Structural Representation as Complexity Management

Manolo Martínez

Cognition can often be modeled as the transformation of a set of variables into another. At least two kinds of entities are needed in this process: signals and coders. Representations are usually taken to be signals, but sometimes they are the coders: sometimes the computational complexity of variable transformations can be strikingly reduced by relying on a structure that mirrors that of some task-relevant entity. These kinds of coders are what philosophers call structural representations.

1 Introduction

Representations are entities that *stand for*, or *are about* other entities. They are a central theoretical posit in cognitive science, and the main ingredient in one of its foundational hypotheses: the idea that cognition can often be seen as consisting of sequences of computations over representations—what is sometimes called the *computational theory of mind*.

While original formulations of the computational theory of mind (e.g., Fodor 1980) conjured a mental model in which quasi-linguistic entities (the representations) are subjected to formal manipulation (the computations), contemporary versions take more seriously the fact that cognitive systems (e.g., brains, or brain plus bodies plus environment, depending on who you ask) are complex systems, where clean, symbol-based processes are, to put it mildly, hard to identify. One popular reaction to this fact consists in seeking analyses of the notion of representation that may be applied to complex systems, with the hope that they will guide us in the discovery of the computational-representational underpinnings of cognition (Millikan 1984; Papineau 1987; Neander 2017; Shea 2018).

Many of these analyses rely, implicitly or explicitly, on information as it is understood in the Shannonian tradition (Claude E. Shannon and Weaver 1998–1949; Cover and Thomas 2006). In this approach to information, we first consider a target system, which can be in different states at different moments in time. Having information about that system consists in being in a position to make better-than-chance predictions about the state it is in. Representations are often taken to be, among other things, entities that carry information, in this sense, about whatever it is that they represent.[1]. In a popular elaboration of this idea, representations are *trackers* (Bourget and Mendelovici 2014) or *detectors* (Gładziejewski and Miłkowski 2017; Facchin 2021): entities that are in the business of tracking (detecting) the states of a target system.

One shortcoming of this conception is that many, perhaps most, representations postulated in cognitive science and neuroscience are not of this sort: cognitive maps (O'Keefe and Nadel 1978), theories (Rehder 1999), and in general entities whose structure resemble that of some target system fit awkwardly, if at all, in the detector mold. Recently, philosophers of cognitive science have been paying a fair amount of attention to these other, so-called *structural representations* (also *SRs* henceforth: Piccinini 2020, sec. 12.2; Shea 2024, chap. 2; Artiga 2023; Gładziejewski 2015, among many others).

It is sometimes suggested that informational connections and structural resemblance are just two different ways in which representations can make themselves useful;[2] and therefore, that one should be a pluralist about the grounds of representational status. I show here that detectors and SRs are more closely integrated than that. Consideration of what information (or, as I will say, *signal*) processing involves reveals a place for SRs alongside detectors in a unified whole: the signal-processing in which much cognition consists depends both on the very *signals* that are processed and transformed (some of which, I suggest, are closely related to the trackers/detectors of the philosophical literature,) and on the devices, the *coders*, that effect those transformations—and these will be closely related to the SRs of the philosophical literature.

Another important idea in cognitive science is that many interesting cognitive phenomena and strategies are the result of resource management: it is better (and sometimes it is necessary) to implement the target transformations, or approximations thereof, in a shorter time, in less space, with processes of lower complexity. In particular, what makes SRs useful, and sometimes necessary, is the need to keep the complexity of transformations manageable. See §2 for more on signal processing as a model for cognition; and §4 for more on signals and coders. §3 introduces some key ideas in the formal study of complexity.

I spell out how complexity management gives rise to SRs in §5 and §6. These sections exploit the idea (which I first encountered in MacKay 2003) that one can reduce the complexity of some transformations by spreading them out across a graph of very simple processes that, crucially, is structurally similar to some extra-cognitive domain to which the transformation applies. I develop this idea for two kinds of processes often encountered in cognitive neuroscience. First, cognitive maps (§5): all efficient path-finding algorithms that we know of rely on SRs. I then do it again for hierarchical processing (§6): the efficient algorithms for object recognition that we know of (that is to say, deep neural

[1]Some accounts make information an essential ingredient in the analysis of representation (Dretske 1986; Isaac 2019; Martínez 2019; see also Mann 2023). Others, most prominently Millikan's biosemantics (1984, 2004) take informational links to be, not representation-constituting relations themselves, but typical consequences of representation-constituting relations.

[2]This seems to be Shea's (2018) position.

networks and their ilk) also work by relying on a data structure that mirrors how visual objects are constructed.[3]

§7 builds on the previous discussions to intervene in two discussions in the philosophical study of representations. First, the argument, popular in 4E cognitive science, that representations are less useful than usually supposed because they can be substituted by the world itself (as "its own model", as the phrase goes). Second, the discussion surrounding the perceived liberality of some accounts of SR. §8 offers some concluding remarks.

2 The Signal-Processing Aim of Cognition

Two ideas on which there is broad consensus in cognitive science and philosophy are, first, that cognition is somehow related to the generation of adaptive behavior; and, second, that what counts as adaptive depends on the circumstances of the agent, information about which is provided by sensory input, broadly conceived. As Valerie Hardcastle puts it: "[t]he very organization of the brain is more intelligible if we take the perspective that the nervous system exists fundamentally for creating motor output in response to sensory input." (1995, 102f).

We can give this idea of "motor output in response to sensory input" a straightforward signal-processing gloss. We model sensory input as a time series, S, and motor output as another one, M. Cognition then becomes the process of transforming S into M—see Fig. 1. In what follows I will call Hardcastle's transformation of sensory input into motor output the *signal processing aim of cognition*. I will assume, without much argument, that the signal processing aim is indeed an important (but not necessarily *the*) aim of cognition.

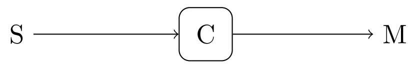

Figure 1: Cognition as signal processing—the embarrasingly simplistic picture: a cognitive system C transforms input, S, into output, M.

Let me reassure readers about a couple of possible worries. First, one does not need to believe that the brain is a computer in order to accept that cognition aims at signal-processing in this sense. Many theorists in the anti-representationalist, anti-computationalist tradition accept it too: for a couple of examples, Rodney Brooks claims that agents "must cope appropriately and in a timely fashion with changes in its dynamic environment" (1991, 145); the very idea of *enaction*, as presented by Varela and colleagues,

[3]These sections can be seen as developing the idea the representational status and computational status are, at least sometimes, tightly intertwined. See, e.g., Lee (2019); Piccinini (2022); William Max Ramsey (2023).

is that perception amounts to "perceptually guided action" (Varela, Thompson, and Rosch 2016, 173). One may take exception with describing perceptually guided action as a transformation from one variable to another, but that would be confusing the math and the territory (Andrews 2021): transformations between variables are just models of the processes that the enactivist and the embodied-cognitive scientist already accept. In particular, this transformation is carried out by a *cognitive system*, C, which is just that: whatever it is that performs the transformation between S and M. We need not assume that C is a brain (as opposed to a brain-in-a-body, or some even more encompassing system) and we need not assume that C doesn't change across time, as a result of its exposition to S. It is also perfectly possible for the value of S at some time to depend on the value of M at earlier times. While Fig. 1 presents the model in its most general form (and its least useful) I will be more interested in particular instances of variable transformation: a signal gets in and another gets out, as part of the performance of some specific cognitive task.

