The Logic of Qualia

Drew McDermott, 2014-08-14

Draft! Comments welcome! Do not quote!

Abstract: Logic is useful as a neutral formalism for expressing the contents of mental representations. It can be used to extract crisp conclusions regarding the higher-order theory of phenomenal consciousness developed in (McDermott, 2001, 2007). A key aspect of conscious perceptions is their connection to the distinction between appearance and reality. Perceptions must often be corrected. To do so requires that the logic of perception be able to represent the logical structure of judgment events, that is, to include the formulas of the logic as objects to be reasoned about. However, there is a limit to how finely humans can examine their own representations. Terms representing primary and secondary qualities seemed to be *locked,* so that the numbers (or levels of neural activation) that are their essence are not directly accessible. Humans feel a need to invoke "intrinsic," "nonrelational" properties of many secondary qualities — their *qualia* — to "explicate" how we compare and discriminate among them, although this is not actually how the comparisons are accomplished. This model of qualia explains several things: It accounts for the difference between "normal" and "introspective" access to a perceptual module in terms of quotation. It dissolves Jackson's knowledge argument by explaining what Mary learns as a fictional but undoubtable belief structure. It makes spectrum inversion logically impossible by providing a degree of freedom between the physical structure of the brain and the representations it contains that redescribes putative cases of spectrum inversion as alternative but equivalent ways of mapping physical states to representational states.

1 Introduction

The purpose of this paper is to clarify a theory of phenomenal consciousness (McDermott, 2001, 2007) by being explicit about the content it finds in metarepresentations

about perception. Briefly, the theory is that consciousness is awareness of perceptual events, awareness mediated by a belief system about how perception works. It is this system that causes us to believe in sensations as key events in any kind of perception, many of which possess intrinsic, nonrelational *qualia* that allow us to discriminate among different sorts of sensation. Many of our beliefs about perception are not literally true, but that doesn't matter. For reasons of biology and culture, we find the standard model of perception to be compelling, to the point that even a convinced skeptic uses it in everyday life (McDermott, 2013).

My theory of phenomenal consciousness is of the genus "second-order" (Rosenthal, 1986, Carruthers, 1998), but not one of the usual species. What is often called "first-order awareness" of one's environment, what babies and cats have, does not require phenomenal consciousness at all, in my opinion. (I will argue for this possibly shocking idea in section 3.1.) Phenomenal consciousness as such manifests itself as the ability to think about perception, to focus on the perception of a thing rather than the thing perceived, and to distinguish the way things seem perceptually from the way they are.

I use logic to make the content of the perceptual self-model as precise as possible. The payoff is to provide crisp solutions to classic problems in the philosophy of perception, including the "knowledge argument" (Jackson, 1986) and spectrum inversion. The use of logic is justified by the idea that the content of any mental representation can be expressed using logical notation, even if this notation isn't or couldn't be used by the brain.

Not everything a creature believes is represented at all. For example, the human visual system "assumes" that if a set of perceived surfaces can be fit together to make a connected object, then that is the object it's looking at. One of the illusions created by Adelbert Ames (Ittleson, 1952) exploited this assumption to fool the eye into perceiving a set of well-behaved objects using surfaces that were actually widely separated in space. This and many other assumptions are "wired into" the human visual system at birth [¡¿], and it would be difficult or impossible to train someone to withdraw them. But it would be absurd to think they are "written down" in the

visual system in anything like a representation system.

However, beliefs that track a changing situation, a simple one like a chess game or a complex one such as a delicate negotiation, must surely be represented. This is almost tautologous: If an agent's brain changes as a situation changes in a way that allows the agent to act appropriately in the situation, some might declare that such changes are the essence of representation.

Fodor makes a closely related observation in (Fodor, 1985). He concedes Dennett's point (Dennett, 1978b, p. 107) that a chess program might "think... it should get its queen out early" without this belief being represented anywhere. But he goes on:

> The rule "get it out early" may be emergent out of its own implementation; out of lower-level heuristics, that is, any one of which may or may not itself be explicitly represented. But the representation of the board — of actual or possible states of play — over which such heuristics are defined *must* be explicit or the representational theory of chess playing is simply false. The theory says that a train of chess thoughts is a causal sequence of tokenings of chess representations. If, therefore, there are trains of chess thoughts but no tokenings of chess representations, it follows that something is not well with the theory. (Fodor, 1985, p. 95)

Of course, not everything that a chess program, or a chess player's brain, does is "a causal sequence of tokenings of chess representations." Human grandmasters, given a chess situation, are able to recall similar situations they've seen before and what was tried then (Chase and Simon, 1973a,b), and there's no obvious reason a chess program couldn't do the same thing.[1] But to the extent that a program or brain knows the current position, remembers how it got there, and ponders alternative future moves, the use of representations is the only idea we have regarding how it does these things.

There are several possible systems of representation, including but not limited to maps (Cummins, 1989), images (Kosslyn, 1983), graphs (Baader et al., 2003),

[1] Although no one knows how to get a program to do it.

and logical formulas (Fodor, 1975). I will assume that they differ primarily in their computational properties, but that they all have content that can be expressed independently of those properties. Computing the distance between two cities might be done in different ways in maps, images, networks, and logical databases, at different speeds and with different degrees of precision and accuracy. However, my focus in this paper is on the *content* of representations, and I will assume that logical formulas can capture this content as well as any other notation, even if they're a bit clumsy in expressing content extracted from, say, visual or mental images.

No matter what sorts of representation exist in the brain, I assume they are used as inputs and outputs of *computational* processes. Some cognitive scientists believe that this is an old-fashioned point of view, superseded by connectionism on the theoretical side and neuroscience on the experimental side. But neural nets are basically a style of algorithm, produced by procedures for learning simple algorithms from data (Hastie et al., 2009, ch. 11).[2] Meanwhile, although neuroscience has had an enormous impact on psychology in recent years, much of it has been via experiments yielding functional-MRI data. Such experiments treat the brain as consisting of modules connected by channels, and look for evidence that one module rather than another is active in solving a particular task. The question *how* the modules accomplish their tasks is pushed off to the future. So far I know of no concrete, noncomputational answer to this question.[3]

The rest of the paper is organized as follows. Section 2 will lay out the logical formalism I will be using. Section 3 will use this formalism to express my theory

[2]It's odd how seldom attention is called to the fact that the output of a neural net is almost always digital, a simple categorical decision.

[3]One alternative is "dynamicism," the doctrine that the brain is best thought of in terms of differential equations involving large numbers of variables whose values are neural-activity levels (and ultimately the activity levels of sensors and effectors). Representation is held to be an unnecessary approximation that will wither away once the differential equations are properly understood. This approach's successes tend to be in the area of behavior requiring tight coupling between sensors and effectors (such as compliant-motion control). In all other cases, so far all we have are manifestoes (van Gelder, 1998), vague appeals (Shanahan, 2010), and irrelevant computational exercises (Spivey, 2007).

of perception and qualia. Section 4 will use the theory to explore puzzles about "what things are like" and spectrum inversion. Finally, section 5 will summarize my conclusions.

2 Logic As A Tool For Expressing Content

2.1 Decoding, Semantics

Representations are abstractions; they are physical patterns some of whose variations "mean something." Consider a writing system such as ancient Egyptian hieroglyphics. Inscriptions using this system were known to Renaissance Europeans, but their first attempts at reproducing them failed to capture the important aspects of the symbols. Having guessed that hieroglyphics were a pictographic system, they felt free to change the spacing and pictures in their attempts at copying them. The circulation of misleading copies made the system that much harder to decipher Pope (1999). We are at a similar stage in our understanding of neural signals. Hieroglyphics are patterns of ink on papyrus; cuneiform writing systems are patterns of indentations in clay; representations in the brain are patterns of neural connections and neural activity. In each case it is not obvious how to derive the symbolic pattern from the physical reality.

In the case of writing systems, the humans that invented and used them knew what the inscriptions meant and how to translate them into intelligible glyphs. In the case of undesigned representation systems such as those in cells and brains, there is no one we can ask for the decoding manual. Indeed, there is no guarantee that there is exactly one correct manual. For example, it is by now well understood that DNA and RNA molecules encode triples of "letters" by aligning sequences of guanine, adenine, thymine, cytosine, and uracil molecules ("nucleobases") according to various rules (Watson et al., 2007), and that the sequences are broken into triples that code for particular amino acids. Some sequences of nucleobases do not code for amino acids, but instead are used as markers indicating where the first triple in a sequence starts. [¡¿] However, in many viruses and even in eukaryotic organ-

isms (Sanna et al., 2008), some genes overlap others by staggering the triples: Gene 2 starts with the second nucleobase of gene 1, so that each of its triples overlaps two triples of gene 1. The discovery that this occurs in eukaryotic organisms is fairly recent. What other discoveries about the proper way to decode DNA remain to be made? [[[Junk DNA?]]]

The only way to react to this kind of indeterminism is to say that there is no correct decoding of a physical system, even one in the brain. Different decodings lead to different results. Of the infinite number of possible decodings a system's states, most yield uninteresting descriptions of the system's behavior. When we're lucky enough to find a decoding that allows us to say, for instance, "This brain structure works like a map," we tend to conclude that it's the "correct" decoding (McDermott, 2001).

As Chalmers has emphasized (1994), choosing a decoding allows us to describe certain physical systems as computers.[4] But a decoding must not be mistaken for the *semantic interpretation* of a representation system. A decoding maps physical structures into representations; semantics maps representations into objects and relationships. A decoding might tell us that a piece of brain tissue and its states constitute a map-like structure. The (correct) semantic interpretation of a map is a piece of space that the map corresponds to with some degree of accuracy, especially if changes in the map track changes in the positions of objects in the piece of space (McDermott, 2007).

2.2 Why Logic?

In the main body of the paper I will be assuming that the *content* of any representation can be adequately captured using a logical notation. The content of a brain state encoding a propositional attitude is what remains the same as the "attitude" part varies. One can consider this to be a proposition, or a mapping from possible worlds to truth values; or perhaps these are the same thing.

[4]Chalmers does not use the word "decoding," but that's what the function is that he calls merely f.

The only attitudes I will be concerned with will be belief and doubt. Normally an agent believes what its senses tell it, but when it comes to doubt them then it invokes the distinction between appearance and reality, which is my main concern. So it might believe it sees a horse illuminated by sun shining through a picket fence, when it's actually looking at a zebra (Stalnaker, 1998). It has misjudged the nature of some of the boundaries, believing them to be illumination boundaries instead of reflectance boundaries (Forsyth and Ponce, 2002).

Visual information seems like a good candidate to require "map-like" representations rather than propositional ones; and indeed the mammalian visual system is full of what neuroscientists call "maps," whose two-dimensional topology corresponds roughly to the structure of retinal images (Damasio, 1999). But this fact is mitigated by the following considerations:

1. Like a highway map, what is found on a map must include some "symbology," which indicates what is found in a location.

2. There is no reason why map-like objects cannot occur as terms in a language-like system.

Because beliefs change, one might want to treat them as time-stamped. The only problem with this idea is that the majority of beliefs remain stable over time. Your knowledge of the layout of your neighborhood changes only as buildings are remodeled, streets change from one-way to two-way, and so forth. So most beliefs are simply stamped as "true now," where "now" means "now and for some time into the future."[5] Only a handful, such as beliefs about your current location, change constantly. An important distinction to make is between *corrections* and *updates,* the former being a replacement of a faulty belief, and the latter being a change in belief reflecting in a change in the world. In this paper I focus on corrections.

[5]Different beliefs will have different "expiration dates."

2.3 The Language

The notation I will be using for expressing belief content is a straightforward higher-order logic called *RCL*, which is like Montague's logic (1974, Dowty et al. 1981), augmented with mechanisms for referring to expressions of the language within the language. Most terms are written as symbols, numbers, or expressions of the form $f(t_1, \ldots, t_n)$, except for a little syntactic sugar for functions such as "+" that allows us to write terms of the form $t_1 + t_2$ rather than $+(t_1, t_2)$.

Formulas are terms of type `Boolean`. Truth-functional connectives are functions from tuples of `Boolean`s to `Boolean`. They are syntactically sugared as well. Intensional functions can be defined using the "^" and "˘" operators, whose semantics requires the notion of *possible world*. "Possibility" and "necessity" operators of various kinds can be defined by specifying accesibility relations among possible worlds using standard Kripkean devices (Cresswell and Hughes, 1996). In addition, it is possible to refer to *RCL* expressions in *RCL*. More on this below.

Although I describe the formal semantics of *RCL* carefully in appendix A, I will not describe any proof theory, any axioms or rules of inference. The reason is that there is no clear dividing line between inference and computation in general. A physical process may be characterized as a computation with respect to a decoding d if the process goes from a state s_1 to a state s_2 such that $d(s_1)$ is the input to the computation and $d(s_2)$ is the output (for a fuller explanation see McDermott 2001[ch. 5] and Chalmers 2011). The output may be a term of any kind. Sometimes the term is an array of numbers used to control an agent's motions, as when a predator uses computations to control its legs, head, and eyes in order to keep its body close to the ground and a prey animal in view. But most computations are only indirectly related to behavior, as when a submodule computes the amount by which the optical image of the prey animal has drifted from the predator's foveas. Such results can almost always be characterized as propositional attitudes; if we focus on beliefs, the computations that produce them are inferences.[6]

[6]In other cases, they can also be characterized as inferences, just more complex ones. The input of a planning module might be a goal G, and the input might be a term B denoting a piece of

At a later point it will be useful to have a predicate `rationale(`e`, `m`, `p`)`, where e is a judgment event, m is a module, and p is a set of propositions that m based its judgment e on. To explain what exactly these terms mean and how they are represented will have to wait until section 3.3.[7]

I will have nothing further to say about computational/inferential processes, except to introduce the symbol "$\vdash P$" to mean that an agent believes (subpersonally or at the person level) that P "now," i.e., until updated or corrected.

2.4 Vision

In this section I show how logic can be be applied to capturing the content of visual input. Further details may be found in appendix B.

The overall purpose of vision is to infer the shape, position, state of motion, and category of objects in an agent's vicinity; or on some occasions to control behavior directly. However, in this paper my focus is on visual appearance and how it corresponds to geometric reality, so I will ignore everything except what is called *early* vision, in which the shapes of surfaces are inferred, for their own sake or as a step toward figuring out the shapes of the objects enclosed by the surfaces. It is at this stage that many visual errors occur. Correcting such errors involves distinguishing appearance from reality, which requires a model of one's own perceptual apparatus. In the "standard self-model," this apparatus and its doings acquire the properties that we group under the heading "phenomenal consciousness."

Reconstructing the surface of an object requires finding "pieces of surface" in the image and collecting those that belong together in a single object. I'll use the fairly standard word *patch* [¡¿] for a piece of surface. A surface may have several

behavior; as a conclusion, it might be expressed as a "binary" propositional attitude, "I should make it be the case that P_1 if I want it to be the case that P_2," where P_1 = "I do B" and $P_2 = G$.

[7] It is a curious fact about Fodor's treatment of the language of thought (Fodor 1975 and many others) that he has never provided any examples of inferences involving this language except deductively valid ones, even though these must be comparatively rare. For instance, even syntactic parsing is not infallible and is hence nondeductive. If the LOT is really to be useful, it must be used by modules capable of making mistakes.

