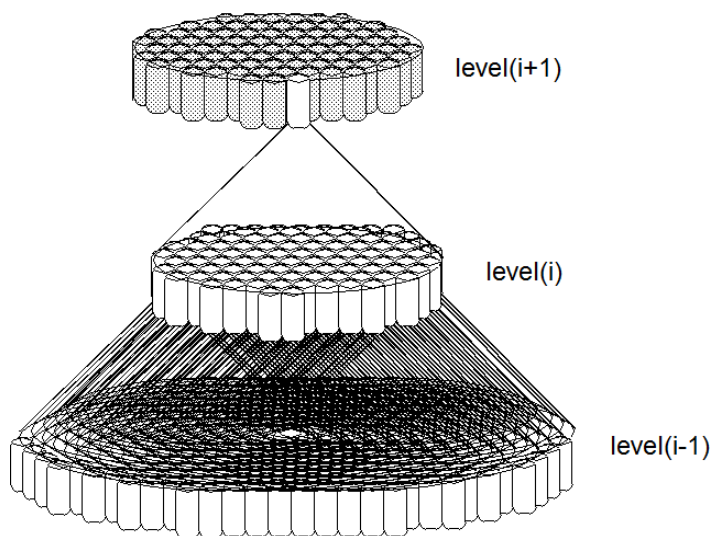


Figure 8. Three levels of a computational hierarchy.

Note in Figure 8 that a single computational unit at level(i-1) has a field of influence in level(i) that is analogous to the receptive field of a unit at level(i+1) in level(i). Receptive fields and fields of influence are defined by the cloud of synaptic contacts that a driver neuron at one level makes with other driver neurons at another level. Receptive fields and fields of influence are characterized by their spatial extent, density of synaptic contacts, and strength of synaptic contacts.

As can be seen from Figure 8, receptive fields grow progressively larger at higher levels in terms of area on the egosphere. At the bottom of the sensory hierarchy, receptive fields consist of a single pixel. Near the top, receptive fields may encompass the entire egosphere. It should be noted that topology is preserved throughout the hierarchy, even though the size of the receptive field grows large at the top.

In the frontal cortex, receptive fields and fields of influence are inverted (i.e., driver signals flow down the behavior generation hierarchy in the form of commands and priorities.) Thus, a computational unit at echelon(i) in the frontal cortex has a receptive field in echelon(i+1), and a field of influence in echelon(i-1).

6. Cortical Computational Units (CCUs)

The fundamental engineering hypothesis underlying the model of computation and representation proposed here is that each cortical hypercolumn together with the subcortical nuclei with which it is interconnected form a *Cortical Computational Unit* (CCU.)

6.1 Internal structure of CCUs in Posterior Cortex

Our model hypothesizes that in posterior cortex, each CCU contains three parts:

- 1) An abstract data structure called a CCUframe that represents properties of an entity or event. Each CCUframe:
 - a) has a physical location with a name or address where it can be accessed
 - b) contains a slot for membership criteria (that may be variable)
 - c) contains slots for attributes and state-variables
 - d) contains slots for pointers

Slots in the each CCUframe are instantiated by sets of neurons within the cortical column. The contents of the slots are defined by firing patterns on those neurons.

- 2) A set of computational processes that are able to:
 - a) segment spatial patterns of pixel (or subentity) attributes within a spatial receptive field into entities with associated CCUentityframes, and/or
 - b) segment temporal patterns of signals (or subevent) attributes within a temporal receptive field into events with associated CCUeventframes
 - c) compute estimated entity or event attributes and state-variables and store them in CCUframe slots for recursive estimation
 - d) compute predicted entity or event attributes and state-variables and store them in CCUframe slots for simulation and planning
 - e) set *belongs-to* pointers in CCUframe slots to higher level CCUs
 - f) set *has-part* pointers in CCUframe slots to lower level CCUs (or pixels)
 - g) compare entity or event attributes in CCUframe slots with class prototype attributes, and when a match occurs, set *is-a-member-of* pointers to class prototypes

These computational processes are implemented by neural circuits in the cortical hypercolumn and cortico-thalamic loop.

- 3) A set of computational processors (e.g., synapses, neurons, and circuits in microcolumns, hypercolumns, and cortico-thalamic loops) that implement the above processes and data structures.

We assume that this neural circuitry is sufficiently complex to mimic any mathematical or logical procedure that can be implemented by a neural fsa.

In our model, CCUs in the posterior cortex are organized into hierarchies of CCU arrays that transform pixels and signals carried by drivers into segmented images, maps, entities, and events that are linked together in situations and episodes. Modulators select parameters, modify processing algorithms, and provide pointers that define relationships linking these various representations together and overlaying them with meaning and emotional value.

A sketch of the internal structure of the hypothesized posterior CCU is shown in Figure 9.

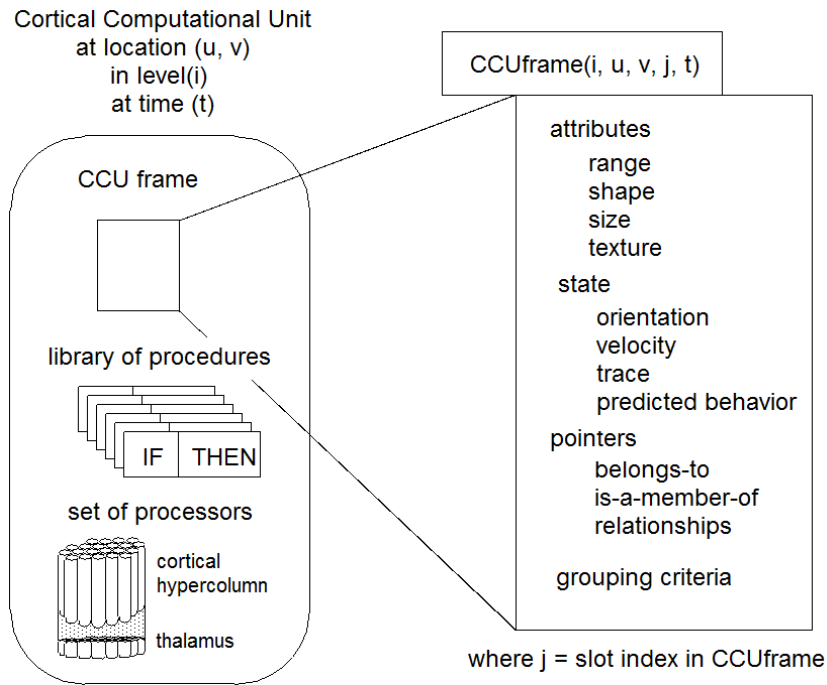


Figure 9. Internal structure of a Cortical Computational Unit (CCU) in posterior cortex consisting of an entity frame, a set of procedures for maintaining the frame, and a set of processors for implementing the procedures. Our theory assumes that this CCU is the computational equivalent of a cortical hypercolumn together with its thalamic nuclei.

Each CCU represents an entity or event (i.e., a spatial and/or temporal pattern) that occurs within its receptive field. Each CCU contains a data structure (i.e., an entity or event CCUframe) that has slots for membership criteria, pointers, attributes, and state-variables. These characterize the entity or event represented by the CCU and describe its relationships to other CCUs. Each CCU also contains computational processes that build and maintain the CCUframe data structure.

The data structure for entity and event frames in CCUs representing hypercolumns in the cortex can be represented as a matrix

$$\text{CCUframe}(i, u, v, j, t)$$

where

i = level index in the sensory processing hierarchy

u = row index of the CCU in the cortical array at level(i)

v = column index of the CCU in the cortical array at level(i)

j = slot index in the CCUframe

t = time

An input/output diagram of a typical CCU is shown in Figure 10.

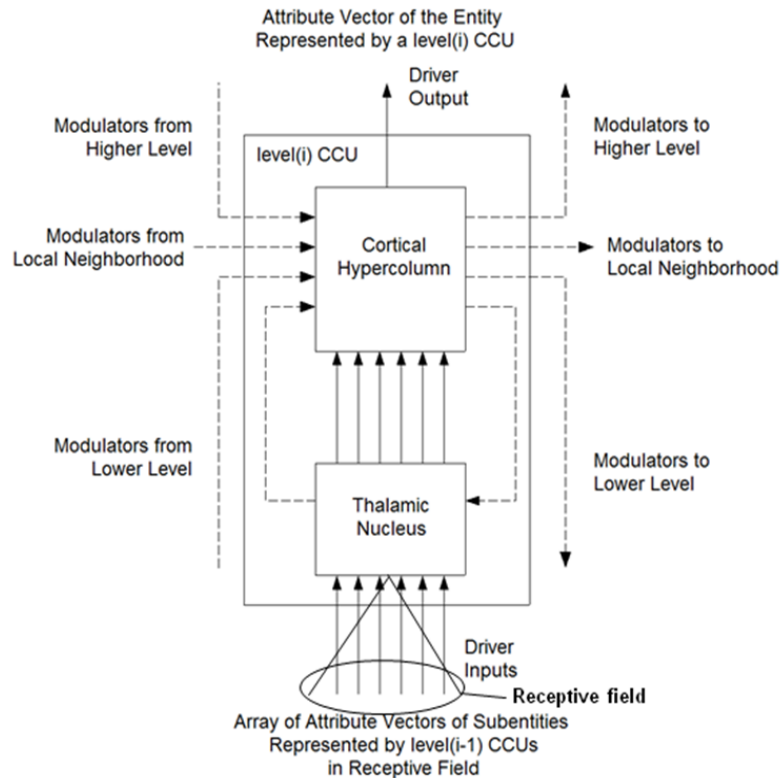


Figure 10. Inputs and outputs of a Cortical Computational Unit (CCU) in posterior cortex.