Second, accepting the signal-processing aim does not involve a commitment to a now implausible feedforward-only picture of the brain (*cf.* Catani and Ffytche 2005, 2232, but see below on object recognition): C might involve as many feedback loops as necessary, and as I said above, there might and often will be feedback loops between S and M themselves. The main commitment one takes when accepting the signal-processing aim is that there is an asymmetry between S and M (again, recognized by most): we take one aim of cognition to be to generate behavior that is adapted to the agent's circumstances, and not to generate circumstances that are adapted to the agent's behavior. This is compatible with niche construction, scaffolding, and other processes in which behavior feeds back into the agent's circumstances (Stotz 2010).

Apart from these two time series: S, which encodes sensory input, M, which encodes motor output; and C, which we think as a possibly evolving, possibly stateful, device that transforms the former into the latter, the signal-processing perspective also requires some way of cashing out the "adaptive" in "adaptive behavior", or the "appropriately" in "cope appropriately". I'll assume that these vague claims of appropriateness can be given (equally vague, if necessary) numerical expression: given a certain run of the S and M series up to time t, some motor responses at t are better than others. More concretely, I'll assume a *loss function* that takes runs of the S and M series up to t as inputs, and a score from 0 to 1 as output, where 0 is maximally appropriate (adaptive, or apt,) and 1 minimally so. Typically our models only consider a few turns into the past, and not the whole history of the time series. Sometimes these series are continuous, and there will be no turns, strictly speaking.

Optimally meeting the signal-processing aim of cognition amounts to always producing the best or one of the best possible motor outputs given some history of circumstances and behaviors. It's unlikely that there is a nomologically possible C able to implement this optimal transformation in a reasonable time; in any event, naturally occurring Cs will typically fall significantly short of that. Much computational cognitive science aims at reconstructing interesting aspects of cognitive function as strategies to adequately meeting the signal-processing aim in the face of, e.g., computational or information-processing shortcomings and constraints (Kirby et al. 2015; Sims 2016; Rosch 1999, among many others). In particular, in this chapter I will be interested in *computational* shortcomings and how they give sometimes rise to structural representations.

One final word for the brain-is-not-a-computer camp, so that they can keep reading with a clear conscience: exploring the computational shortcomings of a certain device, and the strategies by which these shortcomings are negotiated, does not imply that this device is a computer. Most results in complexity theory and information theory are mathematical in nature, and are true of every device, independently of whether we regard them as computers, information processors of any other sort, or none of the above. So, for example, NP-complete problems are (in all likelihood) not solvable in polynomial time. That is to say, not solvable by computers, *or any other means*, in polynomial time (Aaronson 2005). Similarly, the fact that something is not a computer or an information-processor of some sort does not somehow allow it to losslessly compress some source to a size smaller than its entropy. That this cannot be done is a mathematical theorem (Claude E. Shannon and Weaver 1998–1949). These are well-known facts about computability and information processing, on which I expand in §3. My current point is simply that the bar for some phenomenon to fall under their purview is very low—it just needs to be modelable as a transformation of some variables into other variables. Cognition, as conceived by classical-representationalist and 4E cognitive science alike, qualifies.

3 The Complexity of Transformations

Despite occasional proclamations to the contrary (e.g., Floridi 2013), there is basically one, relatively well understood theory of information transmission, for which Claude E. Shannon (1948) laid the foundations (Cover and Thomas 2006; and MacKay 2003 are two excellent textbook expositions). In this body of work, information (or entropy) is uncertainty about the value taken by a *source*: a random variable, V, or a time series, that can take different values with different probabilities, at different times.[4] The entropy of V is notated $H(V)$. Another variable, $\hat{V}$, "carries information" about the source (in information-theoretic parlance, the mutual information between these two variables is nonzero) to the extent that the probability distribution of the source changes when we conditionalize on values of the second variable. Mutual information is notated $I(V, \hat{V})$. This makes entropy and mutual information useful quantities in cognitive science, whenever we are studying a changing system (a source) that a certain cognitive agent is interacting with and keeping track of, via signals that carry some information about that source (Rieke et al. 1999; Dimitrov, Lazar, and Victor 2011).

Agents don't keep track of changing systems just because. The point is to act back on them. Above I suggested that we can think of this cycle of world change and behavioral

[4]Throughout the chapter I use "variable" for both random variables and time series (sequences of random variables.)

response, in full generality, as signal processing: turning variables informative about the world into variables that encode behavioral plans. One important shortcoming of Shannon's approach to information is that it is blind to crucial aspects of this sort of transformation. I will first present the intuitive idea, and then discuss ways to formalize it.

Suppose that we have a random variable, V, whose values are lists of 5 random numbers from 1 to 100. For example: [54, 60, 28, 24, 13]. We are tasked with finding out, for values of V, if a certain number, say, 42, is higher than any number in the list. According to the signal-processing gloss, we are transforming lists of five numbers into a single binary digit: 1 if 42 is higher than all five numbers, 0 otherwise. Doing this implies computing five comparisons.

Suppose now that another random variable, $\hat{V}$, spits a tuple: its first member is a list of the same numbers in the same order but additionally, helpfully, its second member is their maximum. In the example above, one of the values of $\hat{V}$ would be ([54, 60, 28, 24, 13], 60). Now our task is five times simpler: we only need to compare 42 with the second member in the tuple. $\hat{V}$ makes our life easier in a clear, tangible sense; but one that cannot be captured by information theory. Both sources are *informationally equivalent* in the sense that one can convert losslessly from any one of them into the other: they have the same entropy, and the mutual information between them is equal to that entropy (cf. Simon 1978; Larkin and Simon 1987): random variables V and $\hat{V}$ are informationally equivalent iff $H(V) = H(\hat{V}) = I(V; \hat{V})$.

V and $\hat{V}$ are informationally equivalent, but not *computationally equivalent*: some computations are more easily done on one than on the other. In particular, in this case, if each list of numbers in V and $\hat{V}$ has n numbers, computing if 42 is larger than the largest in the list requires n comparisons if we are dealing with V and only 1 if we are dealing with $\hat{V}$.

Unfortunately, the crucial notion in this rough characterization of computational equivalence, "easily done", does not lend itself to as clean a formalization as the one afforded by information theory for informational equivalence. One of the main ideas here[5] is to find a function of some key parameter of the problem (which we take to be a yes/no decision, such as "is 42 bigger than all numbers in the list?") whose value grows as slowly as possible, yet is always bigger than the number of steps it takes to reach a correct decision. In the example above those functions are n and 1 respectively: the number of steps needed (here, comparisons to be made) grows with the size of the list for V, and doesn't grow at all for $\hat{V}$. We say that the *time complexity* of the respective decision problems are $\mathcal{O}(n)$ (pronounced "big oh of n") and $\mathcal{O}(1)$ ("big oh of 1").[6] When the function bounding the number of steps has the form n^k for $k > 0$, we say that the problem in

[5]What follows is a very crude sketch of some ideas in classical complexity theory. An excellent introduction to these ideas for cognitive scientists and philosophers is Rooij et al. (2019). One standard textbook exposition is Papadimitriou (1993).