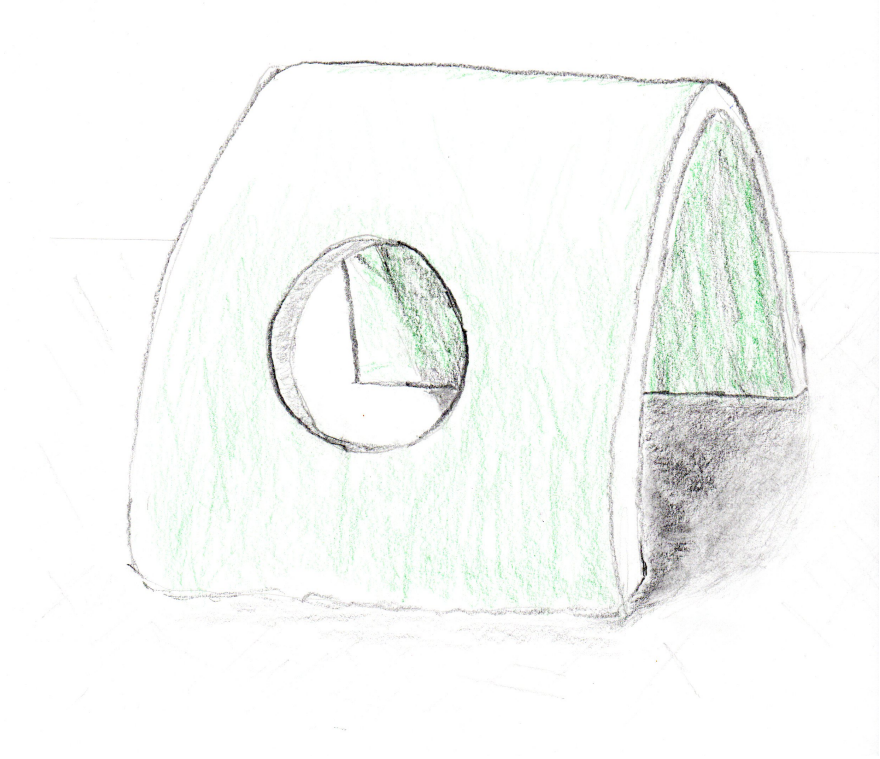


Figure 1: Arch-like object in context

boundaries (because of holes). Each boundary is a cycle of *boundary segments* corresponding to reflectance changes, surface-orientation discontinuities, occlusions, or shadows. Most interesting patches are from physical surfaces, but some are from light sources, or from the sky or other background entity at "infinite" depth, such as a mountain range. Figure 1 shows the image of a simple arch-like object; in figure 2 it has been extracted from its original context. Its visible surface is manifested through five *surface patches.*

The only patches that can be seen clearly at any moment are those that the eyes are looking directly at, so that their images fall in the high-resolution part of the retina of each eye, the *fovea.* As soon as the eye moves on to some other part of the world the patches in the fovea are rapidly forgotten. We're usually unaware of this amnesia because eye motions return the fovea to areas we're interested in, refreshing

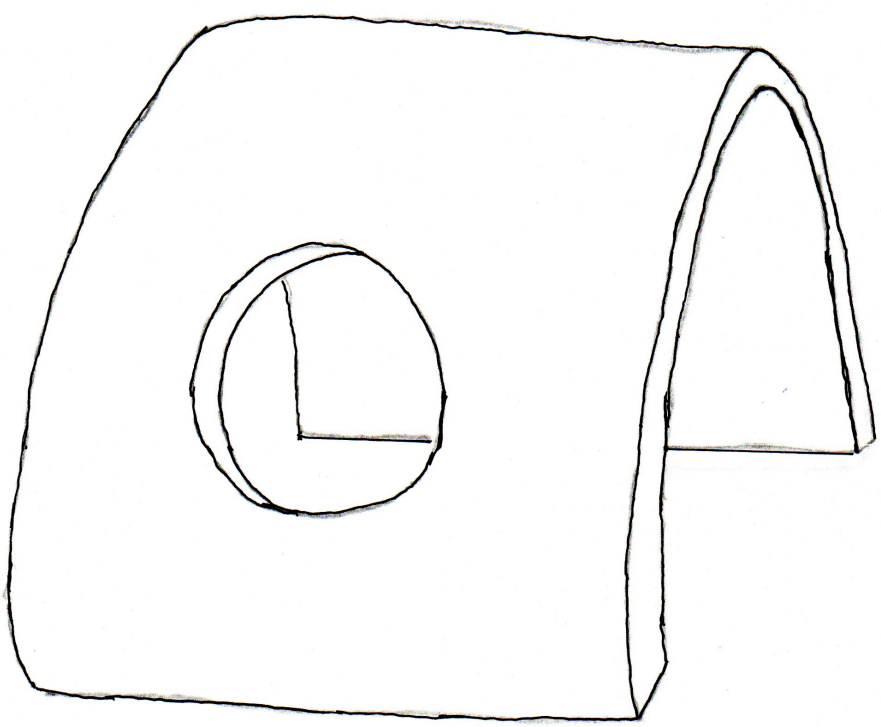


Figure 2: Object of figure 1, extracted

our information about them. The control of eye motions is mostly unconscious; everything in front of us that we want to see we can, if we're blessed with normal vision, so the visual world just seems to be a smooth "three-dimensional picture" of what lies in front of us.

Surface patches belong to *bodies.* How bodies are characterized is still poorly understood [¡¿], but a sometimes urgent distinction is between *animate* and *inanimate* bodies. Most bodies are at rest with respect to the ground, but some are moving, and the direction of motion of animate creatures is of obvious interest. Because the resolution of an object outside the fovea is so bad, the vision system has little chance of resolving a moving object unless it is *tracked*, that is, kept "foveated" as it moves. Of course, the system will often notice moving objects in the periphery and direct attention — and the fovea — to them.

To represent information about patches we will need formulas such as

```
patch_bounds(patch736859, [bound912644])
patch_boundary(bound912644, [...])
```

Brackets are used to delimit lists of objects. Further details will emerge in what follows; the full explanation of the notation and its application to vision is in the appendix.

One use to which all this information is put is to control the movements of the agent whose visual system we're talking about. The feedback loop can be quite tight, as when a quarterback's arm (and legs) are coupled to the foveated intended receiver of a pass. I have no idea what role representation plays in this feedback loop, if any, but it seems unlikely to take the form of predicate-calculus formulas. Even so, such formulas should be able to capture, as well as any other formalism, the content of whatever representation is used.

So far we could be talking about a system with no visual consciousness at all, such as a robot, or a person whose conscious attention is focused on something other than vision. In the next section, we bring consciousness into the picture.

3 The Logic of Qualia

In this, the key section of the paper, I deploy the logical language developed in section 2.3 in order to explore the structure of experience. But first, I lay out my approach to the topic of phenomenal consciousness.

3.1 The Experience Bias

I make the following fundamental claim:

Perception is not necessarily accompanied by experience.

My reason for believing this is the observation that it is possible to behave in surprisingly complex ways without apparently having any experience of the environment the behavior is shaped around. Robot cars can navigate through complex urban landscapes [¡DARPA challenge, Google¿], stopping at red lights and going when they're green. They don't, however, experience anything.[8]

But now look at the terms Lycan (1995b) uses to describe a truck driver navigating a familiar route while thinking of something else:

> Suppose he did in fact stop at a red light. Presumably the light looked red rather than green to him; that is the only reason he would have stopped. So ... he was presented with a red quale; a subregion of his visual field had redness as its phenomenal or qualitative character. (p. 250)

Here it is useful to ask, what is supposed to be the difference between robots and people when it comes to apparently nonconscious driving? The former can navigate without having any experiences; the latter apparently cannot, or Lycan would not have presumed that "... the light looked red rather than green to him; that is the only reason he would have stopped." But this assumption is arbitrary. There is an

[8]The usual formula is that "there is nothing it is like to be a robot car stopping for a red light." For reasons explained in (McDermott, 2001) and in section 4.2, this slogan does nothing for me, and I will do my best to avoid using it.

alternative reason he might have stopped, namely, the reason he shares with a robot car. *It* had to have had some reason to stop, too, but all we mean by that is that a sensor measured the stoplight in certain ways (luminance at various wavelengths relative to the rest of the scene; position on the pole of the brightest light), which led to the brakes being activated or not. But any attempt to find a quale of red, or even a visual field, inside a robot car is purely poetic.

Hence if you need a minimal reason why the truck driver "would have stopped," *no qualia need to be invoked.* Nonetheless, we tend to invoke them. I will use the term *experience bias* for this temptation to assume that in the case of people, as opposed to robots, control of behavior *must* involve experience. It's hard to give this intuition up, but I have yet to hear a convincing reason to accept it.

Georges Rey (1994) has thrown down the gauntlet on this issue, and endorsed what Dennett (1991) calls "Cartesian materialism." Rather than accept that the unconscious truck driver experiences nothing, he claims that the truck driver experiences traffic lights and other entities just as Lycan describes, but promptly forgets the experiences. At first glance there is no way to choose between the two theories. But I find examples like the following compelling: On my way into the building that houses my office, I find myself with an urge, but not an urgent urge, to use the bathroom facilities. I decide to drop my backpack off at my office first, then go to the bathroom. I resume thinking about some technical problem or other, and continue on my way along a path I have traversed a thousand times before. The next thing I know, I am standing in front of a standard piece of men's-room equipment, my backpack still on my back. Now, it is *possible* that I made a conscious decision to go to the bathroom before going to my office, but is it likely that I would consciously decide to change my mind, and later remember only the original decision, not the change in plan?

It is of course indisputable that people *do* sometimes experience things, and that the robots (and other computational systems) we have today do not, not even a little bit. I think what people do when they experience something is pay attention to their perceptual processes (Armstrong, 1968, McDermott, 2001). The structure of

experience is partly due to the concepts used to make sense of our own perceptions, i.e., to the *models* we have of our perceptual systems (Minsky, 1968, McDermott, 2001, Metzinger, 2003). Conscious experience happens somewhere in the zone of attention to our own perceptual processes, and our beliefs about them. In other words, it is a matter of "higher-order" cognition about sensory processes. This point has been argued for before by others (Lycan, 1987, Rosenthal, 1986, Dennett, 1991, Carruthers, 2005), and will be developed in several directions in the rest of this paper. Suffice it to say now that the idea explains phenomenal consciousness by enmeshing us in a belief system (McDermott, 2013) about how our perceptual processes work, a system which is inescapable because self-fulfilling and socially reinforced. When robots have analogous models of their own perceptual systems, they will exhibit phenomenal consciousness, too.

3.2 Beliefs About One's Perceptions

A key fact about human consciousness, and presumably animal consciousness, if such there be, is that one models one's own senses as "fields of seeming." If I stop to think about what I saw, what I thought I saw, and what I am seeing, I find myself in what we might call the *visual world.* This is the world as experienced from a particular point of view, ignoring all nonvisual input. It is not any image produced during the visual process; it's "cyclopean" in Julesz's (1971) sense, being the result of combining information from two eyes in a way we don't normally think about. It seems something like a stable image, but this stability persists as I move my head, eyes, and body about. If I hold my head in a fixed position in space, and think like a painter, I can flatten out the depth dimension and see the visual world as the *visual field,* but this two-dimensional entity is to a great extent an artificial cultural construct, and not the way we normally think of the world visually. The visual world as we normally experience it is three-dimensional.

The philosophical position known as *indirect realism*, now a minority position, holds that the construction of the visual world is a key step in perception (Peacocke, 1983). Indeed, the idea is that the visual world is itself perceived as a step in per-

(a). The Kanizsa-triangle illusion (b) The Federal Express logo

(Source for (a): `http://psychology.about.com/od/sensationandperception/ig/Optical-Illusions/Kanizsa-Triangle-Illusion.htm`)

Figure 3: Two examples of perception

ceiving the "external" world (Jackson, 1977). Brown (2008) uses the term *perceptual display* in roughly the way I use "visual world." He has in mind what we see in cases such as the Kanizsa-triangle illusion (figure 3(a)), in which a small black circle with a wedge cut out of it becomes a disk occluded by a triangular surface when two more notched circles are added whose wedges line up so as to make a triangle with the first wedge.

It's an interesting fact about the vision system that just knowing the surface patch seen in figure 3(a) isn't there doesn't make it go away. To that extent the system is "cognitively impenetrable" (Pylyshyn, 1985). Whereas in the Fedex[9] logo (figure 3(b)), once you've seen the arrow you can't stop seeing it. After seeing the nonexistent surface in figure 3(a), and not seeing the arrow-shaped patch in figure 3(b), one may contemplate the way they *seem* versus the way they *are*. Although these are both artificial cases, the Fedex case is more typical of reality: you do not see the bear in the underbrush until a friend points it out; then you evaluate whether the friend is correct, why the underbrush appeared bear-free at first glance, and what to look for next time (and what to do about the bear).

However, the Kanisza case is more attention-grabbing for us academics safely

[9]Trademark of Federal Express Corporation

ensconced in our armchairs, because of the way the illusion persists. It's not the only such case. Consider what happens when an animal encounters a mirror for the first time, and sees, apparently, a conspecific. Depending on its species, it reacts with amity or animosity, either of which, it turns out, is wrong. But the percept doesn't go away, at least not for human animals; we just learn to deal with, and even exploit, mirror images.

One can see why, if introspection were all one had to go on, one might believe that such displays are normally produced and then perceived — in the course of perception. The problem is that this scheme makes no computational sense. Your brain can't create the visual world without finding all the patches and bodies, so that almost all the work of perception must be done *before* the visual world is created.

I've chosen to discuss the human vision system because so much has been written on the phenomenology of color. I want to be as faithful to it as possible, but doing so would require us to engage with a lot of details about vision that are philosophically irrelevant. In appendix B I propose that patches correspond to areas where curvature and reflectance vary smoothly. The details of such smooth variation are too grubby, so I'm going to do what philosophers usually do, and pretend surface patches are uniformly colored. That is, instead of writing `patch_reflectance_map(`p`, `f`)`, where f is a representation of how reflectance varies across the two dimensions of patch p, I'll write `patch_color(`p`, color(`w`, `g`, `y`))`, where w is the total "lightness" of patch p; g is the percentage of green in the light it reflects; y the percentage of yellow. The percentage of red is $r = 1 - g$; the percentage of blue is $b = 1 - y$. (See appendix B for a discussion of the actual complexity.)

The arguments w, g, and y to the `color` function reflect facts revealed by psychophysics about human "color space"; pieces of surface can be compared for similarity, but also for distance in the "light-dark," "red-green" and "yellow-blue" directions. (Although the complete facts are more complex (Hardin, 1993).) At the subpersonal level there are no doubt computations before the determination of the relative amounts of red and green and of yellow and blue, but the results of these computations are not accessible to (human) psychophysics.

To make some tentative steps toward formalizing introspection suppose our intelligent agent has observed itself making a perceptual judgment. In this formula, the predicate `event` is the same one used to describe mundane events in space and time, such as yesterday's lunch. It is supported by a system of predicates expressing temporal and causal relationships among events, which I will not describe.

⊢ `event(ev301472, judgment(patch_bounded(patch209227,...)` $\wedge$ `...))`

This representation is obviously not right, for the usual reasons: The operator `judgment` is not truth-functional. The usual move at this point is to suppose it's intensional, i.e., that its argument is an intension, that is, an entity that varies from possible world to possible world (appendix A):

⊢ `event(ev301472, judgment(^(patch(patch209227,...)` $\wedge$ `...))).`

I claim this is still not right, because it is unsuitable for making statements about corrections. Suppose you discovered that your judgment that a certain car was grey was wrong (a trick of the light?) and the car was actually dark blue.[10]

⊢ `judgment_correction(ev301472, replace(^grey, ^dark_blue, ???))`

The function `replace` is to indicate how the judgment is to be edited. But there is no function on intensions that can do this sort of editing.

A workable solution is the mechanism of *internal quoting* developed in connection with programming languages. The syntax $'\alpha$, where α is a *RCL* expression, denotes α itself *qua* expression, that is, formal-linguistic object.[11] If α is of type τ, $'\alpha$ is of type `Expr<`τ`>`.