Driver inputs convey attributes and state of entity or event frames from lower-level CCUs (or pixels) in the input receptive field.

Driver output conveys attributes and state of the entity or event represented in the local CCU to higher-level CCUs in the output field of influence.

Modulator inputs from above convey commands, priorities, and pointers from above that select processing algorithms, generate masks and windows, provide prioritized list of entities of attention, suggest gestalt grouping hypotheses, and set *belongs-to* pointers in the local CCU.

Modulator outputs to above convey status of processing results to higher-level CCUs, set alerts regarding regions of the field of view that require attention, and set *has-part* pointers in level above.

Modulator inputs from below convey status of processing results from lower-level CCUs, set alerts regarding regions of the field of view that require attention, and set *has-part* pointers in the local CCU.

Modulator outputs to below convey commands and priorities to lower-level CCUs that select processing algorithms, generate masks and windows, provide

prioritized list of entities of attention, suggest gestalt grouping hypotheses, and set *belongs-to* pointers in the level below.

Modulator inputs and outputs to and from the same level provide lateral inhibition for winner-take-all segmentation and grouping.

Modulator signals between the cortex and thalamus within CCUs provide information for windowing, segmentation, and comparing of model-based predictions with observations.

6.3 Computation in the Posterior Brain

In our model, the posterior brain is primarily occupied with perception directed toward building an internal representation of the external world that is rich, colorful, dynamic and overlaid with knowledge about objects, events, situations, episodes that are endowed with context and meaning. The task of perception is to focus attention on what is important, segment pixels and signals that belong to patterns into groups, compute attributes of the groups, and classify the groups based on their attributes. This is a complex computational task that involves attention, segmentation, grouping, recursive estimation, and classification processes that compare observed entity and event attributes with stored class prototype attributes. In our model, the brain achieves its internal perceptual model of the world through a series of small steps in many hierarchical levels of processing.

Depending on how levels are defined, there may be as many as 12 levels in the visual image processing hierarchy. [29] Rods and cones in the retina make up the first level. There are then two levels of processing for spatial and temporal gradients in the retina before ganglia cell axons enter the optic nerve on the way to the lateral geniculate nucleus of the thalamus. Some retinal fibers divert or send collaterals to the superior colliculus. This provides the first level of visual feedback from the retina to the control system that directs the gaze.

Sensory arrays in the skin (touch, pressure, vibration, temperature, and pain), proprioceptors (muscles, tendons, and joints), and cochlea (sound) also go through at least two levels of preprocessing before they reach the first thalamic gateway to the cortex. Somatosensory data from the skin and proprioceptors close feedback loops first in the spinal cord. Sensory data from the vestibular system (inertial) travels through preprocessing levels in the vestibular nuclei. Output from the vestibular nuclei combines with preprocessed somatosensory information from skin and proprioceptors in closing a second feedback loop through the midbrain motor nuclei and cerebellum. Output from the vestibular nuclei also travels to the superior colliculus where it enables the eyes to be stabilized in an inertial reference frame.

Following these preprocessing stages, visual, somatosensory, and audio data become input to three CCU arrays in the primary sensory cortices. These are labeled V1 (primary visual cortex), S1 (primary somatosensory cortex), and A1 (primary auditory cortex.) Each of these sensory modalities provides four or five levels of unimodal processing before merging with other modalities. The visual

hierarchy merges with the auditory hierarchy in the inferior temporal cortex, and with the somatosensory hierarchy in the posterior parietal cortex. Finally, the three hierarchies merge into a multimodal world model in the region where the occipital, temporal, and parietal cortices come together.

As sensory information travels through these multiple levels of processing, there are four fundamental steps that are required to transform signals from sensors into a coherent model of the world consisting of symbols (objects, events, situations, episodes) and relationships that have identity, temporal continuity, and class membership; and are overlaid with meaning and worth.

6.3.1 Focusing attention

The first step in our CCU model of sensory processing is to focus attention by enhancing inputs from regions of interest in the receptive field, and/or suppressing signals from regions of little or no interest. Focusing attention begins in the retina with lateral inhibition that sharpens gradients in the image. It continues with pointing the eyes and ears toward regions of interest on the egosphere, and reaching out to touch items of interest in the environment. The photoreceptors in the retina, and the tactile sensors in the skin are not uniformly distributed. They are highly concentrated in the fovea, lips, tongue, and finger tips, and are tightly coupled to a control system that can accurately and swiftly position them so to explore interesting regions of space. The ears are shaped to amplify sounds from preferred directions. In many animals, the ears can be actively pointed toward regions of interest.

Beyond the sensory input, computational mechanisms inside the brain are also structured to focus attention. The thalamus is called the “gateway to the cortex.” All driver information traveling upward from sensors to cortex flows through the posterior thalamus first. Thus the thalamus is in a position to focus attention on what is important, by enhancing some sensory information and attenuating others.

There is also evidence that most, if not all, driver signals flowing up sensory processing hierarchies in the cortex from one level to the next, passes through the thalamus on its way to the next higher level. [80] Thus, our CCU model assumes that focusing of attention in the thalamus may occur at every level in the sensory processing hierarchy.

6.3.2 Segmentation and Grouping

Among the most fundamental of processes in perception is the segmentation of pixels and signals into groups (i.e., entities and events). Entities are here defined as spatial patterns of pixels that extend over regions in space. Events are here defined as temporal patterns of signals that extend over intervals in time.

Segmentation is the process of separating pixels that belong to an entity (or signals that belong to an event) from those that do not. Segmentation is based on group membership criteria (a.k.a. gestalt grouping hypotheses.) Grouping is the process

of linking pixels to entities and/or signal to events. Grouping involves setting *belongs-to* pointers in the lower level entity (or event) to the higher level entity (or event), and setting *has-part* pointers in the higher level entity (or event) to the lower level entities (or events.)

Group membership criteria may be based on gestalt properties such as similarity, spatial or temporal proximity, symmetry, or continuity. For example, nearby pixels with similar attributes of color, intensity, or range may be grouped into V1 blob-entities. Nearby pixels with similar gradients of intensity or range may be grouped into V1 edge-entities.

The CCU attribute filters identify those incoming pixels that meet the membership criteria, and establish pointers that link the CCU to the pixels that belong to it. A diagram of the processes and results of segmentation and grouping is shown in Figure 11.

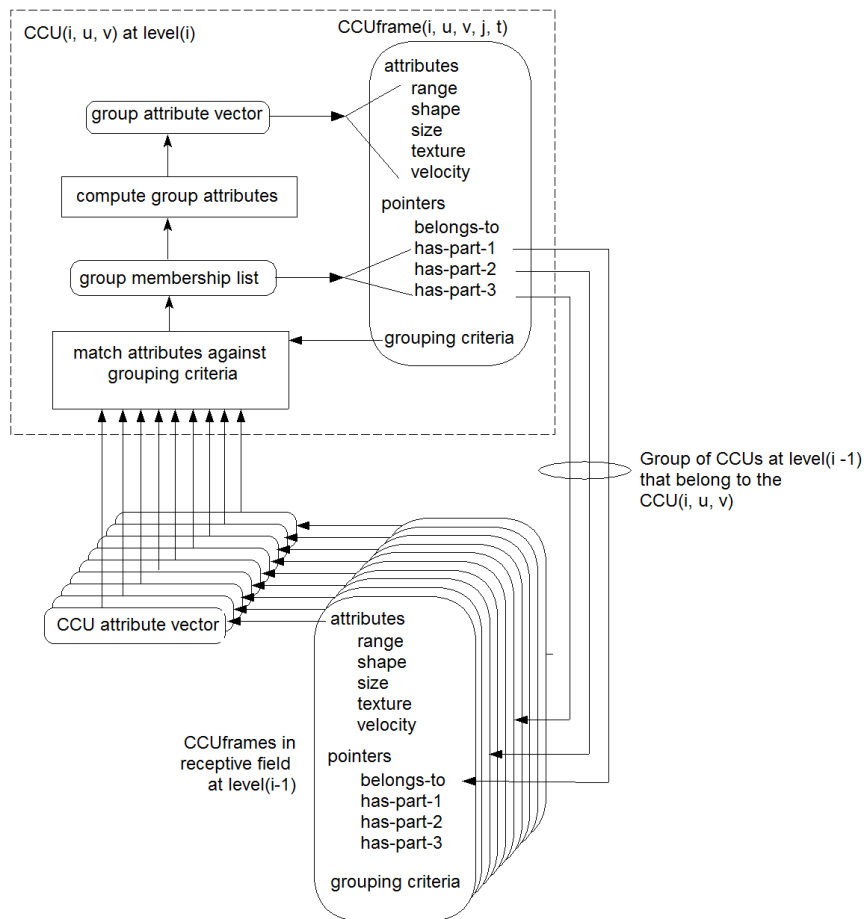


Figure 11. Segmentation and grouping processes compare lower level CCU attributes against upper level CCU grouping criteria. When attributes of lower level CCUs meet higher level grouping criteria, *has-part* pointers are set in the higher level CCU, and *belongs-to* pointers are set in the lower level CCUs.