[6]Slightly more precisely, $f(x)$ is $\mathcal{O}(g(x))$ if $f(x) \leq a \cdot g(x)$ for some positive constant a, for big enough values of x.

question is in complexity class $\mathsf{TIME}(n^k)$, better known as P—these are the tractable problems. If *verifying* a given proof of the answer to a decision problem can be done in $\mathcal{O}(n^k)$ we say that the problem is in $\mathsf{NTIME}(n^k)$, better known as NP (Papadimitriou 1993)—these can be intractable. Analogous *space complexity classes* can be defined when what we bound is the space, or memory, necessary to reach a solution to the decision problem. There are many, *many* complexity classes—the *Complexity Zoo* (Aaronson, Kuperberg, and Granade 2005) catalogues many of them.

This approach to complexity is not without problems.[7] For one, complexity classes are very coarse-grained. For example, complexity class P doesn't care if the problem can be solved in n or n^{500} steps. They also only make sense for decision problems—those with binary, yes/no answers. This is the price one pays for being able to prove nontrivial theorems about the complexity of procedures. One consequence of this situation is that when neuroscientists and cognitive scientists study complexity they often rely on computationally feasible measures and heuristics. For example, transformations are taken to be easy if linear, and hard if nonlinear (DiCarlo, Zoccolan, and Rust 2012); easy if low-dimensional and hard otherwise (Pang, Lansdell, and Fairhall 2016); or complexity is somehow identified with patterns of connectivity (Tononi, Sporns, and Edelman 1994; Reijneveld et al. 2007).

As I said above, constrained optimization, and trade-offs of various kinds are central to computational cognitive science. It's unfortunate that computational complexity remains imperfectly integrated with other formal tools. Still, the approach summarized in this section will allow me to formulate the main idea in this chapter: at least sometimes, it is illuminating to see structural representations as the result of computational-complexity management.

4 Signals and Coders

There is at least one natural role for representations under the signal-processing view of cognition: they are the signals. Information needs to flow downstream, and something needs to take it there. Fig. 2 shows a bare bones version of how this goes: in the process of transforming some upstream variable, U, into a downstream variable, D.[8] at least two kinds of entities are involved: *signals* (the representation node) and *coders* (the sender

[7] Another important approach to complexity is so-called *algorithmic information theory.* Shannon's information theory only makes sense in the context of probabilistic sources, but we also want to make sense of the idea of the informational content of a static object, such as a book, or an algorithm that transforms one variable into another. A suggestion made independently by Kolmogorov (1965), Chaitin (1966), and Solomonoff (1964) equates this informational content with the length of the shortest program (of a designated universal Turing machine, perhaps, Li and Vitányi 2008) that reproduces the object in question. This is called the *algorithmic information content*, or the *Kolmogorov complexity* of the object. I won't be relying on algorithmic information theory here.

[8] More complicated models, e.g., with two or more senders or receivers, or relays between them, are studied in *network information theory* (El Gamal and Kim 2011).

and receiver nodes). The representation node is, like U and D, a time series: it has values that change across time, as a response to changes in U—and, perhaps, indirectly, to changes in D. These representations have the same approximate temporal profile as the changes in U that they are tracking, and are therefore comparatively short-lived. The encoder and decoder nodes, on the other hand, are longer-lived entities (although they may be stateful, and may change), tasked with the transformation of U into D, with the help of intervening signals such as *Representation*.

Figure 2: The main sender-receiver model of representation

There is another, perhaps less obvious role for representations under the signal-processing view: they are the coders. Indeed, one of the main messages of this chapter is that many of the representations postulated in cognitive science that are thought of as signals should be thought of as coders. This mischaracterization has caused some confusion.

Many examples in discussions of the nature of representations by cognitive scientists and neuroscientists (perhaps not so much by philosophers of cognitive science) present representations as what we could call *active maps*: “systematically organized patterns of activity that are isomorphic with the external world” (Poldrack 2021, 1308). Hippocampal place cells (O’Keefe and Nadel 1978) are a very common example of this kind of representation: as usually described, these are cells such that i) their activity correlates with the place occupied by the agent, and ii) cells corresponding to contiguous places are preferentially interconnected. It is often claimed that this kind of arrangement helps agents navigate their immediate environment. Peter Godfrey-Smith, for example, discusses a model of hippocampal maps by Allison Reid and John Staddon (1997, 1998) and argues, correctly in my view, that a Reid-Staddon map is not read by anything, but is rather “a self-reading map, or a map that works without being read at all” (Godfrey-Smith 2013, 57). In the following section I show how thinking of Reid-Staddon and other active maps as coders and not signals, and in particular coders aiming at reducing the computational complexity of minimal-path calculations, removes much of the puzzlement surrounding maps without reader. After that I discuss hierarchical representations, under the same light.

5 Cognitive Maps

I have argued that one important constraint on variable transformations is their complexity: for some variable S, the best possible transformation (according to whatever loss function is currently relevant) might be M, but still, possible coders C that transform S into M may be too complex for this to be a feasible solution to the problem at hand.

Other possible outcome variables M', not optimal but still satisficing, and which are more easily decoded from S, may be preferable.

So, what makes for simpler transformations? An important insight here is that one can do "complicated calculations using simple distributed hardware" (MacKay 2003, chap. 16). The "distributed hardware" in question is composed of simple nodes (capable of e.g., adding, averaging or calculating the maximum or minimum of its neighbors, but little more) connected in a way such that their composition results in the target calculation. Crucially, this often requires connecting them in a way that mirrors some relevant structure. Hence SRs.

Suppose that some reward (food, perhaps) is sitting somewhere on a certain region of space. It will sometimes be useful to learn how to reach this reward as quickly as possible from elsewhere in that space. One can naturally express this task in terms of a coder C: its input is a signal S that says "We are <here>", and it is to output another signal M that says "Go <there>", where "there" is a small step in a direction that takes us closer, ideally closest, to the goal. More precisely, C is governed by a loss function that takes pairs of values of S and M to real numbers, such that the smaller the number, the better the value of M is as a next step to the value of S, in the process of reaching the reward.

One, absolutely awful, way of going about this task is to walk that space at random from some starting point until we reach the reward, keeping track of how much distance we have walked, then starting again and again until we are satisfied with the minimum distance in all of those trials. Another, better but still pretty awful, way is to do this systematically: we start from some location and go right, ahead, left, back, etc., in an ordered sequence, keeping track of which combination of moves takes the least time to reach the reward. Awful as they are, these two methods have the advantage of not needing any internal map. The coder C just needs to keep track of sequences of moves—a very, very big number of sequences of moves indeed (see §7.2,) but no map.

It turns out that this task can be solved much more efficiently if we equip the coder with a map-like data structure, such as a *graph*: a collection of *nodes* (think of them as standing for locations) plus a collection of *edges*, each connecting two nodes (think of edges as saying that two locations are neighbors.) These edges, furthermore, are *weighted*, or associated with a real number, which we can think of as the distance between the two nodes the edge connects. Dijkstra (1959–2022) describes an algorithm that solves the above shortest-path task:[9] we start from the node that holds the reward, and move outwards. In each step, we visit the node that is closest to the current node, and update the distances from all its neighboring nodes to the reward, storing them in the nodes themselves (this is important—see §7.1). Once there are no unvisited nodes, or no unvisited node is a neighbor of the current one, we are done. As a result we have, at

[9] If all the edges weigh the same then an even simpler *breadth-first search* can find shortest paths. Dijkstra's algorithm is the way to solve it for graphs with arbitrary (nonnegative) weights (Skiena 2020, sec. 8.3).

each node, the shortest distance to the reward, and now a shortest path can be found from any node simply by always moving to the neighbor with the shortest associated distance to the reward. See Fig. 3.