So we revise the parameter of `judgment` once again:

⊢ `event(ev301472, judgment('(patch(patch209227,...)` $\wedge$ `...)))`

[10] Constants such as `grey` and `dark_blue` are, we'll assume, known to be equal to `color` terms.

[11] See appendix A.7 for citations of the sources influencing my notation.

The expression `'(patch(...) ∧ ...)` is reminiscent of string expressions such as `"patch(...) ∧ ..."`, which denote the string of characters beginning `p`–`a`–`t`–`c`–`(`–`·`–`·`–`·`. But strings of characters aren't natural tools for expressing judgments about judgments. Characters are the sorts of things that appear in books and newspapers. One can have all sorts of beliefs about them, such as the (all too common) belief that "accommodate" is spelled with just one "m." But it's extremely unlikely that characters are used inside your head to express beliefs, and downright impossible for an illiterate. In fact, given the nature of our hermeneutic enterprise, we're trying not to take a position (at least, not prematurely) about exactly how beliefs are expressed inside your head.

This is why the quoted expression `'(patch(...))` is so much better than character strings. It denotes an expression *in the very language used to write expressions*. It requires no further ontological commitment beyond our hermeneutic decision to decode physical structures as formal expressions in the first place. (For the formal details about quotation, see appendix A.)

Within a quoted expression we can "unquote" subexpressions. If `e` is of type `Expr<Integer>` and `f` is of type `Integer→Integer`, then `'(f(~e))` is the expression obtained by substituting whatever `e` denotes into the argument position of the expression `f(...)`. It is of type `Expr<Integer>`. We can use `~(...)` to quantify into a quoted context:

```
∃(c :  Expr<Color>)
 (event(e503282,
  judgment('(patch_color(p996080, ~c))))
 ∧ event(e521256,
         judgment('(patch_color(p996563, ~c)))))
```

"There is some color that patches `p996080` and `p995563` were both judged to have."

Now all we need to make editing of judgments possible is a way of picking out subexpressions. A straightforward tool is the *path designator*, a sequence of natural numbers that identifies a subexpression in an abstract-syntax tree by specifying the path to it: $[n_1, \ldots, n_k]$ designates the n_k'th child of the n_{k-1}'th child of its

$\ldots n_1$'st child, using the abstract syntax developed in appendix A.1. (The null path expression `[]` picks out the expression you start at.)

The term `replace(`p`,` e_{new}`)`, where p is a path designator and e_{new} is an expression, denotes the operation of replacing the subexpression designated by p with e_{new}, as in

```
⊢ judgment_correction(ev301472,
                      replace([...,1,2], 'dark_blue))
```

where the path designator `[...,1,2]` denotes the occurrence of `grey` that is to be replaced.

There is a limit to how far introspection can be pushed. The problem with my representation so far is that another thing we have no conscious access to are the components (i.e., w, g, and y) of the representation of colors inside our heads. All we can do is compare two or more `color` terms in various ways. The device of quoting *RCL* expressions within *RCL* is too revealing.

So I introduce yet another piece of notation: $^{\bullet}e$ is a "virtual name" for object e. If e is of type τ, $^{\bullet}e$ is of type `Expr<`τ`>`, and so is suitable to appear "unquoted" inside a quote. Suppose an agent has made the following judgment:

```
⊢ patch_color(p382670, color(38, 0.6, 02))
```

Its model of this judgment might be as follows

```
⊢ event(e937707,
        judgment('(patch_color(p382670,
                               ~•(color(38, 0.6, 0.2))))))
```

where the judgment is visible, but only down to the level of the flagged term, not inside it. The color term behaves as if atomic. Any path designator for `•(color(...))` (or an expression containing it) is acceptable; but no path expression can lead to the arguments of the `color` function. The effect is that an occurrence of $^{\bullet}e$ allows conclusions to be drawn about e the object, but not about e the term. The logical reconstruction of vision and other senses has to involve numbers, but it has to

"encapsulate" some of them. The term `color(`w`, `g`, `y`)` contains three numbers, as it inevitably must to capture the computations that the vision system does in disentangling surface reflectance from illumination. (See appendix B.) However, our analysis also has to capture the fact that such numbers are not directly accessible; our intuitive insight goes no deeper than judgments such as those above. The terms are closed, or *locked.*[12]

[[[Relationship to phenomenal concepts]]]

Let's use these notations to explore the logical structure of conscious insights. Suppose one is asked to estimate the relationship between two judgments made at approximately the same time:

```
⊢event(e445125, judgment('(reflectance(p300183, ~•(color(87,0.35,0.05))))))
⊢event(e445136, judgment('(reflectance(p300243, ~•(color(85,0.3,0.08))))))
```

(The two patches `p300183` and `p300243` could be foveated simultaneously or in succession.) One might be asked, Which is redder? And the answer might be `p300243`, whose g component is smaller. In fact, both would look purple to someone with normal color vision, between blue and violet (because $g < 0.5$ and $y < 0.5$), but the one with smaller g looks redder. (If $g \approx 0.5$ then the color is pure blue, yellow, or white.)

Why people can be brought to judge that a certain surface looks red, and that one surface looks redder than another, are matters for psychological theory. What I would like to call attention to is that these judgments are *about other judgments.* As usual, I do not try to guess how these meta-judgments are made, but concentrate on how to represent them. We simply know from psychophysics that normally-sighted people are able to produce conclusions such as

[12]It might seem implausible that a "fat dot" is all that's needed to prevent the system from looking into the term that follows it — a dot that isn't even present in the original, unquoted judgment. But remember that all the formulas shown here are for our eyes only. The system, even if it actually contains a formal language isomorphic to this one, is not "reading" it the way we are. The formulas — or, more likely, the representations whose content the formulas capture — are *used* by the brain, not *read.* That is, they are grist for a mechanical mill, whose principles of operation are nothing like intelligent reading.

$\vdash$`j_dissimilarity(`e_1`, `p_1`, •(color(...)), `e_2`, `p_2`, •(color(...)))`
`> j_dissimilarity(`e_3`, `p_3`, •(color(...)), `e_2`, `p_2`, •(color(...)))`

and

$\vdash$`j_redder(e445136, p300243, •(color(...)),`
`e445125, p300183, •(color(...)))`

The prefixed "`j_`" is to flag (for mnemonic purposes) that these are meta-judgments; what actually makes them meta-judgments is that they are about events that are themselves judgments.

The need for locked terms points up a general fact about the conscious human mind: the concept of real number doesn't come naturally to it, but entities *reminiscent* of real numbers are not rare. For example, we can judge that one stick is longer than another stick (sometimes with great confidence, sometimes with less). The computational analysis of visual length estimates would almost certainly involve numbers, but that doesn't mean that to compare the lengths of two sticks we consciously generate numbers from images and compare *those*. Instead, it's the other way around: we can get numbers from visual judgments of length, to the extent we can, by abstracting from length comparisons.

It may seem as if psychophysical results (such as those of [¡Sternheim and Boynton¿]) show that people do have access to the components of a color. In fact, they can produce comparisons of colors along *many* dimensions, not just those (whatever they are) that occur within a term like `•(color(...))`. My haphazard choice of the three dimensions w, g, and y as defined above has almost no weight, but it's not implausible that colors are represented using *some* small set of numbers (at least three, in people with normal color vision). These can be thought of as coordinates in a given basis of color space.

When it comes to senses other than vision, we know less about "quality spaces." Presumably smells, sounds, pains, and other sensations each have computational representations, but no one is in a position even to say how many dimensions the

spaces of possible smells, sounds, and pains have.[13] All we have access to are quoted terms and similarity judgments.

There is, however, a key difference between length on the one hand and color, smell, and sound on the other, and that is that lengths occur in more than one dimension, and colors, smells, and sounds do not. One can compare not just the lengths of sticks, but also the length of a stick and the height of a person. An object's height can be compared to its width. It's even possible to compare an object's height to the distance to some faraway point. It's easy to imagine Alice growing until she's as tall as a house.

In more formal terms, let's suppose the numbers representing lengths are encapsulated inside terms of the form `len(`l`)`, where l is length in meters. Obviously we don't use the metric system inside our heads,[14] but that's irrelevant. We still must "lock" the terms representing lengths:

$$\vdash \texttt{j\_longer}(e_1, {}^{\bullet}\texttt{len}(l_1), e_2, {}^{\bullet}\texttt{len}(l_2))$$

where e_1 and e_2 are perceptual events in which lengths `len(`l_1`)` and `len(`l_2`)` occur.

However, locked or not, terms such as `len(`l_1`)` and `len(`l_2`)` don't both have to come from judgments made along the same axis of physical space, whereas the analogous statement about color judgments is precisely true. Our vision system parses reflectance judgments along (at least) three axes, each of which uses different units, as it were. There is nothing in color space that works the way length does in physical space.

[13] It is quite common in some cog-sci circles to emphasize that sensory spaces have dazzingly many dimensions (Edelman and Tononi, 2000, Churchland and Churchland, 1997)[¡¿]. (Cf. section 4.3, below.) Whereas computational research into vision tends to focus on finding the low-dimensional surfaces where percepts actually reside (LeCun et al., 1989, Hinton, 2007), in order to make the problems of finding patterns more tractable. I fear the former camp have a subliminal hope that the laws of computational complexity are repealed inside the human skull, but, as for other laws, this hope is probably in vain.

[14] It is not even necessary that the units for short lengths (manipulable by hand) are the same as for long (walkable), so long as they can be compared.

Physical space is special because our bodies can move about in it; you can collide with things and lose things. The further away something is, the less you have to worry about it, most of the time; unless it is something valuable that you need to interact with again, or *it's* looking for *you*. A robot or animal must think about translation in Euclidean space because it has to relocate its base camp at times. It must think about rotation because it can rotate; about reflection, because when talking to a conspecific (or some other almost-symmetrical creature), its *left* is their *right* and vice versa. (In this case the frame of reference is the agent's own body.)

To describe a color to someone, you relate it to an object whose color is known to both you and your interlocutor. You might say that something has a very red orange color, like those orange markers in the Candyland game[15] that are so irritatingly easy to confuse with the red markers. Assuming your interlocutor has played Candyland,[16] they will know what you mean; the axis "red–orange" doesn't shift around too much relative to speaker and hearer. The commonalities among people are not perfect, but sufficient to support the usual color vocabulary.

The key point is that we can move around in physical space, and rotate our point of view, but each of us is more or less stuck at one point in color space or smell space, except for involuntary changes due to aging or accident. This is a central difference between primary and secondary qualities.

However, there is another important, but subtle, difference between the two cases. Our scientific theories are expressed using length, mass, and time intervals. The universe turns out to depend on them in such a basic way that it would be surprising if creatures that evolved in it couldn't sense them. But now consider surface reflectance. It is important to sense the reflectance of surfaces, for various reasons, but the way light interacts with and bounces off surfaces is so complex that any sensing scheme is sure to fail to capture some aspect of that interaction (Byrne and Hilbert, 2003). Fortunately, one doesn't need a perfect reflectance sensor. Given a way to estimate illumination, any wavelength-sensitive light sensor can be considered to be probing the projection of the reflectance along one dimension in "reflectance

[15]Trademark of Hasbro Corporation.

[16]As a small child, or with one.

space." Our three photopigments give us the ability to measure projections along three independent vectors. This is sufficient for tasks such as finding ripe fruit among green leaves [¡¿], or differentiating shading cues from changes in reflectance [¡Holtman-Rice¿].

The important but subtle difference between length/width/height and hue/-saturation/intensity is that the former capture all the dimensions of physical space, using just one kind of unit, whereas the latter capture only three dimensions of reflectance space, a fairly arbitrary three dimensions.

3.3 Qualia

I've used the phrase "secondary quality" to refer to locked `color` expressions (and similarly for other senses). It is tempting to identify locked expressions such as •`(color(85, 0.3, 0.08))` as qualia, the ways experiences feel. But I don't think that's quite right. Qualia are supposed to explain, among other things, how I can tell the difference between, say, red things and green things (or experiences of red and experiences of green). What is it *about* •`(color(85, 0.3, 0.08))` that enables me to distinguish it so readily from •`(color(83, 0.8, 0.05))`? The actual answer is to be found in neural machinery that is not accessible to our internal scanner, but it seems as if there is another answer, just one we can't put into words.

Compare this question: What is it *about* the length of a toothpick that enables me to distinguish it so readily from the height of a tree? Not what is it about the toothpick and the tree — the answer to that question is, "Their lengths." The question is, "What is it about the *lengths* that enables me to compare them?," and that question seems absurd. Whereas the analogous question about color, "What is it about redness that distinguishes it from greenness?," does seem as if it *ought* to have an answer. We really can't come up with any *evidence* why one surface patch looks redder than another, just descriptions such as this, which could come from almost any paper on the subject:

> Qualia are the subjective or qualitative properties of experiences. What it feels like, experientially, to see a red rose is different from what

> it feels like to see a yellow rose. Likewise for hearing a musical note played by a piano and hearing the same musical note played by a tuba. The qualia of these experiences are what give each of them its characteristic "feel" *and also what distinguish them from one another.* (Kind, 2008, emphasis added)

Or refer back to the Lycan quote in section 3.1 (p. 13), where he argues that the "only reason" the driver would have stopped was that a light "looked red rather than green," and the reason for *that* was its "red quale."

To formalize this idea, I propose a type (schema) $\texttt{Quale}_\tau$, "sensory quality of type τ, as in $\texttt{Quale}_{\texttt{Color}}$, and axioms such as

$$\begin{array}{l}
\forall\texttt{(x : Color)} \\
\quad \exists\texttt{(q : Quale}_{\texttt{Color}}\texttt{)} \\
\quad\quad \forall\texttt{(e : Event, p : Patch)} \\
\quad\quad\quad \texttt{(event(e, judgment('(patch_color(p, }\sim^{\bullet}x\texttt{))))} \\
\quad\quad\quad\quad \supset\texttt{(has_quality(x, q)} \\
\quad\quad\quad\quad\quad \wedge\ \texttt{rationale(e, self_scan, \{\^{}(has_quality(x, q))\})))}
\end{array}$$

The predicate $\texttt{has\_quality}(x, q)$ expresses the idea that an object x inside a "$\bullet$" has a quale q. The predicate $\texttt{rationale}(e, m, r)$, introduced in section 2.3, is supposed to supply the basis for a judgment event. The rationale r is of type "set of `Prop`" (see appendix A.6); the "$\hat{}$" transforms expressions of type `Boolean` into expressions of type `Prop`.

Rather than continue to use existential quantifiers, it will be more revealing to introduce a family of "Skolem functions" [¡¿] $\texttt{qual}_\tau$ of type $\tau \rightarrow \texttt{Quale}_\tau$. The axiom above then becomes

$$\begin{array}{l}
\forall\texttt{(x : Color, e : Event, p : Patch)} \\
\quad \texttt{(event(e, judgment('(patch_color(p, }\sim^{\bullet}\texttt{x))))} \\
\quad\quad \supset\texttt{(has_quality(x, qual}_{\texttt{Color}}\texttt{(x))} \\
\quad\quad\quad \wedge\ \texttt{rationale(e, \^{}(has_quality(x, qual}_{\texttt{Color}}\texttt{(x))))))}
\end{array}$$

The term $\texttt{qual}_{\texttt{Color}}(c)$ is supposed to denote the "quale" of color c.[17] Qualia supposedly provide the rationales for all sorts of judgments. Given two terms ${}^{\bullet}x_1$ and ${}^{\bullet}x_2$ that an agent can discriminate, it is their qualia that enables the agent to make the discrimination:

$$
\begin{aligned}
&\forall(\texttt{x1 : Color, x2 : Color, e1 : Event, e2: Event, e3 : Event,}\\
&\quad\texttt{p1 : Patch, p2 : Patch)}\\
&\texttt{(event(e1, judgment('(patch\_color(p1, }{}^{\sim\bullet}\texttt{x1))))}\\
&\quad\wedge\ \texttt{event(e2, judgment('(patch\_color(p2, }{}^{\sim\bullet}\texttt{x2))))}\\
&\quad\wedge\ \texttt{event(e3, judgment('(j\_redder(e1, p1, }{}^{\sim\bullet}\texttt{x1, e2, p2, }{}^{\sim\bullet}\texttt{x2))))}\\
&\quad\supset\ \texttt{rationale(e3, self\_scan, \^{}(has\_quality(x1, qual}_{\texttt{Color}}\texttt{(x1))}\\
&\qquad\qquad\qquad\qquad\wedge\ \texttt{has\_quality(x2, qual}_{\texttt{Color}}\texttt{(x2)))))}
\end{aligned}
$$

It is not at all clear what the point of the `rationale` predicate is here. Part of its point is that it has no point. If qualia play a role in the rationales for sense discriminations, there is no obvious further rationales for judgments involving them. If their natures are self-evident, so that no further rationales are required for discriminations involving them, why bother to invoke them at all? Why must there be anything connected with, say, ${}^{\bullet}$`(taste(a,m,d,g,...))` that explains what it tastes like?