The result of segmentation and grouping is a set of *belongs-to* and *has-part* pointers that link level(i-1) entities (or events) to level(i) entities (or events.) Each CCU in level(i) has a *belongs-to* pointer to the CCU in level(i+1) to which it belongs. It also contains a set of *has-part* pointers to the set of CCUs in level(i-1) that belong to it. The set of *has-part* pointers define a membership list for each level(i) entity (or event) that includes all the level(i-1) entities or events that belong to it. This is illustrated in Figure 12.

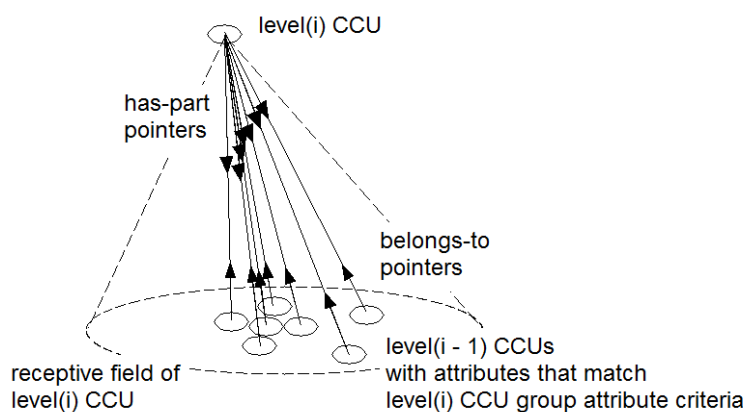


Figure 12. The result of one level of segmentation and grouping. A group of pixels or CCUs at level(i-1) are linked by *belongs-to* pointers to an CCU at level(i). A list of pointers in the level(i) CCU points back to the CCUs at level(i-1).

It might be noted that gestalt theories have fallen out of vogue in machine vision research – not so much because they are unimportant, but because they have proven difficult to implement. Yet, segmentation and grouping are prerequisites to cognitive understanding of images. Determining whether a pixel or a feature belongs to an object or the background is arguably the most fundamental of image processing functions.

Unfortunately, most sensory input is ambiguous, and any segmentation and grouping of features into objects is a hypothesis that must be tested by a recursive process of estimation and prediction. In our model, this involves building an internal model and using that model to generate predictions that can be compared with observations. It also involves computation of correlation and variance. It requires recursive updating of the model, and computation of a measure of confidence in the model. All of this is a precursor to classification. In our model, all of these processes are involved at each level in a multilevel sensory processing hierarchy.

To understand complex dynamic scenes, the vision system must be able to quickly and reliably segment objects from the background. Images of natural environments are filled with clutter, occlusions, and ambiguities. The ability to segment the

image into objects and events that have continuity in space and time is a critical visual skill, as is the ability to perceive and interpret spatial, temporal, logical, causal, syntactic, and semantic relationships.

Psychophysical data suggests that gestalt phenomena result from top-down information flow in the visual processing hierarchy. Top-down modeling has had some impact on machine vision systems, but only occasionally in a form that is biologically plausible. Adaptive Resonance Theory [37, 38, 35] and Bayesian Belief Networks [40] are among the few models that are. Both ART and Bayesian belief networks could easily be implemented in the CCU model presented here. Once in this architectural structure, they could be integrated with a large library of other image processing algorithms and applied to understanding of complex and dynamic imagery.

6.3.3 Estimation of Group Attributes and State

Once pixels are segmented into groups, attributes and state of the group can be computed. Thus, the next step in perception is for computational processes within each CCU to compute an attribute and state vector for the group. Group attributes may include size, shape, color, or texture of the group. Group state may include position, orientation, and motion. Group attribute and state variables can be filtered, and temporal strings of filtered estimates can be used to predict future values of attributes and state.

Of course, any bottom-up segmentation and grouping is only a hypothesis that needs to be tested. Predictions must be compared with observations. When variance between predictions and observations is small, estimated attributes and state can be updated and confidence in the segmentation and grouping hypothesis is strengthened. When confidence exceeds a positive threshold, the hypothesis is confirmed. This is a process of recursive estimation (a.k.a. Kalman filtering.) It is exemplified in the 4D approach to image processing pioneered by Dickmanns [27].

When variance between predictions and observations is large, confidence in the grouping hypothesis is weakened. If confidence falls below a minimum threshold, the grouping hypothesis is rejected. Another grouping hypothesis must be selected, and the process repeated.

Recursive estimation generates a “best estimate attribute vector” for the group, and a “predicted attribute vector” for each pixel (or lower level CCU.) This predicted attribute vector can also be used for generating anticipatory behavior.

6.3.4 Classification

The fourth step in perception is classification of entities and events. Classification is accomplished when a CCU compares its “best estimate” attribute vector with a stored library of class prototype attribute vectors. A match causes the CCU (and the entity or event it represents) to be assigned to the class, and the class name (i.e., pointer) be placed in the *is-a-member-of* slot in the CCUframe.

We hypothesize that class prototypes are generated from statistical summations of exemplars over a historical period. When the system is completely naive, the first exemplar becomes the prototype. From then on, the prototype is updated by each exemplar that is assigned to the class. The newly updated prototype plus the current situation are then used to generate a prediction for the next time interval. Thus each new exemplar is compared with the recently updated prototype, and any variance is used to update the next new prototype. This is a version of a Kalman filter. The prototype is the internal best estimate of the current state of the external entity or event in the external world.

At different hierarchical levels, the historical period over which exemplars are integrated is different. An engineering rule of thumb is that the length of the historical period of interest at a level is roughly equal to the planning horizon for corresponding echelon in the behavior generating hierarchy.

6.3.5 *Hierarchies of Entities and Events*

We hypothesize that the four above mentioned processes: 1) focus of attention, 2) segmentation and grouping, 3) recursive estimation of group attributes, and 4) classification are repeated at many (perhaps all) levels in the sensory processing hierarchies. At each level, each CCU receives driver inputs from a set of lower level CCUs within its receptive field. Within each CCU there are segmentation processes that enable it to determine which lower level CCUs have attributes that meet its grouping criteria. The grouping criteria may contain variables that can be selected by modulator inputs from outside the CCU, or from goals and priorities in the behavior generating parts of the brain, or from higher levels in the sensory processing hierarchy that determine what is important.

At each level of our model, lower level entities and events are segmented and grouped into higher level entities and events with attributes, state, and pointers that define class membership and relationships with other entities and events. The result is that the egospace is filled with a rich, colorful, dynamic internal representation that is segmented into entities and events with names, attributes, and state; and these are linked in situations and episodes that are overlaid with meaning and emotional values.

It should be noted that many neuroscientists and cognitive psychologists disagree with the notion of neuromorphic representations of meaningful objects and events. Neural net modelers are not accustomed to thinking in symbolic terms of objects, events, and relationships defined by pointers, graphs, and grammars. These are concepts from computer science and control theory that are not considered biologically plausible. But the human brain obviously is capable of representing meaningful situations and episodes. Otherwise humans could not conceive of, or talk about such things, or write papers and books about them. Our CCU model suggests how meaningful objects and events could emerge from the computational

capabilities of neuronal circuits when arranged in nuclei, arrays of cortical columns, and nested cortico-thalamic loops.

Of course, there is no physical evidence that meaning-based modules and pointers actually exist in the brain. However, there also is no evidence that they don't. Pointers between CCUs that provide meaning would be extremely difficult to observe, even if one were looking for them. They would consist of sparse vectors carried by thousands of modulator axons flowing between CCUs in different regions of the cortex. Felleman and van Essen [29] report 305 separate pathways between 32 cortical regions in the visual cortex of the macaque. So there are plenty of modulator axons available, and these axons must be doing something. The current wisdom is that these modulators make only more or less random connections with their target neurons. This seems unlikely. Our hypothesis is that these modulator axons encode information in the form of extremely large sparse vectors that act in meaningful ways on CCU arrays, linking them together in meaningful networks, both within and between different regions of the brain.

Figure 13 illustrates how driver and modulator signals in our CCU model flow between layers in the cortex and between levels in the sensory processing hierarchy. At each level in the sensory hierarchy, there are cortico-thalamic loops. Modulator outputs from layer 6 travel back to the thalamus from whence driver inputs originate. Modulator outputs from the thalamus return to layer 1 of the cortical region where driver inputs terminate.

In our model of posterior cortex, modulator outputs from layer 3 and 6 travel up the hierarchy to layer 1 and 4 respectively of the next higher level. Modulator outputs from layer 2 travel down the hierarchy to layer 1 of the next lower level. Loops enable modulators to set pointers that establish relationships between hypercolumns, both within and between hierarchical levels. Loops also enable modulator signals and interneuron circuits to select processing algorithms to be performed on driver signals. Modulator address vectors can effectively page memory into regions where library functions and functional parameters are stored. It should be noted that Figure 13 is a simplified diagram. Some signals skip levels, both up and down the hierarchy, and there are many horizontal data channels between neighboring modules at the same level.