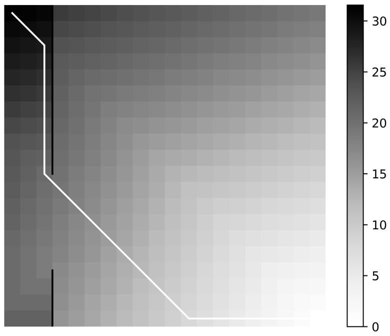

Figure 3: A "maze" with one single wall (the black lines). The color at each node corresponds to its distance to the reward node (at the lower right corner). The white line shows one optimal path from the upper left corner to the reward node.

In a series of papers in the 90s, Reid and Staddon (1997, 1998) developed a model of how cognitive maps are exploited that can be seen as a variation on Dijkstra's algorithm:[10] instead of a systematic process of visiting nodes, they allow an "activation" (a value that is initially 1 at the reward node and 0 elsewhere) to diffuse through the graph following a simple averaging process. If the space is not too complicated (not too many barriers), and diffusion is allowed to happen for long enough, *and* the agent is able to detect small activation gradients (as they diminish exponentially as one gets away from the reward node), then spaces can be navigated efficiently by allowing a Reid-Staddon map to calculate next node from previous node, until the reward node is reached. One can see this as a sort of "neuromorphic Dijkstra": diffusion is, presumably, easier to implement by neural populations than Dijkstra updating. While there is no algorithmic guarantee that Reid-Staddon diffusion will result in optimal paths, it will often result in good enough (satisficing) ones at somewhat lower complexity: a loss-complexity trade-off.

In both the Dijkstra and the Reid-Staddon algorithms, efficient calculation of optimal, or satisficing, paths to a reward is achieved by *message-passing* between nodes in a graph that is structurally similar to the region of space where the reward is. Each node does precious little: they just compare distances to the reward among their immediate neighbors and add their own distance to their minimum (for Dijkstra) or average activations among immediate neighbors (for Reid-Staddon). Maps, or at any rate these maps, are useful

[10]While Reid and Staddon do not seem to be aware of Dijkstra's algorithm, or at least do not make reference to it in their papers, other theorists of hippocampal maps, such as Muller, Stead, and Pach (1996) or Khajeh-Alijani, Urbanczik, and Senn (2015), are, and do.

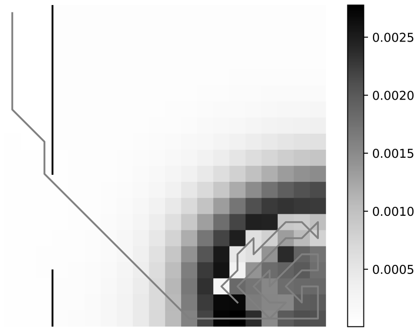

Figure 4: The same maze as in figure 3. The erratic behavior of the agent once it reaches the vicinity of the reward node is by design: the Reid-Staddon map-constructing algorithm has quirks that mimic area-restricted search (Reid and Staddon 1998, 590). All the code necessary to run these simulations and create these figures can be downloaded from [redacted for review].

because almost all of the computational heavy-lifting is done by the graph being wired up the way it is. The question of map reading does not even arise. These maps are just data structures that afford easy calculation of some target quantities. Nothing is supposed to read them; they are just ways of computing go-there from you-are-here.

6 Hierarchical Processing

It is perhaps not terribly surprising that maps are useful in solving spatial tasks—although how precisely they are useful is interesting in its own right. In any event, as it happens, the very same kind of ideas (about passing messages encoding simple computations across distributed processing that is laid out according to a structurally significant graph) are at play in many other, non-spatial tasks. Visual object recognition (the ability to correctly label an object presented visually, DiCarlo, Zoccolan, and Rust 2012) is a case in point. DiCarlo and Cox (2007, 337) propose that we think of the processing that underlies this ability as "untangling object manifolds". That is to say, the information necessary to identify an object is (by hypothesis) in the visual stimulus; it "just" needs to be repackaged ("reformatted," DiCarlo, Zoccolan, and Rust 2012, 417) in a way that can be easily turned into a label-assigning decision. This is a clear example of signal processing, the way I described it in §2.

It is sometimes claimed that invariance is "*the* computational crux of recognition" (DiCarlo, Zoccolan, and Rust 2012, 417, original emphasis): object recognition must somehow be specific enough to, e.g., tell different faces from one another; yet invariant to irrelevant transformations, such as pose, rotation, apparent size (e.g, objects depicted closely or further away) or illuminant. It is not entirely clear how cognitive systems solve this problem, but there is broad consensus that, in mammals, this happens in the visual ventral stream through a hierarchical (Güçlü and Gerven 2015; Poggio and Ullman 2013; Cao and Yamins 2024), mostly feedforward process sandwiched between V1, that detects spatial frequencies and orientations (e.g., Hubel and Wiesel 1968); and the inferotemporal cortex, that seems to support "core object recognition" (DiCarlo, Zoccolan, and Rust 2012), with representations of increasingly complex features in between.

The main algorithmic idea here (dating at least from Hubel and Wiesel 1962; see DiCarlo, Zoccolan, and Rust 2012, 425) is that the ventral hierarchy alternates between linear, AND-like stages (such as the summation of different inputs); with nonlinear, OR-like stages (such as taking the maximum of those inputs). The intuitive idea is that ANDs help with specificity (only accept entities that have this feature *and* that, *and* that other one. . .), and ORs help with invariance (accept entities that have this feature *or* that one, *or*. . .). This intuition aside, a lasagna of alternating linear and nonlinear operations is, in essence, a neural network (Riesenhuber and Poggio 1999); and it is well-known that neural networks are universal function approximators (e.g., Mhaskar 1996). In object recognition, the function they'd be approximating is one from pixels (or some other way of capturing retinal input) to labels ("car", "chair", "face").

The connection with active maps comes at this point. Poggio, Banburski, and Liao (2020) show that, while shallow neural networks (ones with just one, arbitrarily wide, hidden layer) are indeed universal function approximators, one can approximate functions with the same precision with exponentially fewer neurons, *if the neural-network graph is the same as the graph that generates the target function.* The main idea in this result is to think of the function to be approximated as a composition of simpler functions, along a binary tree (see Fig. 5B). If the neural network that is to approximate it has the same width and depth then it can be much smaller than if it simply crams all of its computational complexity into a single hidden layer. Again here, we are trading computational complexity for distributed processing; and, again, this works because we allow our neural network to pass messages in a way that emulates how the actual worldly feature is constructed.[11]

[11] One thing to bear in mind here, though, is that this allusion to the construction of worldly features is handwavy, and potentially misleading. A neural network that classifies photographs into cars and not cars obviously does not recapitulate the construction of actual cars, assembly-line style. It recapitulates the construction of something like *cars as perceived*: from textures and edges, to simple shapes, to more complicated ones (see, e.g, Zeiler and Fergus 2014, fig. 2). Chollet (2021, 205, my emphasis) remarks that it is "the *visual world* [that] is fundamentally spatially hierarchical". Vision scientists tend to find talk of, e.g, edges detection unproblematic, but there is some akwardness to claiming that the visual world *really* is constructed like this—in some flat-footed sense, it really is not.