We live in the physical world, but as conscious beings we also must deal with the *sensory world,* the way things *seem.* When the first ape (or crow, or octopus) distinguished the way things are from the way things seem, they had the visual world and other such "perceptual displays" to deal with. These were marvelous and useful inventions, which enable animals to model, and compensate for, the errors of their own perceptual systems, but they reveal only so much.

I suggest that qualia arise from a flaw in this invention. In thinking about the self, people have to resort to metaphors, and the one they often fall back on is to see the relationship of self to agent in terms of the relationship of agent to world. In other words, they embrace some form of "homunculism," in which a person-like

[17] With a more expressive type system we could introduce a single function `qual` of type $\forall\alpha(\alpha \rightarrow$ `Quale<`α`>`) (Pierce, 2002).

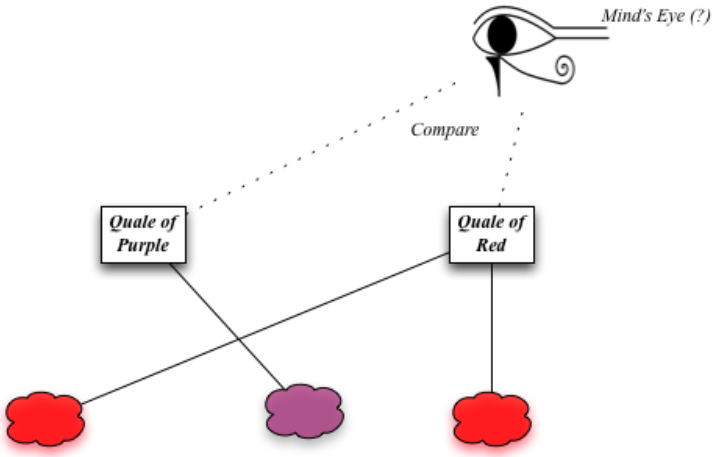


Figure 4: How we think qualia work

entity inside the body explains how personhood works. The idea that the qualia of sensations play a role in distinguishing them may be caricatured as in figure 4. According to this model, judging the relationship between an experienced purple and an experienced red is accomplished by "looking" at them, deriving their qualia, and comparing them. The net effect is to recycle locked terms to explain themselves. The quale of red is like meta-redness, a sort of paint that gets applied to patches labeled red in the visual world to *make* them red.[18] The function $\texttt{qual}_{\texttt{Color}}(c)$ is the suchness of color c, that which makes it the color that it is.

In the terminology of McDermott (2013), belief in qualia are "IFIs," *irresistible framework intuitions.* Such intuitions are not necessarily true, but their truth is irrelevant to the near-compulsion to accept them. Developing a semantics to account for such entities is an important problem, which is beyond the scope of this paper.[19]

In (McDermott, 2001), I speculated that any intelligent agent would have to have

[18]Dennett calls this paint "figment" (Dennett, 1991), p. [[¡¿]].

[19]Solving it would solve a key piece of the problem of intentionality, namely intentional attitudes toward fictional entities (**?**).

qualia, because qualia are simply where explanations about perception cease. I now doubt that. For one thing, there are cases where invoking qualia even in our own case seems wrong; does the difference between left and right feel like it's mediated by the "quality" of the two directions? [[[Need better example, not so closely tied to primary qualities.]]] For another, the formal model developed here provides an alternative stopping point, at locked secondary-quality terms. In fact, it seems like getting a robot to believe in qualitative properties would require implementing the mental glitch exemplified by figure 4, of interest to someone trying to build a model of human psychology, but not because it serves any purpose.

4 Solving Qualia Puzzles

My principal test for the power of the theory described here is how it solves puzzles about what it's like to be a bat (Nagel, 1974) and what one learns when one learns what something is like (Jackson, 1982, 1986). But first, let's look at a fundamental problem with the idea of "self-scanning."

4.1 The scanner problem

David Armstrong (1968) and William Lycan (1987, 1996) treat introspection as a key factor in their theories of consciousness, and analyze introspection as *internal perception*, or *scanning*. We can schematize the idea as in figure 5. Here A is a perceptual module that performs some sort of analysis of input and passes along its conclusions to C. This is the "normal" flow of sensory information. Module B is the scanner that detects what A is doing; it has "introspective" access to A's outputs.[20]

I am sympathetic to this analysis, making consciousness a matter of higher-order perception rather than higher-order "thought" Rosenthal (1986).[21] However,

[20]The terms *normal* and *introspective access* are mine McDermott (2001)[[[Or does Armstrong essentially use the same terms?]]].

[21]My main problem with Rosenthal's version is that he appeals too much to intuition regarding the nature of thoughts; if an unconscious thought is not a computational entity then I have no idea what it is. Let it be noted that Lycan and perhaps Armstrong do not believe *phenomenal*

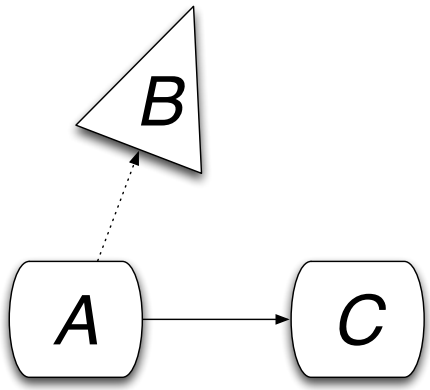


Figure 5: Introspective scanning (according to Armstrong and Lycan): Module B scans the output from A to C

in an influential series of papers (Shoemaker, 1994a,b,c), Sydney Shoemaker has raised some stark objections to the self-scanning view of consciousness. He lists no less than eight different features of real perception that he argues don't apply to the putative "inner sense": (1) Sense perception involves an "organ of perception" (eyes, ears, and such); (2) there are "sense-experiences" distinct from the object of perception; (3) awareness of facts comes about through awareness of objects and one's relationship to them; (4) perception requires ways of tracking objects over time, which in turn requires criteria for re-identifying them; (5) perception of objects requires perception of their "intrinsic, nonrelational" properties (such as colors and shapes); (6) objects of perception are potential objects of attention and inattention; (7) perceptual beliefs are causally produced via reliable mechanisms; (8) the objects and states that are perceived exist independently of being perceived.

In each of these cases, I believe either that Shoemaker is wrong about ordinary perception, or that the feature in question applies to inner sensing as well as normal

consciousness (i.e., qualia) can be explained in terms of introspection, which distinguishes their positions from mine.

sensing. In what follows I repeat each objection in condensed form, together with a quick reply:

(1) Organs of perception

The idea that there are "organs" that deliver sensory information to the mind is somewhat old-fashioned. Information processing begins as soon as energy is transduced by sensor cells; there is no particular boundary with sensing on one side and information processing on the other (the brain side) (Akins, 1996).

(2) Sense experiences distinct from objects of perception

In my model of sensing, sense experiences are not part of the normal flow of sensory information processing at all; the belief that they are is just the experience bias again.

(3) Awareness of facts = awareness of objects and one's relationship to them

Of course inner sensing is not like sensing an object, in that you don't have to get into some position to do it. Shoemaker's main counterexample concerns sensing the self, which is puzzling, because in the normal course of events it is not necessary to attend to the "self" in order to attend to a sensory event.

(4) Tracking objects

As Millikan (1984, ch. 15) has argued, both internal and external sensing require knowing when to reuse a term in the language of thought (or some equivalent system). The same issues arise when reasoning about judgment events, although the opportunities for error are much rarer. You can realize that a pain you are having now has been going on all through a busy day, even though the events of that day may have distracted you from it until evening.

(5) Perception requires sensing intrinsic, nonrelational properties

My purpose in this paper is to "deconstruct" the idea of "intrinsic, nonrelational" properties, to explain what they really are, rather than what our naive intuitions tell us they are. See section 3.3. As I hope I've made clear by now, it's unlikely that recognizing such properties is part of ordinary, nonconscious, perception-guided behavior; they are recognized only in the course of reasoning about percepts.

(6) Perceived objects can be attended to	Perceptual events are indeed possible objects of attention and inattention, but they can go by quickly.
(7) Perceptual beliefs causally produced via reliable mechanisms	I don't understand this objection; it seems as if inner perceptions are caused by mechanisms that are at least as reliable as perceptions of physical objects.
(8) Objects exist independently of being perceived	It just seems obvious that most of what goes on in our minds is unperceived and hence exists independently of being perceived. The most persuasive case to the contrary can be made for pain, but I think this is because a key property of pain is that it demands attention. It's hard to argue that pain is often unfelt (let alone normally unfelt), because pain is a (hopefully) rare state that, when it does occur, keeps coming to consciousness. However, minor pains are often forgotten, and even major ones can be forgotten some of the time. Even people in constant pain have to sleep eventually.

In most of these objections Prof. Shoemaker is displaying the experience bias, the idea that most of what goes on in our minds involves experience. It's hard to get rid of, not just, because when we think about what we're doing, we see consciousness everywhere, but because it's unpleasant to think of ourselves as automata.

However, I think there is another problem with the Armstrong-Lycan model, that (somewhat miraculously) does not appear on Shoemaker's list. In figure 5, what is the signification of the dotted arrow joining A to B, as opposed to the solid arrow joining A to C? In each case there is a causal link: whatever is going on in B and C is partially caused by events in A. Can we say therefore that C is perceiving A? Obviously we'd rather not; C is perceiving the world, or rather the flow of which A→C is a part is the brain's mechanism for perceiving the world. The flow from A to B, on the other hand, *is* part of the perception of the brain by the brain. What's the difference?

The answer is sketched in figure 6, in which the arrows are labeled with information conveyed by the channels they represent. The difference is that the information

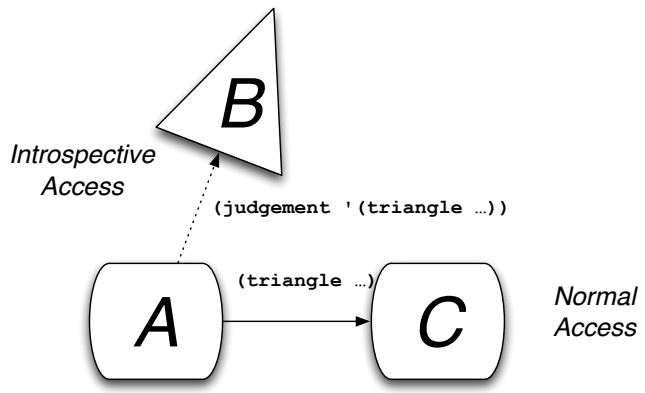


Figure 6: Differentiating introspective from "normal" access

conveyed from A to C is *quoted* as part of the information conveyed from A to B. End of story.

As I've said before, it is not necessary to believe that the signals alluded to in figure 6 are literally in a language of thought, let alone one isomorphic to *RCL*. What's important is that whatever the medium is, it contains a quotation device that functions semantically the way quotation works in *RCL* (see appendix A.5).

It may be that simple information flow is too simple to fit Prof. Shoemaker's definition of "perception." But what else is perception but a causal flow from objects in the environment to the brain? There is a point where the energy or matter emanating from an object directly is converted to neural signals, but this hardly seems like a watershed that absolutely must be present to talk of perception.

4.2 What things are like

And now to the puzzles about what things are like. To begin with, I register my protest against the entire "what it's like" formula. I believe this phrase has two legitimate uses:

1. As a request for information: When one wants to know more about an experi-

ence, one might ask someone who has had it to compare it to other experiences. Q: "What's prison like?" A: "It's no worse than boarding school."

2. As a claim to having been through something: "I know what it's like to be in prison" might be true because the speaker has been in prison. "You don't know what it's like to do X" means "You haven't done X." "You can't know what it's like to undergo Y" might mean "It's impossible for you to undergo Y." (As when a woman says to a man, "You can't know what it's like to give birth.")

In neither of these cases is it necessary to appeal to the existence of a special *fact* concerning the phenomenal quality of experiencing something. Hence if someone wants to claim that it is like something to be a bat, I refuse to grant that this claim makes sense. The word "like" implies a comparison. The *bat* certainly has nothing to compare being itself to; and it is impossible (surely) for a person to become a bat and report back later in any serious sense.[22]

However, these considerations don't dispose of Jackson's celebrated "knowledge argument" (Jackson, 1982, 1986), but the analysis of section 3 will help do that. To quickly review the argument: Mary is a scientist specializing in human color vision. In an experiment that made it through the Instutional Review Board by the skin of its teeth, she was raised from childhood with special black-and-white TV cameras installed in her eyes so that she sees everything in grayscale.[23] It is somewhat ironic that she chooses to major in human perceptual psychology; and slightly perverse in that we assume cognitive science has succeeded prodigiously, so that everything — all physical information — there is to know about human color vision is known, and

[22]But for those who need more convincing: There would have to be a sense in which one's identity could be preserved through the transformation to bat, and possibly back; but there isn't. If you want to claim that there is a fictional narrative with respect to which personal identity is preserved, this I will cheerfully grant; but one could spin the tale so that being a bat was like anything you wanted, so the possibility of such a story does not constrain the truth of the matter enough. I believe I recall going through this reasoning as a child, only I was thinking about what it was like to fly like a bird.

[23]Improvement on Jackson's original scenario, per suggestion by Dennett 2006.

specifically known to Mary. On her twenty-first birthday the cameras are removed and she is allowed to see colored objects. She learns how colors look. She may specifically find out what they are called, so that, having heard the words "red" and "green" all her life, she can finally attach qualia to those words; she finds out what red and green actually look like. But by hypothesis she possessed all physical information; therefore what colors look like — their qualia — are nonphysical facts; therefore physicalism is false.[24]

As discussed in section 4.1, I subscribe to a version of Lycan's "inner perception" theory of phenomenal consciousness (Lycan, 1987, 1995a, 1997). There are two ways to make use of a raw sensory fact: either it is brought to the attention of the agent *qua* person, or it is handled entirely subpersonally. In the second case, I think it is misleading to assume any qualia are involved. It's the former case that concerns us, because Mary is indeed interested at the personal level in the new colors she is experiencing. The dimensions along which sensory data can be classified at this level can be many or few depending on the sensory channel. But the new labels generated for the data become new vocabulary items in Mary's language of thought (Millikan, 1984, Lycan, 1996).[25]

In section 3.3, I used terms of the form $\texttt{qual}_{\texttt{Color}}(c)$ to be these new vocabulary items. These labels are part of a system of "irresistible framework intuitions," a compelling but shallow system of beliefs regarding how colors are distinguished. The shallowness is what makes color qualia "ineffable." The beliefs attach to these symbols are not conclusions in any reasoning scheme, but premises conjured up from nowhere. So the reason why Mary's story is no threat to physicalism is that she has acquired new beliefs, but not *justified* beliefs. She believes that $\texttt{qual}_{\texttt{Color}}(\texttt{red})$ is different from $\texttt{qual}_{\texttt{Color}}(\texttt{green})$ "intrinsically," not because of any relation to anything, but because otherwise the difference between `red` and `green` would be inexplicable.