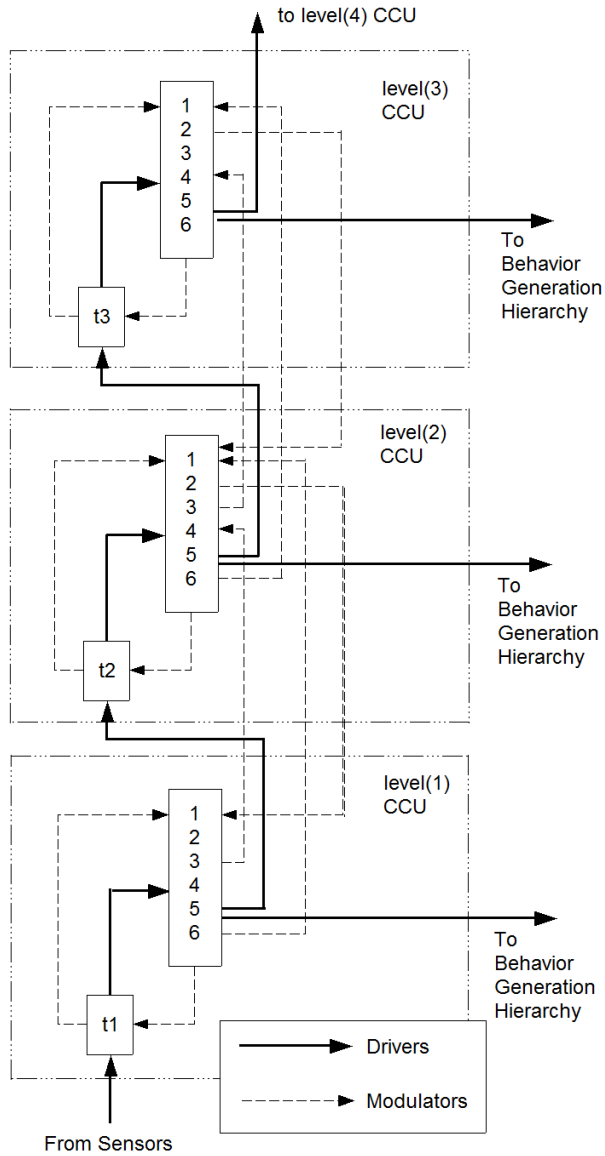


Figure 13. The numbers 1 – 6 in the boxes refer to cortical layers in cortical hypercolumns. The symbols t1, t2, and t3 refer to thalamic nuclei that relay driver information up the sensory processing hierarchy.

An example of a two level entity hierarchy resulting from two levels in the sensory processing hierarchy is shown in Figure 14.

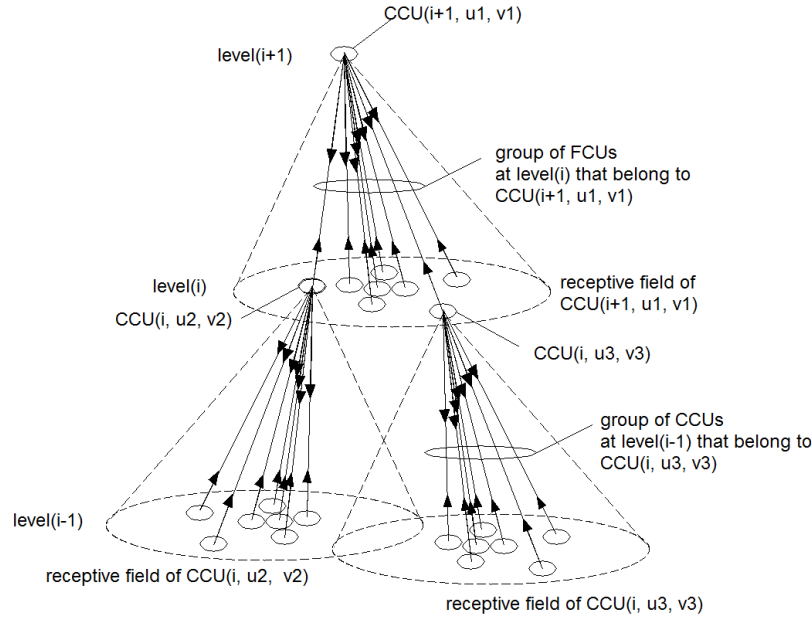


Figure 14. Two levels of segmentation and grouping of entity frames.

A comparison of Figure 14 with Figure 8 makes clear that there are two very different but related types of hierarchies in posterior cortex:

1. Receptive field hierarchies

As shown in Figure 8, receptive field hierarchies are defined by the anatomical connections between driver neurons in CCUs at different levels. Receptive field hierarchies may be modified locally by masking and windowing operations, but otherwise are relatively fixed.

2. Entity and event hierarchies

As shown in Figure 14, entity and event hierarchies are defined by links established by segmentation and grouping processes that group spatial patterns into entities and temporal patterns into events. These links are pointers, and are as enduring or fleeting as the objects and events in the world that they represent. We assume that the pointers that define entity and event hierarchies can be established or modified within a single cortico-thalamic-cortico modulator loop delay (i.e., on the order of 10 ms.)

In our CCU model, entity and event hierarchies link signals to symbols at multiple levels of resolution and abstraction. At each level, CCUframes store group attributes, state-variables, and pointers that define relationships and link entities and events to:

- parents, siblings, and children
- other entities and events in situations and episodes
- classes, lists, rules, graphs, and grammars

At each level in the receptive field hierarchy, computational processes embedded in arrays of CCUs focus attention, segment and group entities and events, compute attributes, and maintain pointers that define spatial and temporal relationships in situations and episodes. All these processes operate in parallel to enable the entire entity and event hierarchy to reconfigure top-to-bottom within the duration of a single saccade (i.e., less than 150 ms.)

6.4 CCUs in Frontal Brain

In our model of the frontal brain, CCUs have similar structure to those in posterior brain. However, in the frontal brain, CCUs are organized into hierarchies of behavior-generating and world-modeling processes that make decisions, select goals, and generate plans that drive behavior. Frontal CCU processes evaluate alternative situations and scenarios, execute rules, and react to unexpected events. Frontal CCU frame slots are populated with neurons that generate parameters and pointers that define rules, tasks, goals, priorities, plans, skills, and behaviors. At the top of the behavior generating hierarchy in prefrontal cortex, high-level goals are set and priorities established. [39] At the bottom, spinal motor neurons send commands to muscles that produce action. At every echelon, links to the limbic system provide evaluation of cost, risk, and benefit of goals, plans, and behaviors. Cortico-thalamic loops in frontal cortex are more complex than in posterior cortex in that they include the basal ganglia and cerebellum where behavioral scripts, body dynamics, and kinematic transforms are computed.

In our model, goals and plans generated in frontal CCUs are compared with observed states and historical traces maintained by posterior CCUs. Differences between goals and observations provide the basis for action. [39] Dynamics of tentative actions can be simulated in the basal ganglia and cerebellum, and expected results returned through the thalamus to the cortex to be evaluated. This is a planning loop. A hierarchy of CCU arrays in the prefrontal, premotor, supplemental motor, and primary motor cortex form a behavior generation hierarchy that transforms intentions (i.e., desired goals and priorities) into purposeful behavior. Modulators select parameters, modify processing algorithms, and provide pointers that link actions together into behaviors that are guided by priorities and constrained by values and rules of conduct.

In our model of the frontal cortex, the driver inputs to CCU arrays represent motives and goals encoded as commands flowing down the behavior hierarchy from high-level decision and planning centers toward the low-level spinal motor centers. At each behavior echelon, commands are transformed into plans of limited resolution with limited planning horizons and limited span of control. Local decisions are made, subgoals are formulated, plans for subordinate CCUs are generated, coordinated, and evaluated, tasks are decomposed, and behavior is controlled so that at the bottom millions of individual muscles work together to accomplish high level goals despite unexpected disturbances and obstacles. [5] At each echelon, driver inputs from posterior CCUs provide spatial and temporal

knowledge about attributes and state of objects and events in the external world. Modulator inputs provide context and select computational processes that operate on these drivers to formulate decisions, set priorities, generate plans, and control behavior.

6.4.1 *Decision-making*

In the biological brain, the ultimate goal is to survive and propagate. The decision of how to decomposed this goal into tactical behaviors (such as hunt for food, avoid predators, acquire a mate, care for young, and migrate or hibernate) are made based on a complex combination of needs and desires that include hunger and thirst, sexual urges, biological rhythms, hormones, emotional states of aggression or depression, motives arising from social situations, bodily health and fitness, and emotional state-variables that label what is good or bad, attractive or repulsive, feared or hoped for, and loved or hated.

There is evidence that computational value judgment modules in the limbic system have evolved from the senses of smell and taste. Olfactory and gustatory sensors provide input for the most basic of decision algorithms, e.g.,

If it smells good, taste it;
And if it tastes good, eat it;
Else don't.

It is widely held that the limbic system evolved from a decision-making system based on odor and taste into the complex system of emotions and values that guide behavior in humans. The four major structures receiving input from the olfactory system are: basal ganglia, amygdala, hippocampus, and frontal cortico-thalamic loops.

The basal ganglia are the parts of the brain where elemental movements of body, limbs, and end effectors are organized into simple behaviors. [43] The basal ganglia contains a library of behaviors that are appropriate under a variety of situations. In lower mammals, input from the olfactory system is a primary factor in the selection of specific behaviors from this library.