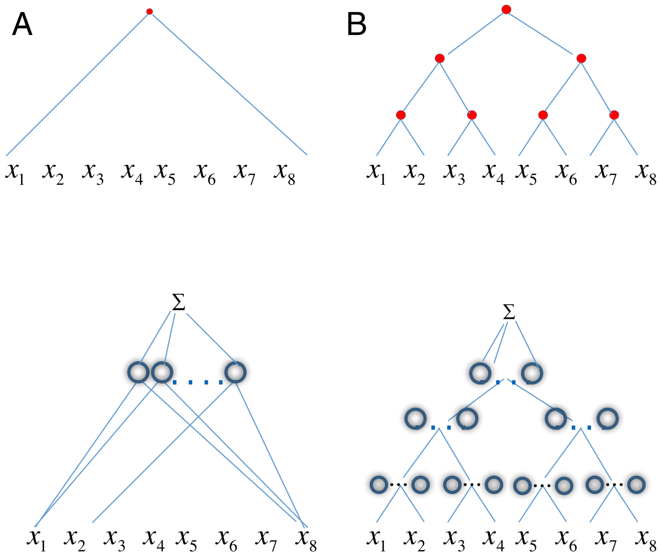

Figure 5: Reproduced from Poggio, Banburski, and Liao (2020), p. 2. Above we have two ways in which a certain collection of worldly variables, $x_1, \ldots, x_8$, could be constructed. On the left the general case (they are generated somehow or other); and on the right the computationally tractable case (they are generated via a hierarchical process, in this case a binary graph of simpler transformations.) Below we have a shallow neural network (left) and a deep one (right). The shallow network needs to have exponentially more nodes than the deep one to match the latter's accuracy of reconstruction, if the deep one mirrors the structure of the variable-constructing graph, above.

7 Some Lessons About Structural Representation

I have offered two examples (cognitive maps and hierarchical processing) where structural representations help reduce the complexity of computations associated with a certain task (finding shortest paths, and object recognition, respectively.) The mechanism, broadly conceived, is the same for both: i) nodes in a graph that bears task-relevant similarity to some worldly structure; ii) very simple computations carried out by each of these nodes; iii) such that their results are passed along and aggregated across the graph.

Beyond message-passing, there are surely other ways in which structural representations can make themselves useful to a cognitive system (see, e.g., §8); but this is a particularly clear, central way. It is also a useful model to have in mind when evaluating claims about SRs. In what follows I draw some lessons about structural representations that rely on this mental model.

7.1 Usefulness

One important source of 4E-style skepticism about the need for structural (or any other) representation in cognition is captured by Rodney Brooks in a characteristically eloquent way:

> [The physical grounding hypothesis] states that to build a system that is intelligent it is necessary to have its representations grounded in the physical world. Our experience with this approach is that once this commitment is made, the need for traditional symbolic representations soon fades entirely. The key observation is that *the world is its own best model.* It is always exactly up to date. It always contains every detail there is to be known. The trick is to sense it appropriately and often enough. (Rodney A. Brooks 1990, 5, emphasis mine)

I find the idea that the world is its own best model appealing and useful. Why, indeed, would you bother constructing a model of a place, say, and rely on *that*, if you can just get your informational input, all the informational input, from the place itself? Philosophers often talk of decouplability in this context: Gładziejewski, for example, claims that "we are not dealing with a truly representational system or mechanism unless the structures we want to treat as representations can be used off-line" (Gładziejewski 2015, 76). But, as Brooks suggests, it might be that the trick to cognition lies precisely on insisting in sensing the world appropriately and often enough, rather than on retreating from it and working with a surrogate.

An alternative reply to Brooks is that *the world does not always support message-passing.* The shortest-path algorithms I reviewed above rely on being able to store a "message" (an accumulated distance or, more or less interchangeably, a measure of diffused activation) at each node that can be then passed onto neighboring nodes. The world sometimes affords this: some rewards give out, e.g., semiochemicals that diffuse more or less according to the Reid-Staddon model—at least so far as the diffusion doesn't have to go around corners. But very often it does not: the reward can be buried; or not be a reward at all, just an important location. In those cases calculating shortest paths in an efficient manner, at least insofar as we know how to do it, requires having a slot per location in which an intermediate result (provisional minimal distances to the reward or activation coefficients, depending on the model) can be inscribed. Again, perhaps sometimes the world itself can play this role: we can bath it in semiochemicals, or cover it in PostIt notes. Often, though, this too will be impossible or inconvenient. Hence the need for a map. The case of hierarchical representations is even worse for the world-as-its-own-model strategy: in object recognition nodes do not represents places, but something like stages in the generation of whole visual-objects from features such as edges or blobs, through increasingly complex features. The notion that we may store messages on edges or on Gabor patterns is close to nonsensical.

There may be other algorithms that solve the above tasks without relying on maps or hierarchies—ones we haven't come up with yet. It is also possible that natural cognitive systems "satisfice" by relying on worse (higher complexity) algorithms to solve them. Finally, it is possible that the world can afford extramental message passing in many more situations than I have considered. All of these possibilities are worth considering; and in particular 4E cognitive scientists have made great strides in showing everyone else that the world is more accommodating to various extended forms of cognition than traditionally supposed. Still, the above considerations provide substantial support to the claim that some SRs are useful, and how.

7.2 The Liberality Challenge

Another important worry about structural representations concerns whether there is a principled way to tell useful from useless structure: it is easy to make any extramental system and any cognitive system "structurally correspond" to each other, by taking one graph, and then labeling some of the entities and relations of each system (there will be many candidates) as nodes and edges of that one graph: now both systems "instantiate the graph." Shea's (2018, chap. 5) response is, as far as it goes, perfectly sensible: not all structural correspondences count, but only those that are *exploitable* by the system.

Shea unpacks exploitability in terms of the claims that the extramental structure is "of significance" to the system, and the structurally corresponding cognitive structure is one that the system is "systematically sensitive" to (Shea 2018, 120). I don't think that much more can be said about the exploitability of structure that provides fully general demarcation criteria between useful and useless. On the other hand, the notions of significance and systematic sensitivity risk being too unspecific to inform substantial theorizing about SRs. For example, Facchin (2021) has argued that what he calls *receptors* (systems the states of which reliably correlate with some extramental variable: the signals of §4, or close enough) count as SRs, among other things because at least sometimes cognitive systems *exploit* structural similarity between receptors and the world (Facchin 2021, sec. 4.1). This is a problem because, Facchin claims, receptors have been independently shown to fail Ramsey's (2007) "job description challenge": we do not need to think of them as representations to understand how they contribute to the system that hosts them.