[24] As noted, it is misleading to phrase what she learned as "what it's like to see colors," because a sentence such as "Mary learned what it was like to see red" just means "Mary saw red for the first time."

[25] If Mary's internal representation of this material is "map-like" rather than "formula-like," then we would say instead that new territory has been added to the map.

Mary knows all of this in advance. So she knows she will come to have certain unanchored but compelling beliefs. She might or might not know the form the new items in her mental vocabulary will take. It's highly unlikely that her brain will literally contains Skolem terms. Perhaps random symbols, such as `Q101714` or some neural analogue, will be assigned instead. What she doesn't know is what it will be like to have hard-to-resist beliefs about these entities.

Here's an analogy I find useful. Suppose Mary were an intelligent duckling, and knew everything there was to know about "imprinting" [¡¿] on the first animal she sees. Nonetheless, she wouldn't be able to predict which animal that would be or how her life will change when she sees it. Of course, the animal she imprints on is not quasi-fictional the way a quale is. But the belief that that animal is her "mother" (or caregiver) is just as *unmotivated* as the belief in qualia. If it seems farfetched that an intelligent animal could imprint the way ducklings do, consider how new parents imprint on their babies. (We prefer to use the term "bonding.")

Don't get me wrong: I am not saying there is no reason for all the beliefs one has about one's sensory data. There is a good reason to believe one sound is louder than another, and, within the (rather arbitrary) framework of human color vision, there is a reason why green is more like yellow than like red. What is unmotivated is the idea that there are such entities as the qualia of red, green, and yellow that justify such judgments.

4.3 Spectrum Inversion

Puzzle number three is the perennial childhood favorite about whether what you see when you see a green object is the same as what I see when I see a red object. In the language of qualia this becomes the question whether the quale of your experiences of green objects is the same as the quale of my experiences of red objects. More generally, it is this complex of puzzles: whether two persons' qualia of the same sensory experience could be different, whether we could ever know, and whether the question is even meaningful.

Initially one might suppose two qualia could simply have been switched around

before birth inside one person's brain, making them different from normal people in this regard. Red and green could be interchanged, or bitter and sweet. However, cognitive scientists are perfectly well aware that the "intrinsic" properties of qualia are really derived from the numbers (= levels of neural activity) that we are treating as residing inside locked terms. Two such terms are more similar the closer their numbers are. To the degree that sensations are rated as more or less pleasant, this axis (or axes) is also determined by the numbers. So a simple switch of red for green or bitter for sweet won't work without other adjustments to keep similarity and pleasantness judgments unchanged.

Hence all recent thinkers who take spectrum inversion to be a real possibility agree that it would be possible only if there were some transformation of color space that involved many colors, such that if two colors were neighbors in the pre-transformation space, their transformed versions would be neighbors in the post-transformation space. Some writers believe there simply is no such transformation. [¡Byrne? Kalderon?¿][[[[26]]]]

However, even if it turns out not to be possible for humans to undergo spectrum inversion, all that is required is that it be possible among some race of possible creatures. [[[Cite Armstrong, Shoemaker]]] Let's envisage a planet of monochromats who use vision only to spot flat gray shapes moving about on a dimly, diffusely illuminated translucent screen. They live on the dark side of the screen; why they need to perceive the shapes and nothing else is unimportant. Call them "Cavians." There is a darkest dark shape, and a brightest portion of the screen, and Cavians call these colors "black" and "white." Perhaps they have names for the shades of gray in between. For centuries wise females among them have speculated whether some of them see white when normal folk see black and vice versa. Other females suppose that such questions are unanswerable or meaningless.[27]

[26]Less has been written about senses other than vision, although our ignorance about how these are represented neurologically makes them more fertile grounds for speculation. Could the quale of bitter and sour be switched in some people? Or is it part of the quale of sourness that it makes your lips pucker?

[27]Cavian males are too busy fighting over mates to worry about such "womanly" questions.

As Cavian scientists begin to understand the brain better and better, the possiblity begins to emerge that perhaps the questions are answerable after all. Then there is a breakthrough: it is discovered that there is a straightforward physical state type G that represents the gray level of a point or patch in a visual scene. Tokens of this type are mapped by a decoding function d into a number between 0 and 1, 0 represeneting black and 1 white. (See section 2.1.)

Further research yields a bombshell: It is discovered that about 1% of the Cavian population is in fact inverted! That is, when exposed to a stimulus that they and virtually all other Cavians agree is a dark gray circle on a near-white background, the piece of their visual cortex representing the circle is in state s (of type G) such that $d(s) \approx 1$ and the piece representing the background is in a state s' such that $d(s') \approx 0$.

Philosophers such as Sydney Shoemaker (1981, 1997) have argued that Cavians are logically possible, so that presumably 1% of them would indeed be inverted with respect to the "normals": Their qualia of black are the same as the normals' qualia of white.[28]

I think this conclusion is not as sound as it might appear. The reason is that there is another equally good description of the situation, namely that 1% of the Cavian population should have their brain states decoded with d' such that $d'(x) = 1 - d(x)$. Under this description everyone experiences the same locked color terms, say $^{\bullet}\mathtt{c}(d)$, with the same qualia, $\mathtt{qual}_{\mathtt{Gray}}(^{\bullet}\mathtt{c}(d))$.[29]

Let me come at this from another angle, echoing some observations made in section 2.1. It is a commonplace that physical systems do not carry labels explaining what they represent; it is less well appreciated that they do not carry labels saying what representations they *are*. Fodor is rightly confident that there is no problem

[28]The Cavian story is mine, and not endorsed by any particular philosopher.

[29]Of course, adjustments to the alternative encoding must be made by modules downstream from the one under examination. If not, but the functional organization of the abnormal Cavians is the same as that of the majority, then one can simply apply the same idea everywhere else. That's more or less what functionalism comes down to; the two populations are alternative physical instantiations of functions describable in the same terms.

having a "language of thought" in the brain (Fodor, 1985), but if and when it is found, it will be an arrangement of physical-state types that can be decoded as symbol tokens arranged into sentences.

In (Block, 1978, p. 83), Ned Block suggests that "qualia may well not be in the domain of psychology," because psychology might be explainable functionally and qualia might not be. [¡Newell 93¿] I believe he would accept what I have to say about decodings, but put it in the "functionalism" column. This conclusion is reinforced by his later distinction (Block, 1995) between "access consciousness" and "phenomenal consciousness," respectively "A" and "P." A-consciousness is, roughly, what one can report on, or, more generally, base behavior on. P-consciousness is what one actually experiences. The P-type can outrun the A-type: one can sometimes experience more than one can report on, because, for instance, things are happening too fast. P-consciousness is supposed to be phylogenetically older than A. Primitive creatures experience much but do not make a distinction between what they experience and what there is, so they cannot be trained to distinguish between the way things seem and the way they are.

For reasons I have explained, I am skeptical about this heavy experience bias, but for now I want to point out that Block's notion of P-consciousness seems to be based on the idea that there is a level of consciousness that is "pre-decoding," as it were. That is, we are directly aware of activity in complex neuron systems (see Block 1995 for some proposals) unmediated by any representation scheme. Kirk's (1994) notion of "raw feeling" seems to be similar, as are Raffman's (1995) unreidentifiable qualia.

The claim that there is such a thing as P-consciousness separate from A-consciousness is difficult to reconcile with epiphenomenalism, because we couldn't know about P-c unless it had a causal effect on *something* in our brains. This causal effect by hypothesis runs in channels separate from those used by A-c. If these channels were post-decoding, as it were, then they would simply be the medium for another type of access consciousness, whether or not they were based on the same or a different decoding from "regulation" A-consciousness. Therefore they must constitute some

kind of direct awareness of brain activity as such. [[[Where does Churchland weigh in on this issue?]]]

The only proposal I know of about how this sort of direct awareness might work is that of Giulio Tononi (Tononi (2008)).[[[[30]]]] He proposes that brain activity is conscious to the extent that it has a high degree of *integrated information,* defined in terms of a (hard-to-compute) function of that activity. A system with n units gives rise to a *qualia space* with 2^n dimensions; a quale is a shape in this space.

It is difficult to know what to make of this theory. On one hand, because it is abstract, it must make use of an information-theoretic decoding of neural activity. On the other, Tononi claims that there is a unique decoding that maximizes Φ, the measure of integrated information (Tononi, 2008, p. 236), an idea that runs counter to the way decodings are to be understood. In an undesigned system such as the brain, there is no one right or wrong decoding. A decoding is interesting to the extent that its use produces nontrivial computational descriptions of a system's behavior. Decodings are analogous to frames of reference in physics; you have to have one, but the correctness of the results obtained does not depend on which one you use.[31]

If these arguments are correct, they suggest that my analysis of qualia in section 3 is at least coherent. Qualia are based on locked terms that are relative to a particular decoding of sensory inputs, but go beyond them in ways discussed in that section. If so, intersubjective spectrum inversion is logically impossible, because if there are symmetries in a sensory channel that allow permutation of qualia while preserving similarity relationships, then those symmetries entail the existence of just as many alternative decodings as there are supposed permutations of qualia.

[30] Cf. Edelman and Tononi (2000)

[31] The analogy isn't perfect. In physics you are supposed to get exactly the same results with respect to every frame of reference. Different decodings may yield different, albeit complementary, results.

5 Conclusion

The use of a formal language to capture the content of representations and metarepresentations sharply illuminates higher-order theories of consciousness. Although the properties of the *medium* (images, maps, or whatever) are suppressed by this maneuver, the medium is relatively unimportant in a higher-order theory.

The key idea of the logical reconstruction is that when one thinks about one's perceptions one must be able to correct them, which requires the ability to refer to them, which requires the logical language to have a *quoting feature* that makes terms into objects of reasoning. Correcting misperceptions is a matter of specifying the editing operations on these terms that transform them into terms that would have been correct in the original context where they were used. When quoting a term, the ability to "see into" that term is limited. Subterms that must be treated as atomic by introspection are said to be *locked*. Typically, a logical analysis of perception requires numbers (e.g., lengths, durations, brightness, loudness), but introspection blocks access to them. The terms containing them can be compared in various ways, but never broken down so that the numbers are visible to the introspective apparatus.

What emerges from this treatment of phenomenal consciousness is a new way to think about secondary qualities. A quality space is "secondary" if we do not move through it at a rate we are aware of or can control, and so cannot reason about transformations into different frames of reference, unlike space and time.[32] What is *not* necessary for a secondary quality is that it possess a characteristic *quale*. Qualia do not inhere in secondary properties as a matter of necessity, but only as an accident of our biological origin. Humans, or at least those humans steeped in European traditions and culture, seem to need to believe that there is a "felt" aspect to a secondary-quality space that accounts for our ability to judge similarities among different points in that space. *How* it accounts for this ability is left unstated, except

[32]Since the distinction between primary and secondary was first drawn, in the seventeenth century, physics has added many primary qualities, such as charge and charm, that we are not naturally aware of at all.

for an intuition that the the qualia involved are "intrinsic," "nonrelational," and so forth. At best this intuition runs in circles.

This logical analysis pays off by providing new ways of thinking about knotty problems in the philosophy of perception. It points to the key distinction between introspective and "normal" flows of information through a perceptual system. It blunts Jackson's "knowledge argument" (Jackson, 1982, 1986) by showing that what Mary learns is *vacuous*, part of a self-referential belief system we find it impossible to stay out of (McDermott, 2013). Knowing in advance that she will feel compelled to believe in the qualia of the new sensations she will have does not make the compulsion go away. Finally, the analysis sheds new light on the possibility of spectrum inversion by providing a new degree of freedom for preserving secondary qualities among candidate "inverts." You can't begin to analyze the representations in an agent without decoding physical structures *as* representations. To verify that someone is truly an invert, it is necessary to rule out the possibility that there is an alternative decoding that restores the representation to that of the uninverted population; or to refute the present approach by showing that in complex perceptual systems there is "direct awareness" of physical structures independent of any decoding.

There is much that remains to be done, including pursuing these ideas:

- On this account, qualia "exist," but only because we all believe they do (except when thinking about philosophy). What is the psychosemantics of such entities?

- It would be useful to generalize the Cavian example. It's easy to show that the result applies to any bijective transformation of the state space, but I believe it can be made to apply to a much broader class of populations.

A Formal Details of Syntax, Meta-syntax, and Semantics

The language we use is called *RCL*, for "Representation Content Language." It is a straightforward application of the ideas of Montague and ¡bibref¿ for intensional languages, augmented with a quotation device for referring to expressions within the language.

A.1 Abstract Syntax

I will be specifying the *abstract syntax* of *RCL*. This is a concept quite familiar to computer scientists but, judging from the literature, less so to philosophers. So let me sketch how it works.[33]

The *surface syntax* of a language connects expression types to sets of character strings. A *syntactically unambiguous* language is one in which a string is of at most one type. The connection can be established by characterizing each expression type using a phrase-structure rule and showing that no string can be produced in two different ways by application of rules. [¡¿] But phrase-structure rules don't just produce strings; they associate with each string a tree known as a "phrase-structure marker." So another way of saying that a language is unambiguous is to say that every string can be assigned at most one phrase-structure marker.

This being so, we could just make the phrase-structure trees *be* the language. Then syntactic ambiguity would be impossible. We can still express elements of the language as strings, using whatever convention we like to turn trees into strings; if the convention maps two trees to the same string, it's at most a presentational blemish. This is the idea of *abstract syntax.*[34] [¡McCarthy¿]

[33]Unfortunately for those trying to use Google to learn about abstract syntax, it is easy to confuse it for the much more common term *abstract syntax tree,* which is the data structure used by compilers to represent a program after it has been syntactically parsed (and usually analyzed in other ways). This sense of the term is descended from the formal sense used here, but with a considerably different emphasis.

[34]Of course, for there to be an algorithm to recover the tree structure from strings, the surface

Formally, a *grammar* $\mathcal{L}$ is a tuple $\langle C, B, R, S\rangle$, where

1. C is a possibly infinite set of *nonterminal symbols*;
2. B (disjoint from C) is a possibly infinite set of *terminal symbols*;
3. R is a possibly infinite set of *syntactic rules*, each of which is a tuple $\langle a, s_1, \ldots, s_n\rangle$, where $a \in C$, $n \geq 1$, and s_i is a class label or terminal symbol;
4. $S \in C$ is the *root symbol*.

All of the "possibly infinite" sets are recursively enumerable.[35] For readability, a rule $\langle a, s_1, \ldots, s_n\rangle$ is normally written a `::=` $s_1, \ldots s_n$.

A *syntax tree* is a finite labeled tree. An internal node of a syntax tree is *covered by rule* c `::=` $s_1, \ldots, s_n$ iff the node is labeled with c and has n children, the i'th child being labeled with s_i. A syntax tree is *generated* by $\mathcal{L}$ if (a) the root is labeled with S; (b) every internal node is covered by a rule $\in R$; and (c) every leaf node is labeled with an element of B.[36] The set consisting of all and only the trees covered by rules of $\mathcal{L}$ is the *(abstract) language* generated by $\mathcal{L}$.

For example, a simple abstract language of arithmetic expressions might be expressed as a tuple $\langle\{\texttt{exp, arith, num, var}\}, \mathbb{Z} \cup V \cup \{\texttt{+}, \texttt{-}, \times, \div\}, R, \texttt{exp}\rangle$, where $\mathbb{Z}$ is the integers and $V = \{\texttt{v}_i \mid i \in \mathbb{N}\}$. The rules R are the elements of the following list:

1. $R_{\texttt{exp1}} =$ `exp ::= num`

forms must not be ambiguous. But (a) it's usually not hard to remove ambiguity, using precedence rules and the like; and (b) it's a problem that may never arise, especially if we're using the language for theoretical purposes.