The amygdala is the part of the brain where emotional feelings are generated. These include rage, fear, hatred, affection, and sadness. Olfactory input to the amygdala triggers emotions that strongly influence behavioral responses. Output from the amygdala controls the hypothalamus which drives autonomic nervous system in controlling heart rate, blood pressure, and routing of blood to various parts of the body. The hypothalamus also generates drives of hunger, thirst, and sexual arousal; and causes the endocrine glands to release hormones such as testosterone, estrogen, adrenaline, and others.

The hippocampus is known to be central to memory. Our hypothesis is that the hippocampus distinguishes between what is important enough to remember, and what is not. When something important is identified, the hippocampus instructs the cortex to remember what that something is, and what is simultaneously happening in the environment. In lower mammals, olfactory input is the primary source of information regarding what is important in the environment. Even in humans, smell has a unique capacity to elicit memories with emotional content.

6.4.2 Planning

In the frontal brain, cortico-thalamic loops include the basal ganglia and cerebellum, with input from the world model in the posterior parietal and anterior temporal cortices. Our hypothesis is that the basal ganglia provide a dynamic model of the whole body, a library of behaviors, and timing parameters for sequencing behavioral activities. [75, 43] The cerebellum provides a kinematic and dynamic models of the limbs and fingers. The posterior parietal provides a rich 3D dynamic model of the external environment. The anterior temporal cortex contributes symbolic representations of objects, events, situations, and episodes. Our model assumes that the hierarchy of frontal cortico-thalamic loops enable the generation of long and complex sequences of mental activity that involve imagining possible future behavior, simulation of expected futures, and evaluation of predicted results prior to acting. Frontal cortico-thalamic loops enable the behavior generation hierarchy to anticipate future situations and take preemptive action, rather than simply reacting to what has already happened. Frontal loops make plans, evaluate the probable results, and select among different possible future behaviors at each level in the behavior generating hierarchy. The length and complexity of intelligent behaviors that can be generated depends on the number of echelons in the behavior generating hierarchy and the sophistication of computational processes and representational structures within each echelon.

From an engineering perspective, the thalamic nuclei that service the frontal cortex appear to act as gates through which simulated results of hypothesized plans flow back to the cortex to be compared with goals. Cortico-thalamic loops in the frontal cortex are where plans are generated for tasks at several echelons in the behavior generating hierarchy. At each echelon, coordinated plans for subordinate BG processes are generated to a planning horizon that is characteristic of that echelon. Timing is critical for coordination of large numbers of actuators in complex tasks. Within each echelon, goals are formulated and priorities established. These are compared against the current state of the world represented in the posterior cortex, and situation-action rules are evoked to select behavioral scripts from the library in the basal ganglia. The selected behavior can then either be executed, or a looping procedure initiated for searching the library of possible behaviors in the basal ganglia for the one with the most desirable result. If this process is implemented in many parallel loops, many alternative plans can be evaluated simultaneously, faster than real-time. For well practiced tasks, this process converges almost immediately with minimal time delay.

Our hypothesis is that the BG hierarchy in the frontal cortex starts at the top with decision-making CCUs in the cingulate cortex. These CCUs provide input goals to strategic planning CCUs in the prefrontal cortex. These generate goals for tactical planning CCUs in the supplemental motor cortex. These generate goals for simple behavior CCUs in the premotor cortex. These generate goals for movement planning CCUs in primary motor cortex. These control dynamic computation modules in the midbrain, and these control servo loops in spinal motor centers. In the brain, some fibers bypass echelons in the BG hierarchy. These presumably provide priming and feedforward information to lower echelons.

6.4.3 *Reacting*

Feedback control loops are closed at every hierarchical level in the nervous system. At each level, inputs from posterior modules provide world model knowledge, and inputs from the limbic system provide value judgment results. At the lowest level, sensory neurons interact directly with motor neurons in the spinal cord to servo force and velocity of motion in the fingers, hands, and limbs. In spinal motor nuclei, signals from proprioceptive sensors in muscles and joints enable computation of gait. At a second level, input from the eyes, ears, and vestibular system interact with the motor system in the midbrain and cerebellum to provide a sense of balance and posture. In the cortex, output from each level of sensory processing in the posterior brain interact with corresponding echelons of behavior generation in the frontal brain. Tight coupling at each level between the sensory processing and behavior generating hierarchies enables rapid reaction to stimuli from the environment. Reflex response to sensory signals begins at the bottom, long before higher levels of perception and world modeling are achieved. The role of high levels of perception is to recognize trends, and the role of high echelons of control is to select goals, set priorities, and generate plans. Higher level modules may prime lower levels so as to anticipate events and shorten stimulus-response times.

In short, our CCU model, like the 4D/RCS architecture conforms to Weng's Type-6 Markoff Decision Processor architecture. [88]

7. Learning and Memory

Contents of long-term memories are stored in synaptic strengths and numbers. These memory elements are relatively permanent, and storage typically requires many repetitions or rehearsals. Contents of short-term memories are stored in fsa's and delay line loops. These memory elements are volatile and transient, and storage occurs within milliseconds in a single event. The hippocampus is involved in the transfer of short-term memories into long-term memories.

Learning takes place in many places in the brain, and is achieved through a variety of procedures and mechanisms. Learning procedures include reinforcement learning based on pain, pleasure, and perceived success or failure; error correction learning from a teacher (e.g., an external instructor or an internal critic); and Hebbian learning based on the relative timing of pre- and post-synaptic activity. Various kinds of learning mechanisms involve a variety of complex chemical and biological processes that occur in and near synaptic junctions.

Learning is important in building an internal model of the world, and in acquiring skills and knowledge of how to perform tasks. Learning is the brain's way of programming itself to understand its environment and to acquire skills that are likely to succeed in accomplishing behavioral goals. Building modules that learn is an important aspect of reverse engineering the brain.

However, learning does not occur everywhere in the brain. It takes place at a relatively few specific sites within an *a priori* structure. The basic architecture of the brain is not learned. It is genetically determined, just as is number of fingers and toes, the location of the eyes and ears, and the arrangement of rods and cones in the retina. Learning makes only microscopic modifications within the architectural structure that gives the brain its unique capabilities.

To really understand learning, we need to first recognize the *a priori* structure in the brain. Sur and Leamey [83] show that patterning and wiring of the brain results from a combination of genetics and experience. Genesis and differentiation of cortical layers begin three weeks before birth in the ferret, and are complete for layers IV, V, and VI of the cortex at birth. The basic architecture that partitions the brain into thalamus, tectum, and occipital, temporal, parietal, and frontal cortices is established even earlier, well before any input arrives from the periphery. Thus, learning plays an important part only well after the basic architectural framework is well established. More to the point for this paper, the learning of higher level skills such as perception, cognition, and behavior occurs long after the basic architecture of the brain is in place.

The problem addressed in this paper is how the brain computes functions, how it represents information, how it transforms sensory signals into an internal model of the external world, and how it uses that model to make decisions, generate plans, decompose tasks and act appropriately. Knowledge of *how* the brain performs these functions is a prerequisite to understanding *how it learns* to perform these functions. In short, we need to understand how the brain *works* before we can fully understand how it *learns to work*. Until we have a good model of the functional architecture of the brain – what the various computational modules do, and how they are linked together in a cognitive architecture – we are unlikely to understand how skills are learned, how knowledge is acquired, structured, and stored, or how relationships between entities, events, and classes in situations and episodes are established and broken.

8. Relationship to other models

The CCU model of computation and representation presented here differs from many other models in that it is an architecture, not an algorithm or collection of algorithms. An architecture organizes the execution of algorithms, and provides communication channels between them. Only when the architecture is populated with algorithms, does it exhibit behavior. An architecture can accommodate a wide variety of algorithms, and one algorithm can often be switched out in favor of a different algorithm. Indeed, without the ability to select different algorithms for different situations, any complex system will inevitably be brittle with many catastrophic failure modes.

The CCU model presented here attempts to model the architecture of the brain as closely as possible, given the present state of knowledge regarding the detailed circuitry of the cortex and its underlying support structures. In that respect, it is similar to the Riesenhuber-Poggio (R-P) model of the visual cortex. [76] The main difference is that the CCU model assumes computational powers that are considerably more sophisticated than is commonly considered biologically plausible in the neuroscience community. The R-P model assumes only a limited set of computational functions consisting of alternating layers of S and C cells, where the S cells learn to compute Gabor functions, and the C cells learn to pool the outputs from S cells and select maxima. The R-P hierarchy can be implemented by feedforward neural nets. In contrast, the CCU model assumes computational capabilities that include the full range of mathematical, logical, and linguistic constructs that are employed in computer science and semiotics. It is obvious that these capabilities must exist somewhere in the brain, for they can be observed in human behavior. Our model assumes that they are distributed throughout the cortex in the CCU hierarchy.

A hypercolumn and its supporting subcortical nuclei contain upwards of 100,000 neurons that are connected in complex circuits that contain multiple nested loops. These are arranged in arrays and hierarchies where information flows up, down, and sideways. This elaborate circuitry must be doing something more complex than can be modeled by feedforward neural nets. We assume the circuit connections within the brain are not random, but are highly structured such that CCUs are able to compute complex functions and procedures. We assume that cortico-thalamic and cortico-cortico loops enable Markov Decision Processes such as described by Weng. [88] In short, we assume that the human brain contains the biological equivalent of many, if not all, of the computational powers of computer science, including symbolic functions, procedures, pointers, and data structures.