Regardless of how one feels about Facchin's argument, this line of attack is opened by tying structural-representation status to nonspecific notions of exploitability. Being more specific helps. The idea I have been developing here is that the structural homomorphisms "that count" are those that meaningfully reduce the complexity of a target transformation—they are exploitable in that precise sense, and to that precise degree. For example, Dijkstra's algorithm can be completed in a number of steps that grows as the square of the number of nodes (Skiena 2020, 260): its complexity is $\mathcal{O}(n^2)$. The complexity of a naive brute-force exhaustive search over all paths is much higher. For illustration, consider only "self-avoiding walks" (those that do not cross themselves) on a 10×10 grid, using a Manhattan neighborhood (only nodes up, down, left or right count as neighbors, not those across a diagonal). The problem of calculating the optimal path from the top left node to the bottom right node with a Dijkstra-style approach takes $10^2 = 100$ steps, but enumerating all paths between them and then finding the shortest takes involves considering 1568758030464750013214100 different paths (Bousquet-Mélou, Guttmann, and Jensen 2005, Table 2).[12] This is a very striking computational difference; one that clearly makes cognitive maps useful. While we don't know that brains implement Dijkstra-style algorithms over these maps, "it is difficult to imagine solutions of the planning problem which do not involve this core idea", (Khajeh-Alijani, Urbanczik, and Senn 2015, 2). For hierarchical representations, as we have seen, the difference in complexity is also very large: the number of nodes in a shallow net that is able to approximate a certain function F to some precision is exponentially larger than for a network that approximates it to the same precision, but has the same structure as the compositional graph of F. Here, though, it is size and not time that grows. The job description challenge is met even more explicitly: you can do object recognition with three layers, but you can do it vastly more efficiently if your processing graph mimics the generative structure of the objects you are recognizing.

I am claiming that a SR is exploitable if it reduces the complexity of a target transformation, but of course the usual provisos apply. Exploitability doesn't entail actual exploitation (Rathkopf 2017). For many problems there will be various different, similarly useful SRs, and complexity considerations will not allow us to pick one among them.

[12]Shea (2024, 36), discussing cognitive maps, sketches an optimal-path algorithm that "starts by activating a target location to be reached and runs through sequences of co-activation that trace back to the current location. It [then] picks the shortest of the simulated sequences and, switching to online mode, follows that sequence in reverse in order to reach the target." In this section of his book Shea is not thinking about complexity questions, but simply offering an intuitive algorithm over a cognitive map. It is still interesting to note that the time that this search would take, unless suitably constrained, grows super-exponentially with the number of place cells.

8 Concluding Remarks

Cognition can often be interpreted in terms of variables being transformed into other variables. These transformations are sometimes dramatically simplified by off-loading part of the requisite computations to a data structure, homomorphic in some relevant respects to the real-world phenomenon the transformations in question are related to. I have suggested that these structures are a central example of what philosophers call structural representation.

One potentially awkward aspect of seeing some “variable transformers”, aka coders, as representations is that coders are not there to be read (interpreted, consumed, etc.), which is what the mainstream sender-receiver approach claims representations are for. Coders are there to transform variables into other variables. Sometimes, as in the examples discussed in this chapter, they need to *be* structurally similar to some worldly entity in order to do this efficiently, but nothing need be there to take note of that similarity. So, this seems to result in a pluralistic metaphysics of representation: there are signal-like representations and coder-like representations, and no non-trivial genus they are both species of. Shea (2018, chap. 2) agrees, for different reasons: he sees structural correspondence and correlational information as two different kinds of exploitable relation, perhaps among others. This is surely true enough, but, as I have argued above, we should not make our theories of representation rest on notions as general as exploitability or significance. Another, and I suggest better, way of integrating signal-like and coder-like representations in a meaningful whole is, precisely, as the two main types of nodes in the sender-receiver architectural motif (Fig. 2): sometimes signals need to carry world-related information from one variable to another (this may result in a representational role), *and* in order to do this efficiently coders may need to be structurally similar to some entity relevant to the task at hand. This latter fact also commonly elicits attributions of representational role in neuroscience and cognitive science (Poldrack 2021). Our task as philosophers of cognitive science is not to correct this usage, but to understand it.

I have spent some time here spelling out how distributed processing helps explain computationally efficient variable transformation. I have focused on message passing, but surely there there are instances of complexity management that do not fit the message-passing mold, or do so imperfectly. Imperfect fit will sometimes invite thinking of the phenomenon in question as a peripheral, or borderline, case of structural representation. Some other times it will invite looking for other complexity-management strategies that still rely on implementing a world-homomorphic graph (e.g., as a way to keep the volume of axonal tracts as small as possible, Kriegeskorte and Diedrichsen 2019, sec. 3.4). On the other hand, I suspect that the usefulness of modeling SRs after public maps is sometimes overblown.

I regard the complexity-management argument for SRs as continuous with, and complementary to, other defenses of the role of representations in cognition. For example, Piccinini’s “argument from complex control”:

1. Nervous systems perform complex control functions in a computationally tractable way.
2. Performing complex control functions in a computationally tractable way requires processing structural representations.

Therefore, nervous systems process structural representations. (Piccinini 2020, 264)

Here I have provided some complexity-theoretic detail in favor of Piccinini's second premise.

Similarly, Clark and Toribio famously characterize a "representation-hungry" problem as one that "involves reasoning about absent, non-existent, or counterfactual states of affairs" or "requires the agent to be selectively sensitive to parameters whose ambient physical manifestations are complex and unruly (for example, open-endedly disjunctive)" (Clark and Toribio 1994, 419). Nothing I have said contradicts the importance of representation-hungry problems in the emergence of representations. On the other hand, at least some of the tasks that I have discussed here are not representation-hungry in this sense. A laboratory mouse trying to navigate a region of space can be fully "coupled" to its surroundings and still need representations for the simple reason that the space in question does not have slots in which to store partial distances. Nor is navigation open-endedly disjunctive, complex, or unruly. Indeed it is a perfectly tractable problem, solvable in polynomial time. Focusing on the complexity of transformation (when this analysis is at all feasible) sometimes provides a more explicit spelling-out of the need for reliance on representations than appeals to representation-hungriness.

Finally, nothing of what I have said should be taken to imply that SRs are everywhere useful. The "distributed hardware" trick may work when the target variable transformation i) can be approximated by many sufficiently simpler transformations, that are ii) hooked up in a world-mirroring way. Even if the existence of universal function approximators perhaps means that the first condition is universally met, there's no reason to think that every cognitively relevant transformation also meets the second condition.

References

Aaronson, Scott. 2005. "NP-complete Problems and Physical Reality." *SIGACT News* 36 (1): 30–52. https://doi.org/10.1145/1052796.1052804.

Aaronson, Scott, Greg Kuperberg, and Christopher Granade. 2005. "The Complexity Zoo." https://cse.unl.edu/~cbourke/latex/ComplexityZoo.pdf.

Andrews, Mel. 2021. "The Math Is Not the Territory: Navigating the Free Energy Principle." *Biology & Philosophy* 36 (3): 30. https://doi.org/10.1007/s10539-021-09807-0.

Artiga, Marc. 2023. "Understanding Structural Representations." *The British Journal for the Philosophy of Science*, November, 728714. https://doi.org/10.1086/728714.