[35] We could do away with these infinities by making the rules and terms more complicated. For example, we could finitize a language with the natural numbers as terminal symbols by expressing n as $\texttt{s}(\texttt{s}(\ldots\texttt{s}(0)))$ (`s` iterated n times). One way to think about this is that we could move the recursive enumerability of the symbols and rules into the grammar. However, when we come to the language we care about (section A.3), such maneuvers would obscure the important syntax.

[36] Obviously, a node can be covered by at most one rule, because there is nothing to a rule but the labels to the left and right of the " `::=` " symbol.

2. $R_{\texttt{exp2}} = \texttt{exp ::= var}$

3. $R_{\texttt{exp3}} = \texttt{exp ::= arith}$

4. $R_{\texttt{arith}} = \texttt{arith ::= exp, op, exp}$

5. $R_{\texttt{op1}} = \texttt{op ::= +}$

6. $R_{\texttt{op2}} = \texttt{op ::= -}$

7. $R_{\texttt{op3}} = \texttt{op ::=} \times$

8. $R_{\texttt{op4}} = \texttt{op ::=} \div$

9. $R_{\texttt{num}} = \texttt{num ::=} i$, where $i \in \mathbb{Z}$

10. $R_{\texttt{var}} = \texttt{var ::=} \texttt{v}_i$, where $i \in \mathbb{N}$

Figure 7a shows a tree generated by this grammar (using `x` and `y` instead of $\texttt{v}_0$ and $\texttt{v}_1$). Another way to display this tree is given in figure 7b.

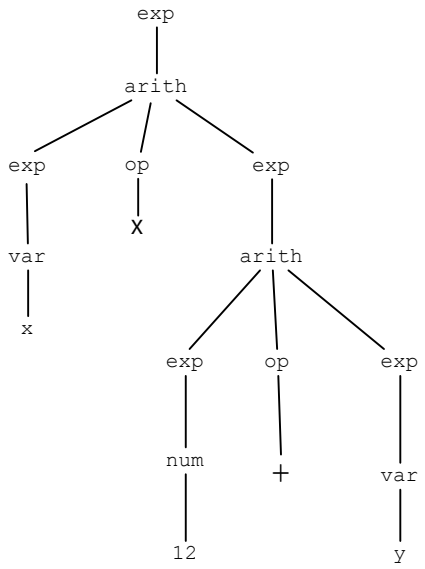
In text, I will use the "picture"

$$\begin{array}{c} \alpha \\ \| \\ \nu_1, \ldots, \nu_k \end{array}$$

to denote a syntax tree whose top node is labeled α, with k children $\nu_1, \ldots, \nu_k$.

A.2 Typed Abstract Grammars

When it comes to more sophisticated languages than the example of section A.1, mere syntactic correctness is not enough. The symbols must fit together in a way that makes sense and avoids paradoxes. For example, in $f(x, a)$, if x is a variable of type τ_1 and a is a constant of type τ_2, then f must be a function that takes two arguments, of types τ_1 and τ_2. If its result is of type τ, then the whole expression is of type τ and the type of f is $(\tau_1, \tau_2) \to \tau$. In general if $\tau_1, \ldots, \tau_n, \tau$ (with $n \geq 1$) is



$\langle$exp, $\langle$arith,$\langle$exp, $\langle$var,x$\rangle\rangle$,
$\langle$op, $\times\rangle$,
$\langle$exp, $\langle$arith,$\langle$exp, $\langle$num, 12$\rangle\rangle$,
$\langle$op,+$\rangle$,
$\langle$exp, $\langle$var,y$\rangle\rangle\rangle\rangle\rangle$

a. Graphic representation of tree b. Mathematical structure of tree

Figure 7: Syntax tree for $x \times (12 + y)$

a finite list of types and τ is a type, then $(\tau_1, \ldots, \tau_n) \rightarrow \tau$ is the type of functions that take n arguments of types τ_i and have values of type τ.

Using Church's (1940, Andrews 2014) approach to types, all syntax trees will be annotated with superscripts indicating their type. Terminal symbols come with these superscripts. Nonterminals get their types from superscript labels on the rules. For a syntax-tree node d covered by rule c^{τ} `::=` $s_1^{\tau_1}, \ldots, s_n^{\tau_n}$, if each child i is labeled with τ_i, then d may be labeled with τ. In other words, the type superscripts are used in syntactic rules as implicit *type-inference rules* each of which assigns a type superscript to a nonterminal given the type superscripts on its children. If the root of a syntax tree receives a type label using these rules, the tree is *typable*, otherwise *untypable*.

Formally, a *typed abstract grammar* is a tuple $\langle C, B, R, S, Y \rangle$. C, B, R, and S are as before, except that now the terminal symbols B are ordered pairs, $\langle sym, type \rangle$, always written sym^{type}. Y is a set of atomic types. Y and the symbols in B are disjoint, a property we emphasize by using upper case for the first letter of atomic-type symbols and lower case for the first letter of terminal symbols.

Every rule in R is now an ordered pair

$$\langle a \texttt{ ::= } s_1 \ldots s_n \,;\ \tau \leftarrow \tau_1, \ldots, \tau_n \rangle,$$

that is, an ordinary syntax rule plus a type-inference rule. Every nonterminal node of a syntax tree is labeled with an ordered pair $\langle c, \tau \rangle$, where c is a nonterminal and τ is its inferred type. The *language* generated by $\mathcal{L}$ is the set consisting of all and only the typable trees covered by rules of $\mathcal{L}$.

The grammar of section A.1 can be enhanced with types thus: There is one atomic type `Int`. Constants such as `19` are now written as $\texttt{19}^{\texttt{Int}}$. Variable $\texttt{v}_i$ becomes $\texttt{v}_i^{\texttt{Int}}$. `+` ,`-`, $\times$, and $\div$ each get superscript $(\texttt{Int}, \texttt{Int}) \rightarrow \texttt{Int}$. The rule $R_{\texttt{arith}}$ becomes

$$\texttt{arith}^{\tau} \texttt{ ::= } \texttt{exp}^{\tau_1}, \texttt{op}^{(\tau_1, \tau_2) \rightarrow \tau}, \texttt{exp}^{\tau_2}.$$

However, nothing very interesting happens in adding type labels to the trees in this case.[37] We are ready to move on to the language we actually care about.

[37] Unless you count the interpretation of $\div$ as having integer range as interesting.

A.3 Syntax of *RCL*

RCL is a family of languages, each characterized by its atomic-type symbols and constants. There are a finite number of atomic types and constants. No matter what else varies, there are always at least three atomic types, including `Boolean` and `Index` and at least one *nonlogical* type. One might want many nonlogical types, such as `String`, `Person`, and so forth, or just one, such as `Entity` (corresponding to Montague's e, (Montague, 1974)).[38] If τ is a type, then `Expr<`τ`>` is the type of expressions of type τ.[39]

A type is *index-protected* if `Index` occurs only on the left-hand side of $\rightarrow$. `Prop` is an abbreviation for `(Index)`$\rightarrow$`Boolean`. `AccRel` is an abbreviation for (`Index`, `Index`) $\rightarrow$ `Boolean`.[40]

The set of constants varies from dialect to dialect of *RCL*, but always includes $\texttt{false}^{\texttt{Boolean}}$ and $\supset^{(\texttt{Boolean},\texttt{Boolean})\rightarrow\texttt{Boolean}}$. More about these symbols and how they relate to other truth functions appears after the formal grammar.

The terminal symbols of *RCL* fall into three disjoint groups:

1. The constant symbols, each labeled with an index-protected type.

2. An infinite set of variable symbols $v_0, v_1, \ldots$. The same variable symbol v may be used with different types; v^{τ_1} and v^{τ_2} are considered to be the same variable if and only if $\tau_1 = \tau_2$. Variables may be labeled only with index-protected types. In examples, variables have arbitrary, sometimes mnemonic, names.

3. Odds and ends such as numbers and strings, whose properties we do not bother to formalize. Each is assumed to have an obvious atomic type.

To the right of some syntax rules in the following grammar is the annotation *Surface:* Σ, which indicates the surface syntax form normally used for expressions

[38] `Boolean` is the equivalent of t in Montague's notation. I write $(\tau_1) \rightarrow \tau_2$ for Montague's $\langle \tau_1, \tau_2 \rangle$.

[39] There is no equivalent in Montague for `Expr<`$\ldots$`>`; for the sources for this and other nonstandard notations in this paper, see section A.7.

[40] `Prop` is an abbreviation for "proposition"; `AccRel` for "accessibility relation." See below.

covered by that rule. The subscripts on repeated nonterminals on the right-hand sides of syntax rules are purely for matching their occurrences with occurrences of nonterminals in the surface form. I use extra parentheses in surface forms where they improve clarity.

The root symbol for the grammar is `term`. A *formula* is a tree labeled $\texttt{term}^{\texttt{Boolean}}$.

$\texttt{term}^{\tau}$ ::= *constant*$^{\tau}$ (one such rule for each *constant* of index-protected type τ)

$\texttt{term}^{\tau}$ ::= v_i^{τ} (One such rule for each $i \in \mathbb{N}$ and index-protected type τ.)

$\texttt{term}^{\tau}$ ::= $\texttt{app}_i^{\tau}$ $(i \in \mathbb{N}^+)$

$\texttt{term}^{\texttt{Boolean}}$ ::= $\texttt{equal}^{\texttt{Boolean}}$

$\texttt{term}^{\tau}$ ::= $\texttt{lambda}_i^{\tau}$ $(i \in \mathbb{N}^+)$

$\texttt{term}^{\texttt{Boolean}}$ ::= $\texttt{exist}_i^{\texttt{Boolean}}$ $(i \in \mathbb{N}^+)$

$\texttt{term}^{\texttt{Index}\to\tau}$ ::= $\texttt{intens}^{\texttt{Index}\to\tau}$

$\texttt{term}^{\tau}$ ::= $\texttt{extens}^{\tau}$

$\texttt{term}^{\texttt{Boolean}}$ ::= $\texttt{posmod}^{\texttt{Boolean}}$

$\texttt{term}^{\texttt{Expr<}\tau\texttt{>}}$ ::= $\texttt{quote}^{\texttt{Expr<}\tau\texttt{>}}$

$\texttt{term}^{\tau}$ ::= $\texttt{unquote}^{\tau}$

$\texttt{term}^{\texttt{Expr<}\tau\texttt{>}}$ ::= $\texttt{lock}^{\texttt{Expr<}\tau\texttt{>}}$

$\texttt{app}_n^{\rho}$ ::= $\texttt{term}_0^{(\tau_1,\ldots,\tau_n)\to\rho}, \texttt{term}_1^{\tau_1}, \ldots, \texttt{term}_n^{\tau_n}$ $(n \in \mathbb{N}^+)$

Surface: $\texttt{term}_0(\texttt{term}_1, \ldots, \texttt{term}_n)$

$\texttt{equal}^{\texttt{Boolean}}$ ::= $\texttt{term}_1^{\tau}, \texttt{term}_2^{\tau}$ (one such rule for each type τ)

Surface: $\texttt{term}_1 = \texttt{term}_2$

$\texttt{lambda}_n^{(\tau_1,\ldots,\tau_n)\to\tau}$ ::= $v_{i_1}^{\tau_1}, \ldots, v_{i_n}^{\tau_n}, \texttt{term}^{\tau}$ $(n \in \mathbb{N}^+, v_{i_j}$ pairwise distinct)

Surface: $\lambda(v_{i_1} : \tau_1, \ldots, v_{i_n} : \tau_n)\texttt{term}$

$\texttt{exist}_n^{\texttt{Boolean}}$ ::= $v_{i_1}^{\tau_1}, \ldots, v_{i_n}^{\tau_n}, \texttt{term}^{\texttt{Boolean}}$ $(n \in \mathbb{N}^+, v_{i_j}$ pairwise distinct)

Surface: $\exists(v_{i_1} : \tau_1, \ldots, v_{i_n} : \tau_n)\texttt{term}$

$\texttt{intens}^{\texttt{Index}\to\tau}$ ::= $\texttt{term}^{\tau}$ *Surface:* $\hat{}\texttt{term}$

$\texttt{extens}^{\tau}$ ::= $\texttt{term}^{\texttt{Index}\to\tau}$ *Surface:* $\check{}\texttt{term}$

$\texttt{posmod}^{\texttt{Boolean}}$ ::=

constant$^{(\tau_1,\ldots,\tau_n)\to(\texttt{Index},\texttt{Index})\to\texttt{Boolean}}, \texttt{term}_1^{\tau_1}, \ldots, \texttt{term}_n^{\tau_n}, \texttt{term}_{n+1}^{\texttt{Boolean}}$

(One rule for each *constant* of the given type and each $n \in \mathbb{N}$.)

Surface: <*constant*>$(\texttt{term}_1, \ldots, \texttt{term}_n, \texttt{term}_{n+1})$

$\texttt{quote}^{\texttt{Expr<}\tau\texttt{>}}$ `::=` $\texttt{term}^{\tau}$ *Surface:* `'term`

$\texttt{unquote}^{\tau}$ `::=` $\texttt{term}^{\texttt{Expr<}\tau\texttt{>}}$ *Surface:* $^{\sim}\texttt{term}$

$\texttt{lock}^{\texttt{Expr<}\tau\texttt{>}}$ `::=` $\texttt{term}^{\tau}$ *Surface:* $^{\bullet}\texttt{term}$

For semantic reasons, we impose the following technical restriction on the grammar of *RCL*. Define the *quote-nesting count* (QN count) of a syntax-tree node as follows: The count of the root node is 0. The count of a non-root node is the same as the count of its parent, unless the parent is labeled with `quote` or `unquote`. In the former case, the child's count is 1 greater than the parent's; in the latter, 1 smaller.

The technical restriction is to allow only syntax trees in which every node has a nonnegative QN count. Such a tree is said to be *QN-nonnegative*. This will not be important until section A.5.

Theorem 1 *No term is assigned a non-index-protected type by this grammar.*

Proof: By structural induction on the grammar. The only not-completely-trivial case is the rule for $\texttt{app}_i$; it will produce a term of non-index-protected type if ρ is non-index-protected. But then $\texttt{term}_0$'s type $(\tau_1, \ldots, \tau_n) \to \rho$ is non-index-protected, which is ruled out by the induction hypothesis. QED

In the next two sections I present a formal semantics for *RCL*. Here I explain informally what the various expressions mean.

$a = b$ means that a and b are the same entity.

The expression $\supset (p, q)$, normally written $p \supset q$, means "p only if q." All the other connectives can be defined in terms of $\supset$ and `false`:

- $\neg p =_{\text{df}} p \supset \texttt{false}$
- $\texttt{true} =_{\text{df}} \neg\texttt{false}$
- $p \vee q =_{\text{df}} (\neg p) \supset q$
- $p \wedge q =_{\text{df}} \neg(\neg p \vee \neg q)$

An expression

$$\lambda(u_1 : \tau_1, u_2 : \tau_2, \ldots, u_n : \tau_n)e^{\tau}$$

is the function, of type $(\tau_1, \ldots, \tau_n) \rightarrow \tau$, that has value e given arguments u_i. Note that the τ_i do not have to be atomic.

The quantifier $\exists(u_1 : \tau_1, u_2 : \tau_2, \ldots, u_n : \tau_n)(p)$ means that p is true for some assignment of values of the appropriate types to the variables u_i. $\forall(u_1 : \tau_1, u_2 : \tau_2, \ldots, u_n : \tau_n)(p)$ is defined as $\neg\exists(u_1 : \tau_1, u_2 : \tau_2, \ldots, u_n : \tau_n)(\neg p)$.