We assume that all the capabilities of the CCU, including the computation and representation of symbolic information are implemented by neurons, and could be modeled at the neuron and synapse level by various connectionist schemes. However, from a reverse engineering perspective, once we demonstrate that connectionist models can segment pixels into groups, and can compute group

attributes, and that group attributes have physical locations that can be addressed, we can use mathematical shorthand and build CCU models that can be described in terms of software procedures and abstract data structures.

For example, we assume that the recursive estimation mechanisms hypothesized by the CCU model, has many of the features of Perlovsky's concept of neural modeling fields (NMF) and dynamic logic. [73] NMF is a model-based neural architecture that is a multi-level, hetero-hierarchical system. At each level in NMF there are concept-models and dynamic logic that generate top-down neural signals that are compared with input, bottom-up signals. Similarly in the CCU model, stored information in the form of class prototypes are selected by task priorities to be molded into dynamic models that generate specific predictions that can be compared with the incoming sensory experience. The working models are constantly updated so as to track and predict the current instance.

The CCU hypothesis is also compatible with classification algorithms that involve appearance-based recognition algorithms. [70] We assume that objects hypothesized by higher level CCUs can generate modulator signals that elicit dynamic template images in lower level CCU arrays for matching against observed exemplars. The recursive estimation process and top-down inputs hypothesized in Figure 5 for the 4D/RCS and CCU models could enable processes such as Grossberg's Adaptive Resonance Theory [35, 37, 38], or various view-based pattern matching algorithms. [30]

Similarly for the theory of top down facilitation of visual recognition proposed by Bar et al. [17] The idea that low spatial resolution images might be processed quickly through the dorsal path via the macro cellular channel, and passed to the orbito-frontal cortex, which generates top down information to steer bottom-up visual processing in the ventral path is fully compatible with the CCU model.

The CCU model is similar in some respects to confabulation theory proposed by R. Hecht-Nielsen. [42] Both models postulate cortical computational units that involve looping interactions with the underlying thalamic nuclei. However, CCUs differ in confabulation modules several important respects. Perhaps most fundamentally, confabulation theory hypothesizes that only one operation, namely winner-take-all competition between symbols on the input to modules (a.k.a., confabulation) is the only information-processing operation involved in thought. [42] In contrast, our CCU model hypothesizes that a much richer set of mathematical, logical, and linguistic operations take place within CCUs involving both iconic and symbolic data.

At a functional level, the CCU model differs from confabulation theory in the representation of objects and attributes. Confabulation theory hypothesizes computational modules that detect attributes. Objects are then defined by patterns of links between attributes. Thus, objects are represented as diffuse networks of links between cortical areas. In contrast, in the CCU model, entities and events are

represented by CCUs that reside at particular locations in the cortex and can be addressed. In posterior cortex, each CCU detects attributes, segments and groups inputs into spatial and temporal patterns (i.e., entities and events), computes attributes and states of those patterns (e.g., size, shape, center of mass, texture, and motion), and generates pointers to other CCUs that represent other entities and events. These pointers define situations and episodes, belongs-to and has-part relationships, and link CCUs to classes and emotional values.

The CCU model also differs with confabulation theory on the size and number of units. The CCU model assumes that CCUs are isomorphic with cortical hypercolumns, which number about 1,000,000 in the human. Confabulation theory hypothesizes modules that are much larger and number only about 6000 in the cortex. [42]

9. Reverse Engineering the Brain

Reverse engineering the brain implies building machines that are functionally equivalent to the brain in their ability to perceive, think, decide, and act in a purposeful way to achieve goals in complex, dynamic, and possibly hostile environments, despite unexpected events and unanticipated obstacles, while guided by internal values and rules of conduct.

The rapid growth of knowledge about the brain, combined with recent developments in intelligent systems engineering, and continued exponential growth in computational power and memory capacity suggests that it may become possible to reverse engineer the brain in the foreseeable future.

Computational modeling of attention, perception, and memory are active areas of cognitive neuroscience research. [23, 39, 72, 37, 74] Mathematical and theoretical descriptions of neural activity have modeled many brain functions such as decision-making, information encoding, and levels of consciousness [18, 33, 85]

Recently, several large research programs in the U. S. and Europe have begun to pursue the ambitious goal of modeling the biological brain at the synaptic level. [84, 71, 28] Some efforts have focused on building supercomputer models of the cortex. [59] Models the size of small mammalian brains have been constructed. [13, 46] Other programs are pursuing a genomic brain atlas. [55]

A number of authors have suggested that a deep understanding of the mechanisms in the brain that give rise to the phenomena of mind is feasible. [51, 67, 78, 1] Some predict this understanding will be achieved within two decades, and machines with human level capabilities in perception and cognition will become widespread soon thereafter. [53, 1]

9.1 Computational Modeling

Any discussion of reverse engineering the brain should begin with a definition of the level of fidelity and resolution in the model of reality that is being considered. The human brain contains about 10^{11} neurons and on the order of 10^{14} synapses. No one has any idea of how the higher levels of perception and cognition emerge from individual neurons and synapses. This suggests that modeling the human brain at the spike and synapse level of resolution may be beyond the state of the art for some time to come, both in computational power and scientific understanding.

However, modeling at the CCU level of resolution may be feasible today. There are only about 10^6 CCUs in the brain, and significantly fewer computational modules in the subcortical, midbrain, and spinal regions of the human brain. This is a large number, but quite manageable given the current state-of-the-art for modern multicore supercomputers. The latest version of the RoadRunner supercomputer now runs at 10^{15} floating operations per second. [77] Allocating this compute power to modeling a million CCUs provides 10^7 floating operations per CCU per modeling cycle (assuming a real-time requirement of 100 cycles per second.) Thus, modeling the human brain at the CCU level of resolution seems feasible today on supercomputers, and might become economical within two decades on laptop class machines (assuming that Moore's law continues to hold during that period.)

Modeling at the CCU level enables behavior generating functions to be expressed in terms of decision-making, planning, and control processes arranged in hierarchies of control loops that link strategy and tactics to neuromuscular signals. It enables knowledge to be represented in terms of images, maps, and trajectories patterns that are segmented and grouped into entities, events, situations, and episodes with attributes and state. It enables the use of pointers that link iconic to symbolic representations that are overlaid with meaning and emotional value. It enables skills to be defined in terms of situation-action rules, Markov decision processes, and model-based control theory. It enables emotions to be modeled in terms of value judgment functions that assign worth and attractiveness to objects, and apply cost-risk-benefit analysis to actions and plans. It enables perception to be modeled in terms of focus of attention, segmentation and grouping, recursive estimation of group attributes, and classification. Examples of these modeling methods can be found in [5, 58, 88].

This suggests a coarse-fine approach to reverse engineering the brain. That is, model the whole brain at the CCU level of resolution, and model CCUs at the neuron and synapse level of resolution. [59, 16] There are on the order of 10^5 neurons in a typical cortical hypercolumn, and perhaps 10^8 synapses. These are manageable numbers from a computer modeling perspective. Markram has shown that this level of modeling is feasible with supercomputers today. [59] Once it is understood what the overall functionality of each functional module is, and what the input and output variables are, it becomes much more likely that synapse and spike level modeling will produce plausible models of whole brain functionality.

Arrays of CCUs map easily onto modern multicore computer architectures. Assume a multicore machine with 1000 cores, each of which models 1000 CCUs. Each core runs the same basic kernel on a thousand CCU models. Each CCU model has its own unique abstract data structure and set of functional processes that are called based on the current input and internal state. Upon each modeling cycle, each CCU produces its unique output which is delivered to a set of subscriber CCUs.

9.2 Communications modeling

Of course, functional modeling of computational units is only part of the problem. While it seems likely that modern supercomputers have sufficient computational power to emulate the functionality of the brain at the CCU level of resolution, communication between CCUs might turn out to be the bigger technical challenge. The CCUs need to be connected together such that outputs from CCUs in one modeling cycle become inputs to other CCUs in the next modeling cycle.

This issue needs more information from the neuroscience community. How many CCUs communicate with each other? How often? How much information is communicated? How is it routed? And most importantly, “What is the syntax (i.e., format) and semantics (i.e., meaning) of the messages?” [33]

The modeling of mouse cortex achieved recently at the IBM Almaden Research Center provides some insight into how to model communications in the brain. [13] In this project, a simulator called C2 was designed to operate on a Blue-Genie/L supercomputer with 32,768 processors. Neurons were simulated on a 1 ms clock rate, and synapses were simulated on an event-driven basis. There are about 55 million neurons and 440 billion synapses in the rat brain. The IBM team was able to simulate the entire rat brain including communications between neurons and synapses in 1/9 real time (i.e., 1 s of model time in 9 s real time.)

It should be noted that communications modeling at the CCU level requires significantly less bandwidth than communications modeling at the spike and synapse level. Information encoded as state-variables, vectors, matrices, symbols, and pointers is much denser than information encoded as strings of spikes. Thus, the required communications bandwidth is less.

Assume each CCU outputs an attribute vector of 1000 bytes and a set of 100 pointers encoded as 16-bit addresses. Also assume a modeling cycle rate of 100 times per second. The total output data rate for a single CCU then is 120,000 bytes per second. If this information is communicated through common memory, it can simply be placed in an output register to be read by each of its subscriber CCUs. Assuming a single core models 1000 CCUs, this communication rate becomes 12 gigabytes per second per core.