Bourget, David, and Angela Mendelovici. 2014. "Tracking Representationalism: William Lycan, Fred Dretske and Michael Tye." In *Philosophy of Mind: The Key Thinkers*, 209–35. London: Bloomsbury. https://books.google.com/books?hl=es&lr=&id=SFGpAgAAQBAJ&oi=fnd&pg=PT171&dq=tracking+representationalism&ots=UKVXCLqHl0&sig=bJrGjhsTM5XH6F80CRH-ndgCQDw.

Bousquet-Mélou, M., A. J. Guttmann, and I. Jensen. 2005. "Self-Avoiding Walks Crossing a Square." September 15, 2005. https://doi.org/10.48550/arXiv.cond-mat/0506341.

Brooks, Rodney A. 1990. "Elephants Don't Play Chess." *Robotics and Autonomous Systems* 6 (1–2): 3–15.

Brooks, Rodney A. 1991. "Intelligence Without Representation." *Artificial Intelligence* 47 (1–3): 139–59. https://doi.org/10.1016/0004-3702(91)90053-m.

Cao, Rosa, and Daniel Yamins. 2024. "Explanatory Models in Neuroscience, Part 1: Taking Mechanistic Abstraction Seriously." *Cognitive Systems Research* 87 (September): 101244. https://doi.org/10.1016/j.cogsys.2024.101244.

Catani, Marco, and Dominic H. Ffytche. 2005. "The Rises and Falls of Disconnection Syndromes." *Brain* 128 (10): 2224–39. https://doi.org/10.1093/brain/awh622.

Chaitin, Gregory J. 1966. "On the Length of Programs for Computing Finite Binary Sequences." *Journal of the ACM (JACM)* 13 (4): 547–69.

Chollet, François. 2021. *Deep Learning with Python.*

Clark, Andy, and Josefa Toribio. 1994. "Doing Without Representing?" *Synthese* 101 (3): 401–31. https://doi.org/10.1007/BF01063896.

Cover, T. M., and J. A. Thomas. 2006. *Elements of Information Theory.* New York: Wiley.

DiCarlo, James J., and David D. Cox. 2007. "Untangling Invariant Object Recognition." *Trends in Cognitive Sciences* 11 (8): 333–41. https://doi.org/10.1016/j.tics.2007.06.010.

DiCarlo, James J., Davide Zoccolan, and Nicole C. Rust. 2012. "How Does the Brain Solve Visual Object Recognition?" *Neuron* 73 (3): 415–34. https://doi.org/10.1016/j.neuron.2012.01.010.

Dijkstra, E. W. 1959–2022. "A Note on Two Problems in Connexion with Graphs." In *Edsger Wybe Dijkstra*, edited by Krzysztof R. Apt and Tony Hoare, 1st ed., 287–90. New York, NY, USA: ACM. https://doi.org/10.1145/3544585.3544600.

Dimitrov, Alexander G., Aurel A. Lazar, and Jonathan D. Victor. 2011. "Information Theory in Neuroscience." *Journal of Computational Neuroscience* 30 (1): 1–5. https://doi.org/10.1007/s10827-011-0314-3.

Dretske, Fred. 1986. "Misrepresentation." In *Belief: Form, Content, and Function*, edited by Radu J. Bogdan, 17–36. Oxford University Press. https://philarchive.org/rec/DREM.

El Gamal, Abbas, and Young-Han Kim. 2011. *Network Information Theory.* Cambridge university press.

Facchin, Marco. 2021. "Structural Representations Do Not Meet the Job Description Challenge." *Synthese* 199 (3–4): 5479–5508. https://doi.org/10.1007/s11229-021-

03032-8.
Floridi, Luciano. 2013. *The Philosophy of Information.* Oxford University Press. https://books.google.com?id=l8RoAgAAQBAJ.
Fodor, Jerry A. 1980. *The Language of Thought.* 1 edition. Cambridge, Mass: Harvard University Press.
Gładziejewski, Paweł. 2015. "Explaining Cognitive Phenomena with Internal Representations: A Mechanistic Perspective." *Studies in Logic, Grammar and Rhetoric* 40 (1): 63–90. https://doi.org/10.1515/slgr-2015-0004.
Gładziejewski, Paweł, and Marcin Miłkowski. 2017. "Structural Representations: Causally Relevant and Different From Detectors." *Biology and Philosophy* 32 (3): 337–55. https://doi.org/10.1007/s10539-017-9562-6.
Godfrey-Smith, Peter. 2013. "Signals, Icons, and Beliefs." In *Millikan and Her Critics*, edited by Dan Ryder, Kenneth Williford, and Justine Kingsbury, 41–62. Wiley-Blackwell Chichester. https://petergodfreysmith.com/PGS_SignalsIconsBeliefs.pdf.
Güçlü, Umut, and Marcel A. J. van Gerven. 2015. "Deep Neural Networks Reveal a Gradient in the Complexity of Neural Representations Across the Ventral Stream." *Journal of Neuroscience* 35 (27): 10005–14. https://doi.org/10.1523/JNEUROSCI.5023-14.2015.
Hardcastle, Valerie Gray. 1995. "A Critique of Information Processing Theories of Consciousness." *Minds and Machines* 5 (1): 89–107. https://doi.org/10.1007/BF00974191.
Hubel, D. H., and T. N. Wiesel. 1962. "Receptive Fields, Binocular Interaction and Functional Architecture in the Cat's Visual Cortex." *The Journal of Physiology* 160 (1): 106–54. https://doi.org/10.1113/jphysiol.1962.sp006837.
———. 1968. "Receptive Fields and Functional Architecture of Monkey Striate Cortex." *The Journal of Physiology* 195 (1): 215–43. https://doi.org/10.1113/jphysiol.1968.sp008455.
Isaac, Alistair M. C. 2019. "The Semantics Latent in Shannon Information." *The British Journal for the Philosophy of Science* 70 (1): 103–25. https://doi.org/10.1093/bjps/axx029.
Khajeh-Alijani, Azadeh, Robert Urbanczik, and Walter Senn. 2015. "Scale-Free Navigational Planning by Neuronal Traveling Waves." *PLOS ONE* 10 (7): e0127269. https://doi.org/10.1371/journal.pone.0127269.
Kirby, Simon, Monica Tamariz, Hannah Cornish, and Kenny Smith. 2015. "Compression and Communication in the Cultural Evolution of Linguistic Structure." *Cognition* 141 (August): 87–102. https://doi.org/10.1016/j.cognition.2015.03.016.
Kolmogorov, Andrei N. 1965. "Three Approaches to the Quantitative Definition of Information." *Problems of Information Transmission* 1 (1): 1–7.
Kriegeskorte, Nikolaus, and Jörn Diedrichsen. 2019. "Peeling the Onion of Brain Representations." *Annual Review of Neuroscience* 42 (July): 407–32. https://doi.org/10.1146/annurev-neuro-080317-061906.
Larkin, Jill H., and Herbert A. Simon. 1987. "Why a Diagram Is (Sometimes) Worth Ten Thousand Words." *Cognitive Science* 11 (1): 65–100. https://doi.org/10.1111/j.1551-6708.1987.tb00863.x.

Lee, Jonny. 2019. "Structural Representation and the Two Problems of Content." *Mind & Language* 34 (5): 606–26. https://doi.org/10.1111/mila.12224.