Because technically the types of variables are part of their names, it is possible to produce formulas such as

$$\exists(\texttt{s : Int, s : String})(\texttt{length}(\texttt{s}^{\texttt{String}}) = \texttt{s}^{\texttt{Int}})$$

If we avoid taking advantage of this "feature," we can drop type superscripts on variables in surface syntax.

The expression $\hat{}\alpha$ is of type $(\texttt{Index}) \rightarrow \tau$ when α is of type τ; it is the *intension* of α, which gives its reference with respect to every index ("possible world"). The inverse of "$\hat{}$" is "$\check{}$". The expression $\check{}\beta$ is of type τ if β is of type $\texttt{Index} \rightarrow \tau$. $\check{}\hat{}\alpha = \alpha$.

Modalities are symbols declared to be of type

$$(\tau_1, \ldots, \tau_n) \rightarrow \texttt{AccRel}$$

For example, suppose the language has the modality `P` (for "past"), of type `AccRel`. (Here $n = 0$, so the first set of arguments is absent.) The denotation of `P` is supposed to be a function R such that $R(w_1, w_2)$ is true just in case w_2 is earlier than w_1. So `<P>`(p) means that at some time in the past, p was true. The dual modality, `[P]`(p) means that at every time in the past, p was true. In general, `[`M`]`$(a_1, \ldots, a_n, p)$ is an abbreviation for $\neg$`<`M`>`$(a_1, \ldots, a_n, \neg p)$.

For the doxastic modality `Bel`[ieve], of type $(\texttt{Agent}) \rightarrow \texttt{AccRel}$, we want something like `[Bel](Alexis, flat(earth))` to be true with respect to an index w_1 if $\hat{}$`(flat(earth))` is true when evaluated with respect to any w_2 such that

$$\texttt{Bel}(\texttt{Alexis})(w_1, w_2)$$

is true, i.e., any w_2 compatible with what Alexis believes in w_1. We can be more precise about this after explaining the semantics.

A.4 Formal Semantics of *RCL*: Types

This and the next section describe how to assign meanings to terms and formulas in *RCL*, and, in particular, the way the world must look if it is described by a given collection of formulas. This is a quite different topic from discovering (or deciding) what a set of formulas (or equivalent representational structures in another medium) in an agent's head *actually* describe in the world. That topic goes by the label *psychosemantics* (Fodor, 1988); here we discuss *formal semantics*, the theory of possible interpretations of a set of terms.

An *interpretation* of a dialect of *RCL* is a tuple $\langle W, T, F\rangle$, where

- W is a set of *index points*;
- T is a function from the type symbols to nonempty sets of objects. $T(\texttt{Index}) = W$. $T(\texttt{Boolean}) = \{\mathbf{0}, \mathbf{1}\}$. $T(\tau_1) \cap T(\tau_2) \neq \emptyset$ only if $\tau_1 = \tau_2$.[41]
- F is a function whose domain is the constants of *RCL*, such that $F(b^\tau)$ is an object of type τ (τ may or may not be atomic). Further constraints will be added to F below.

For an interpretation to do useful work, we must extend T and F to functions Ty and D, which define the set correspond to every type and the object corresponding to every term, respectively. In this section we'll define $Ty_{(T,W,Tr)}(\tau)$, *the type set for* τ *with respect to* T, W, and Tr. $Ty_{(T,W,Tr)}()$ is a *type-definition function*, one that maps types to sets; if we define it right, it will be a *universal* type-definition function that maps *every* type defined in terms of T, W, and Tr to a set. W is the *set of indices*. Tr is the set of typable syntax trees according to some grammar. The only grammar we care about in this paper is that of *RCL* itself, but we don't

[41] For technical reasons, the range of T must be a set of individual objects with no set-theoretic structure. They are not sets, sequences, functions, graphs, or any other complex entity. This assumption is used in the proof of theorem 4.

have to pin down the grammar yet. However, in order to make "•" work, we assume that the grammar defining Tr does not include the nonterminal `nam`, which we use to build abstract syntax trees of the form

$$\begin{array}{c}\texttt{nam}\\ \|\\ x\end{array}$$

where x is an arbitrary object.[42] Such a tree is a "virtual name" for x; it will turn out to be of type `Expr<`τ`>` if x is of type τ. There is no surface form for this tree because it is not part of the language.[43]

In what follows, I use the notation $S \Rightarrow R$ to mean the set of all total functions from S to R. (It's a more graceful substitute for R^S.) So $(S_1 \times \cdots \times S_n) \Rightarrow R$, $n \geq 1$, is a set of ordered pairs whose first element is a tuple $\in (S_1 \times \cdots \times S_n)$ and whose second element is $\in R$; if $n = 1$, this will be taken to be equal to $S_1 \Rightarrow R$. The case where $n = 0$ does occasionally occur: $() \Rightarrow R$ is equal to R, and, in the metalanguage, $f()$ is a synonym for f.

We will explore Ty first, then, in the next section, D. $Ty_{(T,W,Tr)}$ is defined inductively as the union of a series of functions V_i. Defining the series is complicated by our desire to keep W separate from the universe of objects: in RCL, no expression can denote a possible world. For any type-definition function Y, let $I(Y) = Y \cup \{\langle \texttt{Index}, W\rangle\}$, i.e., Y extended with the mapping from `Index` to the set of possible worlds.

Okay, let's get the induction going by letting $V_0 = T$. Now define the function $E(F)$, where F is a type-definition function, thus:

$$E(F) = F \cup \{\langle \texttt{Expr<}\tau\texttt{>}, \{t^\tau \in Tr \mid \tau \in \operatorname{dom} F\} \cup \left\{ \begin{array}{l}\texttt{nam}\\ \quad\| \quad \mid x \in F(\tau)\\ \quad x\end{array} \right\} \rangle\}$$

[42] The tree notation was introduced at the end of section A.1.

[43] Indeed, it could not be, because x is an *object*, not a syntax tree.

$$\cup \{\langle (\tau_1, \ldots, \tau_n) \to \rho,\ I(F)(\tau_1) \times \cdots \times I(F)(\tau_n) \Rightarrow F(\rho)\rangle$$
$$| \, n \geq 1 \text{ and } \tau_i \in \operatorname{dom} F \cup \{\texttt{Index}\} \text{ and } \rho \in \operatorname{dom} F\}$$

E adds to a type-definition function F definitions for `Expr<`τ`>`, where τ is already defined by F, and $(\tau_1, \ldots, \tau_n) \to \rho$, where ρ is defined by F, and τ_i is either defined by F or is `Index`.

We can now continue and finish off the V_i series thus:

$$V_{i+1} = E(V_i)$$

and

$$V = \cup_{i=0}^{\infty} V_i$$

Theorem 2 *Define the depth of types recursively thus: The depth of a type symbol is 0. If τ has depth d then the depth of* `Expr<`τ`>` *is $d+1$. If the maximum depth of $\tau_1, \ldots, \tau_n$, and τ is d, then the depth of $(\tau_1, \ldots, \tau_n) \to \tau$ is $d+1$.*
All the types defined in V_0 are of depth 0. All the types defined in $V_{i+1} \setminus V_i$ are of depth $i+1$.

Proof: Obvious induction.

Theorem 3 *V is a least fixed point of E that includes T. That is: (a) $E(V) = V$ and (b) there is no V' such that $T \subseteq V' \subset V$ and $E(V') = V'$.*

Proof: By induction on the depth of types.

V is a fixed point for obvious reasons. To show it is the least fixed point, let V' be a smaller one: i.e., let $E(V') = V'$; let the restriction of V to $\operatorname{dom} V' = V'$; and let τ be a type of least depth such that $\tau \in (\operatorname{dom} V \setminus \operatorname{dom} V')$. τ has depth > 0 because V and V' both include T and $V_{n+1} \setminus V_n$ assigns extensions only to types of depth $n+1$. By hypothesis, the components of τ are $\in \operatorname{dom} V'$. So $\tau \in \operatorname{dom} E(V')$. But $E(V') = V'$, so $\tau \in \operatorname{dom} V'$, contradicting the assumption that $\tau \in (\operatorname{dom} V \setminus \operatorname{dom} V')$. QED

Theorem 4 *Every type is mapped by V to a nonempty set disjoint from the set corresponding to any other type.*

Proof: Every type is in the domain of V, because every type has a depth, and V_i picks up all the objects at depth i.

The proof of disjointness is by induction on the depth at which types are defined by the V_i sequence. It's clearly true for V_0, because types defined by T are disjoint. So assume it's true for all V_l with $0 \leq l < i$, and suppose at level i two types receive values with nonempty intersections. That is, there are two types τ and τ' and an object x such that $x \in V_i(\tau) \cap V_i(\tau')$. τ' is of depth j, with $0 < j \leq i$.[44]

There are three cases to deal with:

1. $\tau = \texttt{Expr<}\rho\texttt{>}$, $x^\rho \in Tr$ and $\rho \in \operatorname{dom} V_{i-1}$
2. $\tau = \texttt{Expr<}\rho\texttt{>}$, $x = \begin{array}{c} \texttt{nam} \\ \| \\ y \end{array}$, with $y \in V_{i-1}(\rho)$
3. $\tau = (\tau_1, \ldots, \tau_n) \rightarrow \rho$ and x is a function in $I(V_{i-1})(\tau_1) \times \cdots \times I(V_{i-1})(\tau_n) \Rightarrow V_{i-1}(\rho)$

I'll spell out the proof for the last case; the others are similar. In case 3, we have $\tau' = (\tau_1, \ldots, \tau_n) \rightarrow \rho$,

$$\begin{aligned} x &\in I(V_{i-1})(\tau_1) \times \cdots \times I(V_{i-1})(\tau_n) \Rightarrow V_{i-1}(\rho) \\ x &\in I(V_{j-1})(\tau'_1) \times \cdots \times I(V_{j-1})(\tau'_n) \Rightarrow V_{j-1}(\rho') \end{aligned}$$

Because $S \Rightarrow R$ is the set of *total* functions from S to R, it must be the case that $I(V_{j-1})(\tau'_k) = I(V_{i-1})(\tau_k)$ for $1 \leq k \leq n$ and $V_{j-1}(\rho') \cap V_{i-1}(\rho) \neq \emptyset$. All of the τ_i and τ'_i and both ρ and ρ' are of depth $< i$. By induction hypothesis, $\tau'_k = \tau_k$ and $\rho' = \rho$. Therefore $\tau = \tau'$. QED

[44] This is where we use the assumption that the range of $T = V_0$ consists of individual objects with no set-theoretic structure.

The universe U can be defined thus:

$$U = \cup\{S \mid \text{ for some index-protected type } \tau, V(\tau) = S\}$$

Finally, $Ty_{(T,W,Tr)} = I(V)$.

A.5 Formal Semantics of *RCL*: Expressions

Now we can define the central semantic function D, making use of Ty, defined in section A.4. It might be useful to review the definition of *interpretation* from the beginning of that section.

Let E be a *variable assignment*, i.e., a function from variables of the language to elements of U. A *type-legal variable assignment with respect to T and W* must map v^τ to an object of type $Ty_{(T,W,Tr)}(\tau)$. The notation $E^{v^\tau}_x$, where $x \in Ty_{(T,W,Tr)}(\tau)$, is the function that agrees with E everywhere except possibly for $E(v^\tau)$: $E^{v^\tau}_x(v^\tau) = x$. (I remind you of the formal requirement that variables be superscripted with their types.)

In this section I will use w to denote an index, or, more dramatically, a "possible world."

Now, loading D up with as many parameters as it will bear and more, I will define $D(E, w, \langle W, T, F\rangle)[\alpha]$, which assigns a denotation to α with respect to type-legal variable assignment E, index $w \in W$, and interpretation $\langle W, T, F\rangle$. Except for the handling of $'e$ and $^\bullet e$, the definition of D will be in terms of surface syntax. Abbreviating $\langle W, T, F\rangle$ as $\mathcal{M}$, and suppressing type superscripts when necessary to avoid crippling near-sightedness, D is defined by:

1. If v^τ is a variable of type τ, $D(E, w, \mathcal{M})[v^\tau] = E(v^\tau)$.

2. If c is a constant or function symbol, $D(E, w, \mathcal{M})[c] = F(c)$. These two built-in constants have fixed meaning:

   - $D(E, w, \mathcal{M})[\texttt{false}] = \mathbf{0}$
   - $D(E, w, \mathcal{M})[\supset] = \{\langle\langle\mathbf{0}, \mathbf{0}\rangle, \mathbf{1}\rangle, \langle\langle\mathbf{0}, \mathbf{1}\rangle, \mathbf{1}\rangle, \langle\langle\mathbf{1}, \mathbf{0}\rangle, \mathbf{0}\rangle\langle\langle\mathbf{1}, \mathbf{1}\rangle, \mathbf{1}\rangle\}$

3. $D(E,w,\mathcal{M})[\lambda(u_1:\tau_1,\ldots,u_n:\tau_n)(e^\tau)]$ is the set of ordered pairs

$$\{\langle x,r\rangle \mid x=\langle x_1,\ldots,x_n\rangle \in Ty_{(T,W,Tr)}(\tau_1)\times\cdots\times Ty_{(T,W,Tr)}(\tau_n)$$
$$\text{and } r=D(E^{u_1u_2\ldots u_n}_{x_1x_2\ldots x_n},w,\mathcal{M})[e]\}.$$

4. $D(E,w,\mathcal{M})[e_0(e_1,\ldots,e_n)]=r$, where

$$\langle\langle D(E,w,\mathcal{M})[e_1],\ldots,D(E,w,\mathcal{M})[e_n]\rangle,r\rangle \in D(E,w,\mathcal{M})[e_0]$$

5. $D(E,w,\mathcal{M})[x=y]=\mathbf{1}$ if $D(E,w,\mathcal{M})[x]=D(E,w,\mathcal{M})[y]$, else $\mathbf{0}$

6. $D(E,w,\mathcal{M})[\exists(u_1:\tau_1,u_2:\tau_2,\ldots,u_n:\tau_n)(e)]=\mathbf{1}$ if there are $x_1,x_2,\ldots,x_n$ ($x_i\in Ty_{(T,W,Tr)}(\tau_i)$) such that $D(E^{u_1}_{x_1}\cdots{}^{u_n}_{x_n},\mathcal{M})[e]=\mathbf{1}$; otherwise, $\mathbf{0}$.

7. $D(E,w,\mathcal{M})[\hat{}e^\tau]=$ the function $\in$ `Index` $\Rightarrow Ty_{(T,W,Tr)}(()\tau)$ whose value at w' is $D(E,w',\mathcal{M})[e]$.

8. $D(E,w,\mathcal{M})[\check{}e^{\texttt{Index}\to\tau}]=f(w)$, where $f=D(E,w,\mathcal{M})[e]$. Note that $f(w)\in Ty_{(T,W,Tr)}(\tau)$.