Fortunately, most communications between CCUs take place between neighboring CCUs within the same array in the same core. This can be modeled through on-chip shared cache memory where subscriber CCUs simply read from publisher CCU's output buffers. Communications with distant CCUs are rarer, and can be modeled through an inter-core buss. Thus, the communications requirements seem within the current state of the art.

The specific connectivity between CCUs needs to be informed by neuro-anatomical data. Receptive fields and fields of influence of both drivers and modulators need to be reflected in the structure of the publish/subscribe network. The new technique of diffusion spectrum imaging appears to be a promising approach to building a detailed map of the major communications pathways in the brain. [89] This may enable the level of detail needed to model communications in the brain.

10. Summary

The model presented here reasons about the brain from the top down. The goal is to understand both form and function. Our model suggests a computational architecture constrained by known anatomy and neurophysiology by which sensory signals can be transformed into an internal representation of the external world that is both visually rich and overlaid with meaning and purpose. It suggests how the neural structures in the brain can perceive the world as a rich, dynamic, colorful place filled with people, places, and events that can be assigned worth, and overlaid with emotional values.

In our model, neocortical hypercolumns and their associated thalamic and other subcortical support nuclei are functionally equivalent to *Cortical Computational Units* (CCUs.) Each CCU contains a CCUframe data structure plus the computational processes and processors that maintain it.

In the posterior cortex, CCUframes have slots that contain entity or event attributes, state-variables, and pointers that define relationships. Relationships include *belongs-to* and *has-part* links to higher and lower level CCUs, as well as class membership relationships, relationships that characterize situation and episodes, and more general relationships such as strings, graphs, rules, and grammars. Cortico-thalamic loops within CCUs enable windowing, segmentation, recursive estimation, and classification.

At each level in the posterior sensory processing hierarchies, images are windowed, segmented, grouped, characterized, and classified. At the bottom, pixels represent the receptive fields of sensory neurons. At the top, CCUs represent complex situations and episodes. These top-level CCUs are linked by *has-part* pointers all the way down to the pixels of immediate experience. Bottom-level pixels are linked by *belongs-to* pointers all the way up to CCUs that represent situations and episodes. At every level, links to the limbic system provide evaluation of worth,

attractiveness, and emotional value (e.g., fear, love, hate) associated with objects, events, situations, and episodes.

At each echelon in frontal cortex, CCUs have slots that define goals, priorities, tasks, objects, and agents. Pointers define plans, skills, and behaviors. Cortico-thalamic loops include the basal ganglia and cerebellum where models of body dynamics and kinematics are stored. At the top are high-level goals and priorities. At the bottom are individual motor neurons that send commands to muscle fibers. At every echelon, links to the limbic system provide evaluation of the cost, risk, and benefit of goals, plans, and actions. At every echelon, decisions are made, goals are selected, tasks are decomposed, plans are generated and evaluated, and behavior is controlled. The result is that millions of individual muscles at the bottom work together to accomplish high level goals generated at the top, despite unexpected disturbances and obstacles.

Finally, we have suggested that reverse engineering the brain at the functional module and CCU level of fidelity might be feasible in the near term with high performance sensors and super computer processing capabilities. Within two or three decades this capability might be practical on lap-top class machines.

In the future, we hope to elaborate our model to address how the brain reasons about the past, contemplates the future, makes decisions and plans, sets goals and priorities, and responds to conditions in the environment in a manner thought most likely to achieve its high-level goals.

References

- [1] J. S. Albus, Reverse Engineering the Brain, Proceedings of AAAI Fall Symposium on Biologically Inspired Cognitive Architectures, Washington, D.C. (2008)
- [2] J. S. Albus, R. Bostelman, T. Chang, T. Hong, W. Shackleford, M. Shneier, Learning in a Hierarchical Control System: 4D/RCS in the DARPA LAGR Program, *Journal of Field Robotics* 23 (2006) 975-1003.
- [3] J. S. Albus, A. J. Barbera, RCS: A cognitive architecture for intelligent multi-agent systems, *Annual Reviews in Control*, 29, Issue 1, (2005) 87-99
- [4] J. S. Albus, et al., 4D/RCS Version 2.0: A Reference Model Architecture for Unmanned Vehicle Systems, NISTIR 6910, National Institute of Standards and Technology, Gaithersburg, MD, 2002. Available on line at http://www.isd.mel.nist.gov/documents/albus/4DRCS_ver2.pdf

- [5] J. S. Albus and A. Meystel, *Engineering of Mind: An Introduction to the Science of Intelligent Systems*, Wiley, New York, 2001
- [6] J. S. Albus, *The Engineering of Mind*, *Information Sciences* 117, (1999) 1-18.
- [7] J. S. Albus, *The NIST Real-Time Control System (RCS): an application survey*, *Journal of Experimental and Theoretical Artificial Intelligence*, 9 (1997) 157-174
- [8] J. S. Albus, *Brains, Behavior, and Robotics*, BYTE/McGraw Hill, Peterborough, N.H., 1981
- [9] J. S. Albus, *A New Approach to Manipulator Control: The Cerebellar Model Articulation Controller (CMAC)*, *Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control*, September (1975) 220-227
- [10] J. S. Albus, *Data Storage in the Cerebellar Model Articulation Controller (CMAC)*, *Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control*, September (1975) 228-233
- [11] J. S. Albus, *Theoretical and Experimental Aspects of a Cerebellar Model*, Ph. D. Thesis, University of Maryland, College Park, MD, 1972
- [12] J. S. Albus, *A Theory of Cerebellar Function*, *Mathematical Biosciences*, 10, (1971) 25-61
- [13] R. Ananthanarayanan and D. Modha, *Anatomy of a Cortical Simulator*, *Supercomputing 07: Proceedings of the ACM/IEEE SC2007 Conference on High Performance Networking and Computing*, Reno, NV. November 10-16 (2007)
- [14] J. R. Anderson, *Rules of the Mind* Erlbaum, Hillsdale, NJ (1993)
- [15] H. Asanuma, *Recent developments in the study of the columnar arrangement of neurons within the motor cortex*, *Physiol. Rev.* 55 (1975) 143-156
- [16] G. Ascoli and J. Krichmar, *L-Neuron: a modeling tool for the efficient generation and parsimonious description of dendritic morphology*, *Neurocomputing*, 32-33 (2000) 1003-1011
- [17] M. Bar, K.S. Kassam, A.S. Ghuman, J. Boshyan, A.M. Schmid, A.M. Dale, M.S. Hämmäläinen, K. Marinkovic, D.L. Schacter, B.R. Rosen, and E. Halgren "Top-down facilitation of visual recognition" *Proceedings of the National Academy of Sciences USA*, vol. 103 (2006) 449-54
- [18] W. Bialek and F. Rieke, *Reliability and information transmission in spiking neurons*, *Trends in Neuroscience* 15:11 (1992) 428-34

- [19] J. A. Bornstein, The Army Ground Robotics Program, Proceedings of SPIE Volume 4715, Unmanned Ground Robotics IV, (2002)
- [20] R. Brooks, *Cambrian Intelligence: The Early History of the New AI*, MIT Press, Cambridge, MA, 1999
- [21] R. Brooks, A robust layered control system for a mobile robot, *IEEE Journal of Robotics and Automation*, 2 (1986) 14-23
- [22] R. Camden, B. Bolt, S. Schipani, J. Bornstein, T. Runyon, F. French, C. Shoemaker, A. Jacoff, A. Lytle, “Autonomous Mobility Technology Assessment Final Report,” Army Research Laboratory Technical Report ARL-TR-3471 (2005)
- [23] G. L. Chadderdon and O. Sporns, A large-scale neurocomputational model of task-oriented behavior selection and working memory in prefrontal cortex, *Journal Cognitive Neuroscience* 18 (2006) 242-57
- [24] P. S. Churchland, *Brain-Wise: Studies in Neurophilosophy*, MIT Press, Cambridge, MA (2002)
- [25] P. S. Churchland and T. J. Sejnowski, *The Computational Brain*, MIT Press, Cambridge, MA (1992)
- [26] P. Cisek, T. Drew, and J. Kalaska, (eds.) *Computational Neuroscience: Theoretical Insights into Brain Function*, 165, Elsevier (2007)
- [27] E. Dickmanns et al., The Seeing Passenger Car ‘VaMoRs-P’, International Symposium on Intelligent Vehicles ’94, Paris, October 24-26 (1994)
- [28] FACETS program <http://facets.kip.uni-heidelberg.de/public/index.html>
- [29] D. J. Felleman and D. C. Van Essen, Distributed hierarchical processing in primate visual cortex, *Cerebral Cortex*, 1 (1991) 1-47
- [30] R. Fergus, P. Perona, A. Zisserman, Weakly supervised scale-invariant learning of models for visual recognition, *International Journal of Computer Vision*, 3 (2007) 273-303
- [31] J. M. Fuster, *Cortex and Mind: Unifying Cognition*, Oxford University Press, Oxford (2003)
- [32] General Dynamics Robotic Systems website
<<http://www.gdrs.com/robotics/programs/program.asp?UniqueID=22>>