Li, Ming, and Paul Vitányi. 2008. *An Introduction to Kolmogorov Complexity and Its Applications. Texts in Computer Science*. Vol. 9. Springer, New York,.

MacKay, David JC. 2003. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press.

Mann, Stephen Francis. 2023. "The Relevance of Communication Theory for Theories of Representation." *Philosophy and the Mind Sciences* 4 (December). https://doi.org/10.33735/phimisci.2023.10992.

Martínez, Manolo. 2019. "Representations Are Rate-Distortion Sweet Spots." *Philosophy of Science* 86 (5): 1214–26. https://doi.org/10.1086/705493.

Mhaskar, H. N. 1996. "Neural Networks for Optimal Approximation of Smooth and Analytic Functions." *Neural Computation* 8 (1): 164–77. https://doi.org/10.1162/neco.1996.8.1.164.

Millikan, Ruth Garrett. 1984. *Language, Thought and Other Biological Categories*. Cambridge, MA: The MIT Press.

———. 2004. *Varieties of Meaning*. The MIT Press.

Muller, R U, M Stead, and J Pach. 1996. "The Hippocampus as a Cognitive Graph." *Journal of General Physiology* 107 (6): 663–94. https://doi.org/10.1085/jgp.107.6.663.

Neander, Karen. 2017. *A Mark of the Mental: In Defense of Informational Teleosemantics*. MIT Press.

O'Keefe, John, and Lynn Nadel. 1978. *The Hippocampus as a Cognitive Map*. Oxford : New York: Clarendon Press ; Oxford University Press.

Pang, Rich, Benjamin J. Lansdell, and Adrienne L. Fairhall. 2016. "Dimensionality Reduction in Neuroscience." *Current Biology* 26 (14): R656–60. https://doi.org/10.1016/j.cub.2016.05.029.

Papadimitriou, Christos. 1993. *Computational Complexity*. 1st edition. Reading, Mass: Pearson.

Papineau, David. 1987. *Reality and Representation*. Oxford: Basil Blackwell.

Piccinini, Gualtiero. 2020. *Neurocognitive Mechanisms: Explaining Biological Cognition*. Oxford, New York: Oxford University Press.

———. 2022. "Situated Neural Representations: Solving the Problems of Content." *Frontiers in Neurorobotics* 16 (April): 846979. https://doi.org/10.3389/fnbot.2022.846979.

Poggio, Tomaso, Andrzej Banburski, and Qianli Liao. 2020. "Theoretical Issues in Deep Networks." *Proceedings of the National Academy of Sciences* 117 (48): 30039–45. https://doi.org/10.1073/pnas.1907369117.

Poggio, Tomaso, and Shimon Ullman. 2013. "Vision: Are Models of Object Recognition Catching up with the Brain?" *Annals of the New York Academy of Sciences* 1305 (1): 72–82. https://doi.org/10.1111/nyas.12148.

Poldrack, Russell A. 2021. "The Physics of Representation." *Synthese* 199 (1–2): 1307–25. https://doi.org/10.1007/s11229-020-02793-y.

Ramsey, William M. 2007. *Representation Reconsidered*. Cambridge: Cambridge University Press. https://books.google.com?id=GW59h6Qs5xcC.

Ramsey, William Max. 2023. "The Hard Problem of Content Is Neither." *Review of Philosophy and Psychology*, December, 1–22. https://doi.org/10.1007/s13164-023-00714-9.

Rathkopf, Charles. 2017. "Neural Information and the Problem of Objectivity." *Biology & Philosophy* 32 (3): 321–36. https://doi.org/10.1007/s10539-017-9561-7.

Rehder, Bob. 1999. "A Causal Model Theory of Categorization." In *Proceedings of the 21st Annual Meeting of the Cognitive Science Society*, 595–600.

Reid, Alliston K., and J. E. R. Staddon. 1997. "A Reader for the Cognitive Map." *Information Sciences* 100 (1): 217–28. https://doi.org/10.1016/S0020-0255(97)00042-X.

———. 1998. "A Dynamic Route Finder for the Cognitive Map." *Psychological Review* 105 (3): 585–601.

Reijneveld, Jaap C., Sophie C. Ponten, Henk W. Berendse, and Cornelis J. Stam. 2007. "The Application of Graph Theoretical Analysis to Complex Networks in the Brain." *Clinical Neurophysiology* 118 (11): 2317–31. https://doi.org/10.1016/j.clinph.2007.08.010.

Rieke, Fred, David Warland, Rob de Ruyter Van Steveninck, and William S. Bialek. 1999. *Spikes: Exploring the Neural Code*. Cambridge, MA: The MIT Press.

Riesenhuber, Maximilian, and Tomaso Poggio. 1999. "Hierarchical Models of Object Recognition in Cortex." *Nature Neuroscience* 2 (11): 1019–25. https://doi.org/10.1038/14819.

Rooij, Iris van, Mark Blokpoel, Johan Kwisthout, and Todd Wareham. 2019. *Cognition and Intractability: A Guide to Classical and Parameterized Complexity Analysis*. Cambridge ; New York, NY: Cambridge University Press.

Rosch, Eleanor. 1999. "Principles of Categorization." In *Concepts: Core Readings*, edited by Eric Margolis and Stephen Laurence, 189–206. The MIT Press.

Shannon, Claude E. 1948. "A Mathematical Theory of Communication." *The Bell System Technical Journal* 27 (3): 379–423.

Shannon, Claude E, and Warren Weaver. 1998–1949. *The Mathematical Theory of Communication*. University of Illinois press.

Shea, Nicholas. 2018. *Representation in Cognitive Science*. Oxford: Oxford University Press.

———. 2024. *Concepts at the Interface*. 1st ed. Oxford University PressOxford. https://doi.org/10.1093/9780191997167.001.0001.

Simon, Herbert A. 1978. "On the Forms of Mental Representation." In *Perception and Cognition*, edited by W. Savage, 9–3. University of Minnesota Press.

Sims, Chris R. 2016. "Rate–Distortion Theory and Human Perception." *Cognition* 152: 181–98.

Skiena, Steven S. 2020. *The Algorithm Design Manual*. Texts in Computer Science. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-54256-6.

Solomonoff, Ray J. 1964. "A Formal Theory of Inductive Inference. Part I." *Information and Control* 7 (1): 1–22.

Stotz, Karola. 2010. "Human Nature and Cognitive–Developmental Niche Construction." *Phenomenology and the Cognitive Sciences* 9 (4, 4): 483–501. https://doi.org/10.100

7/s11097-010-9178-7.
Tononi, G, O Sporns, and G M Edelman. 1994. "A Measure for Brain Complexity: Relating Functional Segregation and Integration in the Nervous System." *Proceedings of the National Academy of Sciences* 91 (11): 5033–37. https://doi.org/10.1073/pnas.91.11.5033.
Varela, Francisco J., Evan Thompson, and Eleanor Rosch. 2016. *The Embodied Mind: Cognitive Science and Human Experience*. Revised edition edition. Cambridge, Massachusetts ; London England: The MIT Press.
Zeiler, Matthew D., and Rob Fergus. 2014. "Visualizing and Understanding Convolutional Networks." In *Computer Vision – ECCV 2014*, 818–33. Springer, Cham. https://doi.org/10.1007/978-3-319-10590-1_53.