- [33] J. I. Gold and M.N. Shadlen, Banburismus and the brain: decoding the relationship between sensory stimuli, decisions, and reward, *Neuron* 36 (2002) 299-308.
- [34] R. Granger, Engines of the brain: The computational instruction set of human cognition. *AI Magazine* (2005)
- [35] S. Grossberg, How does a brain build a cognitive code? *Psychological Review*, 87 (1980) 1-51
- [36] S. Grossberg, L. Finkel, D. Field, (eds.) *Vision and Brain: How the Brain Sees*. Elsevier, London, UK (2004)
- [37] S. Grossberg, Towards a unified theory of neocortex: laminar cortical circuits for vision and cognition, In Cisek, P., Drew, T., and Kalaska, J. (eds.) *Computational Neuroscience: Theoretical Insights into Brain Function*, 165, Elsevier (2007)
- [38] S. Grossberg and T. R. Huang, ARTSCENE: A neural system for natural scene classification, *Journal of Vision*, 9 (2009) 1-19
- [39] M. E. Hasselmo, A model of prefrontal cortical mechanisms for goal-directed behavior, *Journal Cognitive Neuroscience*, 17:7 (2005) 1115-29
- [40] J. Hawkins and D. George, Hierarchical Temporal Memory Concepts, *Numenta* (2006) http://www.numenta.com/Numenta_HTM_Concepts.pdf
- [41] B. Hayes-Roth, *A Blackboard Architecture for Control*, Artificial Intelligence (1985)
- [42] R. Hecht-Nielsen, *Confabulation Theory: The Mechanism of Thought*, Springer (2007)
- [43] J. R. Hollerman, L. Tremblay, W. R. Schultz, Involvement of basal ganglia and orbitofrontal cortex in goal-directed behavior, *Progress in Brain Research* 126 (2000) 193-215
- [44] D. H. Hubel and T. N. Wiesel, Receptive fields and functional architecture of monkey striate cortex, *Journal of Physiology*, 195 (1968) 215-243
- [45] ISO 10303 Standard for the Exchange of Product model data (STEP), International Standards Organization standard for the computer-interpretable representation and exchange of industrial product data.
- [46] E. M. Izhikevich, Polychronization: Computation with spikes, *Neural Computation*, 18 (2006) 245-282

- [47] J. H. Jackson, On the anatomical and physiological localization of movements in the brain, *Lancet* 1 (1873) 234
- [48] E. Jones, *The Thalamus: Second Edition Vol.1 and 2*, Cambridge University Press, New York (2007)
- [49] E. Kandel, J. Schwartz, T. Jessell, *Essentials of Neural Science and Behavior*, McGraw-Hill, NY (1995)
- [50] C. Koch, *The Quest for Consciousness: A Neurobiological Approach*. Roberts & Co. Publishers, Englewood, CO (2004)
- [51] C. Koch and G. Tononi, Can Machines be Conscious, *IEEE Computer*, 6, June (2008) 54-59
- [52] J.J. Koenderink and A.J. van Doorn, Dynamic shape. *Biol. Cybernet.* 53 (1986) 383-396
- [53] R. Kurzweil, *The Singularity is Near, When Humans Transcend Biology*, Viking Press, NY (2005)
- [54] J. Laird, A. Newell, P. Rosenbloom, SOAR: An Architecture for General Intelligence, *Artificial Intelligence*, 33, (1987) 1-64
- [55] E. S. Lein, M. J. Hawrylycz, N. Ao, M. Ayres, *et al.*, Genome-wide atlas of gene expression in the adult mouse brain, *Nature* 445, 168, January 11 (2007)
- [56] D. Lenat, Cyc: A Large-Scale Investment in Knowledge Infrastructure, *Communications of the ACM* 38, no. 11, November (1995)
- [57] I. Lopez-Juarez, J. Corona-Casturea, M. Peria-Cabrera, K. Ordaz-Hernandez, On the design of intelligent robotic agents for assembly, *Information Sciences*, 171, 4 (2005) 377-402
- [58] R. Madhavan, E. Messina, J. Albus, (eds.) *Intelligent Vehicle Systems: A 4D/RCS Approach*, Nova, NY (2007)
- [59] H. Markram, The Blue Brain project, *Nature Reviews Neuroscience* 7, (2006) 153-160
- [60] D. Marr, A Theory of Cerebellar Cortex, *Journal of Physiology*, (London), 202 (1969) 437-470
- [61] W. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *Bulletin of Mathematical Biophysics* 5 (1943) 115-133

- [62] M. M. Mesulam, From sensation to cognition, *Brain* 121 (1998) 1013-1052
- [63] A. M. Meystel and J. S. Albus, *Intelligent Systems: Architecture, Design, and Control*, John Wiley & Sons, Wiley Series on Intelligent Systems, NY (2002)
- [64] A. D. Milner and M. A. Goodale, *The Visual Brain in Action*, Oxford University Press, Oxford (2006)
- [65] M. Minsky, *The Society of Mind*, Simon & Shuster, NY (1985)
- [66] M. Minsky, *The Emotion Machine*, Simon & Shuster, NY (2006)
- [67] H. Moravec, *Mind Children: The Future of Robot and Human Intelligence*, Harvard University Press, Cambridge, MA (1998)
- [68] V. B. Mountcastle, The columnar organization of the neocortex, *Brain* 120 (Pt. 4) (1997) 701-722
- [69] V. B. Mountcastle, Modality and topographic properties of single neurons of cat's somatic sensory cortex, *J. Neurophysiol.* 20 (1957) 408-434
- [70] H. Murse and S. Nayar, Visual learning and recognition of 3-D objects from appearance. *International Journal of Computer Vision* 14 (1995) 5-24
- [71] NEOVISION2, DARPA
https://www.fbo.gov/index?s=opportunity&mode=form&id=36a69675c012ce8c6ec4cccbdc36767b&tab=core&_cview=0
- [72] R. C. O'Reilly, Biologically-Based Computational Models of High-Level Cognition, *Science*, 314, October 6 (2006) 91-94
- [73] L. I. Perlovsky, Cognitive high level information fusion, *Information Science*, 177 (2007) 2099-2118
- [74] T. A. Portocello and R. B. Velloti, (eds.) *Visual Cortex: New Research*, NOVA Science Publishers, NY (2008)
- [75] S. M. Rao, D. Harrington, A. W. Mayer, The evolution of brain activation during temporal processing, *Nature Neuroscience* 4:3 (2001) 317 – 323
- [76] M. Riesenhuber, T. Poggio, Neural Mechanisms of object recognition. *Curr Opin Neurobiology* 12 (2002) 162-168
- [77] RoadRunner supercomputer website
http://en.wikipedia.org/wiki/IBM_Roadrunner

- [78] W. Schneider, M. Cole, S. Pathak, Reverse Engineering the Brain with a Circuit Diagram Based on a Segmented Connectome and System Dynamics, Proceedings of AAAI Fall Symposium on Biologically Inspired Cognitive Architectures, Washington, D.C. (2008)
- [79] J. Simpson, R. Hocken, J. Albus, The Automated Manufacturing Research Facility of the National Bureau of Standards, Journal of Manufacturing Systems, 1 (1982) 17-32
- [80] S. M. Sherman and R. W. Guillery, Exploring the Thalamus and It's Role in Cortical Function, MIT Press, Cambridge, MA (2001)
- [81] M. Shneier, T. Chang, T. Hong, W. Shackleford, R. Bostelman, J. Albus, Learning traversability models for autonomous mobile vehicles, Autonomous Robots 24 (2008) 69-86
- [82] C. Shoemaker, Development of Autonomous Robotic Ground Vehicles: DoD's Ground Robotics Research Programs: Demo I through Demo III", In R. Madhavan, E. Messina, J. Albus, (eds.) Intelligent Vehicle Systems: A 4D/RCS Approach, Nova, NY (2007)
- [83] M. Sur and C. A. Leamey, "Development and plasticity of cortical areas and networks," Nature Reviews Neuroscience, 2 (2001) 251-262,
- [84] SyNAPSE website,
http://www.spectrum.ieee.org/blog/semiconductors/devices/techtalk/darpace_seat_of_your_pa
- [85] G. Tononi, An Information Integration Theory of Consciousness. BMC Neurosci. 5: 42 (2004)
- [86] R. C. Truex and M. B. Carpenter, Human Neuroanatomy. Williams and Wilkins Co., Baltimore, MD (1969)
- [87] A. Turing, Computing machinery and intelligence, Mind 50 (1950) 433-460
- [88] J. Weng` On Developmental Mental Architectures," Neurocomputing, vol. 70, no. 13-15, (2007) 2303-2323
- [89] M. Wiegell, T. Reese, D. Tuch, G. Sorensen, V. Wedeen, Diffusion Spectrum Imaging of Fiber White Matter Degeneration, Proc. Intl. Soc. Mag. Reson. Med 9 (2001) 504

Biographical Sketch

Dr. James S. Albus founded and led the Intelligent Systems Division at the National Institute of Standards and Technology for 25 years. He is currently a Senior Fellow at the Krasnow Institute for Advanced Studies at the George Mason University. He is the author of more than 200 scientific papers and six books on robotics, intelligent systems, and brain modeling. He received a B.S. in Physics from Wheaton College (Illinois), a M.S. in Electrical Engineering from Ohio State University, and a Ph.D. in Electrical Engineering from University of Maryland.