9. $D(E,w,\mathcal{M})[\texttt{<}M^{(\tau_1,\ldots,\tau_n)\to\texttt{AccRel}}\texttt{>}(e_1^{\tau_1},\ldots,e_n^{\tau_n},p^{\texttt{Boolean}})]=\mathbf{1}$ if and only if $D(E,w',\mathcal{M})[p]=\mathbf{1}$ for some w' such that $r(w,w')=\mathbf{1}$, where, letting $x_i=D(E,w,\mathcal{M})[\texttt{e}_i]$, $r=D(E,w,\mathcal{M})[M](x_1,\ldots,x_n)$

The only expressions left out of this account are of those of the form $'e$ and $^\bullet e$. These are explained best in terms of abstract syntax.

$$D(E,w,\mathcal{M})\begin{bmatrix}\texttt{lock}^{\texttt{Expr<}\tau\texttt{>}}\\ \|\\ e^\tau\end{bmatrix}=\begin{matrix}\texttt{nam}\\ \|\\ x\end{matrix}\quad\text{if } D(E,w,\mathcal{M})[e^\tau]=x$$

$$D(E,w,\mathcal{M})\begin{bmatrix}\texttt{quote}^{\texttt{Expr<}\tau\texttt{>}}\\ \|\\ e\end{bmatrix}=QQ^1[e]$$

The function QQ^i refers to the subscripts E, w, and $\mathcal{M}$, but I will suppress them in what follows. (i is always taken to be ≥ 1.) All the important stuff is in the last few items of the following list:

- $QQ^i \left[\begin{array}{c} \texttt{term} \\ \| \\ T \end{array}\right] = \begin{array}{c} \texttt{term} \\ \| \\ QQ^i[T] \end{array}$
- $QQ^i[c] = c$, if c is a constant symbol
- $QQ^i[v^\tau] = v^\tau$, if v is a variable and τ is its type.
- $QQ^i \left[\begin{array}{c} \texttt{app}_n \\ \| \\ t_0, \ldots, t_n \end{array}\right] = \begin{array}{c} \texttt{app}_n \\ \| \\ QQ^i[t_0], \ldots, QQ^i[t_n] \end{array}$
- $QQ^i \left[\begin{array}{c} \texttt{equal} \\ \| \\ t_1, t_2 \end{array}\right] = \begin{array}{c} \texttt{equal} \\ \| \\ QQ^i[t_1], QQ^i[t_2] \end{array}$
- If $\alpha = \texttt{exist}$ or $\texttt{lambda}$, $QQ^i \left[\begin{array}{c} \alpha \\ \| \\ u_1, \ldots, u_n, t \end{array}\right] = \begin{array}{c} \alpha \\ \| \\ u_1, \ldots, u_n, QQ^i[t] \end{array}$
- If $\alpha = \texttt{intens}$, $\texttt{extens}$, or $\texttt{lock}$, $QQ^i \left[\begin{array}{c} \alpha \\ \| \\ t \end{array}\right] = \begin{array}{c} \alpha \\ \| \\ QQ^i[t] \end{array}$
- $QQ^i \left[\begin{array}{c} \texttt{posmod} \\ \| \\ c, t_1, \ldots, t_n, t_{n+1} \end{array}\right] = \begin{array}{c} \texttt{posmod} \\ \| \\ c, QQ^i[t_1], \ldots, QQ^i[t_{n+1}] \end{array}$
- $QQ^i \left[\begin{array}{c} \texttt{quote} \\ \| \\ t \end{array}\right] = \begin{array}{c} \texttt{quote} \\ \| \\ QQ^{i+1}[t] \end{array}$

- $QQ^{i+1}\left[\begin{array}{c}\texttt{unquote}\\ \|\\ t\end{array}\right] = \begin{array}{c}\texttt{unquote}\\ \|\\ QQ^{i}[t]\end{array}$
- $QQ^{1}\left[\begin{array}{c}\texttt{unquote}\\ \|\\ t\end{array}\right] = D[t]$ (Here I've omitted the arguments $(E, w, \mathcal{M})$ of D.)

Basically, $QQ^1[T]$ = the tree T itself, with subtrees marked as $^\sim e$ replaced by $D[e]$. However, following (Taha and Sheard, 2000), I have generalized this idea to handle occurrences of `quote` nested inside other `quotes`. We don't particularly need this capability, but it is just as easy to include it as to forbid it. The superscript on QQ says how many levels deep in `quotes` we are.

This definition assigns no independent meaning to $^\sim e$ outside of a quoted expression, which is okay, because the restriction on QN count (see section A.3) means there aren't any. Inside a `quote`, the `unquote` gets replaced by the expression denoted by e, if it occurs inside just one level of quotation.

Lemma 5 *When the definition of D is applied to a legal (QN-nonnegative) syntax tree, the D-QQ recursion is such that (a) QQ^i ($i \geq 1$) is applied only to nodes with QN count i and D is applied only to nodes with QN count 0; and (b) the value of $D(E, w, \mathcal{M})[^\sim e]$ is never required for any e.*

Proof: By induction on distance of a syntax-tree node from the root of the syntax tree (its *depth*), and the definition of QN count. The root cannot be an `unquote` node because it is always labeled `term`. Now assume the hypothesis is true for an arbitrary node N at depth d, and consider the child N' of such a node. We need consider only nodes labeled `quote` or `unquote`. In the former case, if N has QN count 0 then by induction hypothesis we require the value of $D(\ldots)[N]$, which is $QQ^1[N']$. If N has QN count $q \geq 1$ then by hypothesis we require $QQ^q[N]$ and therefore we apply QQ^{q+1} to N'. If N is labeled with `unquote`, then by induction hypothesis we require $QQ^q[N]$ for $q \geq 1$. If $q = 1$ then we'll need $D(\ldots)[N']$, else $QQ^{q-1}[N']$. QED

Using this lemma, we can prove

Theorem 6 *If e^τ is a syntactically legal RCL expression of type τ, $D(E, W, \mathcal{M})[e] \in Ty_{(T,W,Tr)}(\tau)$.*

Proof: This one will require induction on the *height* of a syntax-tree node, defined in the obvious way: The height of a leaf is 0, and the height of an interior node is 1+ the maximum height of its children. We first strengthen the theorem to the simultaneous statement that

$$D(E, w, \mathcal{M})[e^\tau] \in Ty_{(T,W,Tr)}(\tau)$$

$$\text{and } QQ^i(E, w, \mathcal{M})[e^\tau] \in \begin{cases} Ty_{(T,W,Tr)}(e') & \text{if } i = 1 \text{ and } e = \begin{bmatrix} \texttt{unquote} \\ \| \\ e' \end{bmatrix} \\ Tr^\tau & \text{otherwise} \end{cases}$$

The base case is for nodes of height 0, i.e., leaves. For the D side, the theorem is true because of the way E and F are defined. (I remind you that $\mathcal{M} = \langle W, T, F \rangle$.) For the QQ^i side, it is true because constant and variable nodes are labeled with their types.

Now assume the theorem is true for all nodes of height $< n$, and consider a node N of height n. For the D side, the most important case is the label `quote`, whence the theorem follows from the induction hypothesis for QQ^1.

For other sorts of syntax tree, the proofs are straightforward. I'll just do the `app` case; the others are similar. An `app` node with surface form $e_0(e_1, \ldots, e_k)$ has $k + 1$ children. The type of e_0 must be $\tau_0 = (\tau_1, \ldots, \tau_k) \to \rho$, where τ_i is the type of e_i, $1 \le i \le k$. By induction hypothesis, $D[e_0] \in Ty[e_0]$, a set of pairs representing a total function whose domain is $Ty[\tau_1] \times \cdots \times Ty[\tau_k]$ and whose range is $Ty[\rho]$; and $D[e_i] \in Ty[\tau_i]$. Hence there is a pair in the set whose first element is $\langle D[e_1], \ldots, D[e, k] \rangle$, and $D[N] =$ the second element of that pair, which is $\in Ty[\rho]$.

For the QQ^i side, if $i = 1$ and N is labeled `unquote`, the theorem follows from the induction hypothesis for D. All other combinations of `quote/unquote` follow from lemma 5 and the induction hypothesis for $QQ^{i\pm1}$. The other labels are trivial. QED

Finally, let's revisit modalities. Earlier I introduced the `Bel` function, of type (`Agent`) → `AccRel`. `[Bel]`(a,p) is supposed to mean "a believes that p." We can now specify the semantics of `Bel` more fully:

$$D(E,w,\mathcal{M})[\texttt{[Bel]}(a,p)] = \mathbf{1} \text{ iff } D(E,w',\mathcal{M})[p] = \mathbf{1}$$

for all w' such that $F(\texttt{Bel})(x)(w,w') = \mathbf{1}$, where $x = D(E,w,\mathcal{M})[a]$. That is, $\texttt{[Bel]}(a,p)$ is true w.r.t. w if and only if p is true in every possible world w' doxastically accessible from w as far as agent x knows.

The point of the "ˇ^" notation is to lift expressions into their values as functions of indices, so that `Bel` speaks in terms of these functions; but the ˇ operator cashes the ^ out by applying the lifted expression to the world reached after jumping to some other world (Dowty et al., 1981). These lifts and jumps are implicit in the `[]`,`<>` notation for modalities.

A.6 Path Designators, Lists, and Sets

In the body of the paper, we introduce the notation $\texttt{[}i_1\texttt{,...,}i_n\texttt{]}$ to denote a path through an abstract-syntax tree, namely, the i_n'th subtree of the $\ldots i_1$'st subtree of the given tree. We don't need to treat these expressions as part of the syntax or semantics of the *RCL* logical system, but simply as syntactic sugar for plain-vanilla terms. The technique is standard in programming languages. [¡¿]

We introduce a type `List` and a binary function "`::`", of type `(Object, List)` → `List`, always written between its two arguments (like "`+`"). The empty list is a constant with name `[]`. The expression $x_1\texttt{::(}x_2\texttt{::(...}x_n\texttt{::[]))}$ is the list whose jth element is x_j. It may be abbreviated $\texttt{[}x_1\texttt{,...,}x_n\texttt{]}$.

The only fly in the ointment is that we don't actually have a type `Object` that every type matches. Instead, we must introduce a family of types $\texttt{List}_\tau$, meaning "list of elements of type τ." The operator `::` must actually be a *family* of functions $\texttt{::}_\tau$; and there must a separate empty list $\texttt{[]}_\tau$ for each type τ. We have neglected this technicality, not just for path designators, but for other uses of lists, especially in appendix B.

Sets are like lists except, of course, for being unordered. Indeed, we can treat the notation $\{x, y, z\}$ as syntactic sugar for `set(`$[x, y, z]$`)`, where the `set` function produces a set given a list of its elements. Of course, a complete treatment of set theory would require many ramifications, but we'll use the notation only for the purpose of listing small unordered collections.

A.7 Sources and explanations for notations

In this section I briefly explain where the notations for nonstandard types and expressions came from. Most of them originated with functional-programming-language theorists.

The notation `Expr<`τ`>` for the type of expressions of type τ is from the F# programming language [¡¿]. The angle brackets here are from a more general use of such brackets for "parameterized types" (also called "generic types") (Thatcher et al., 1978). Generic types would have been useful in this paper for speaking about lists; lists of elements of type τ would be `List<`τ`>`. Similarly, we could have had a type `Quale<Color>` instead of $\texttt{Quale}_{\texttt{Color}}$. However, using a generalized facility of this kind might have required complexity in the analytic apparatus that would have distracted from the main point (Reynolds, 1984). The *RCL* language is basically a straightforward Montague-style logic except for the ability to quasi-quote *RCL* expressions in *RCL* (Quine, 1940).

The notation for quasi-quotation is borrowed from the Lisp programming language (McCarthy, 1960, Graham, 1996) and from Taha and Sheard's (2000) MetaML. In the surface syntax I use the prefix character "’" to indicate a quasi-quoted expression, and the prefix character "~" to indicate a subexpression within which quoting is suspended. In the abstract syntax, these are trees labeled with `quote` and `unquote` respectively. The concept of "locked terms" (section 3.2), and the notation for them ("•" and `lock`), are my own invention.

[[[Combining worlds and times into a single `Index` type is based on (Moore, 1985).]]]

[[[With a more powerful type system, we might use *existential types* [¡¿] to

conceal the structure of `color` terms instead of resorting to the •(...) notation.]]]

B Details of subpersonal visual representation

Among the many wonderful things the human vision system does is separate out judgments of illumination from reflectance. *Illumination* is the light falling on a surface; *reflectance* is the percentage of light falling on the surface that is re-emitted.[45] The pattern of energy directly received by the eyes is called *luminance*, the product of illumination and reflectance. Refactoring luminance into its two components is almost impossible, but the human vision system does a surprisingly good job. I am not going to worry about how illumination is represented, and in any case the brain doesn't seem as interested in that as in reflectance, which can reveal important information about objects at a distance.[46]

As sketched in section 2.4, the vision system must extract knowledge about three-dimensional objects starting with information about their surfaces, which they analyze as surface patches.[47] Although a patch has a location in the visual field, it's not just a blob; it has *boundaries*. A boundary is a one-dimensional structure consisting of a loop of *boundary segments*.[48] What defines a segment is the pair of patches on either side of it. A transition between segments occurs when the pair

[45] In reality, this is not a simple scalar quantity. It varies with the viewing angle and angle of incidence of the light. But in a philosophy paper ruthless oversimplification of such matters is the order of the day.

[46] Although shading due to an object's surface curvature can be made to yield information about its shape [¡¿].

[47] All kinds of qualifications: Some objects may not be well separated from other objects. Some patches represent transparent surfaces, so there can be more than one patch covering a given point. The visual field changes slightly with slight movements of the head, revealing depth from occlusion and parallax. This is probably a crucial part of the phenomenology, because, while holding still is something hunters, including our ancestors, actually do and did, absolute stationarity is not required.

[48] I ignore the problem of objects that extend beyond the borders of an image. Images formed by biological eyes obviously do not have tidy rectangular borders the way computer-vision images do. Given the way the visual world is constructed out of multiple foveations from more than one vantage point, the whole issue of borders must be dealt with differently in the biological system.

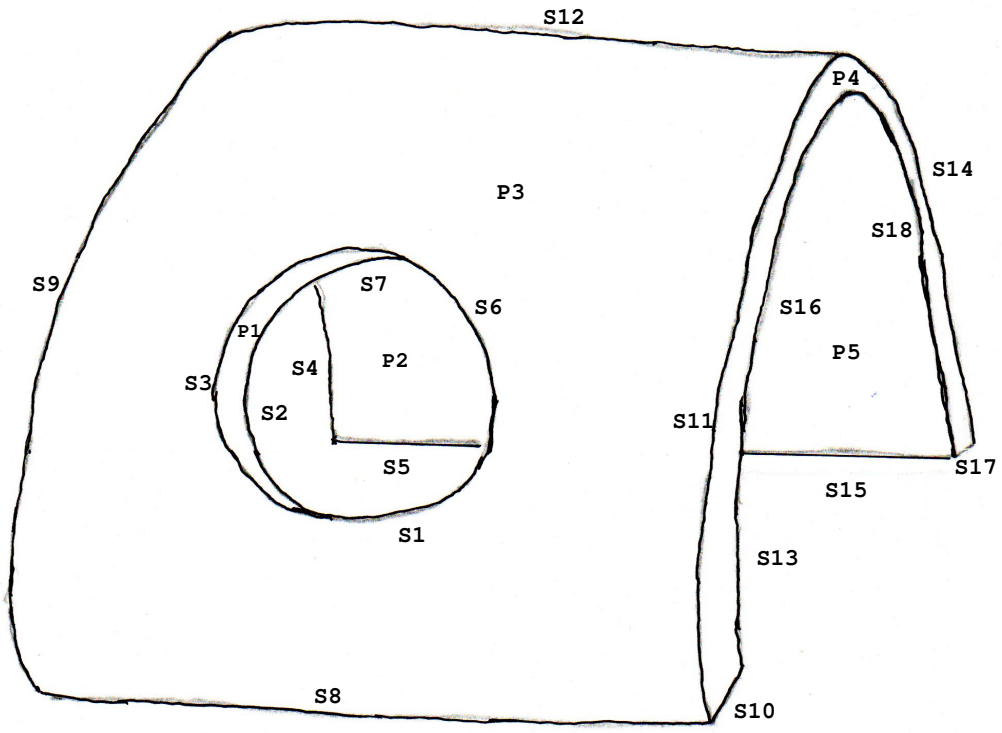


Figure 8: Object of figure 2 with patches and boundary segments labeled

changes.

Figure 8 shows the boundaries and segments a vision system might extract from an image or sequence of images of the arch in figure 2. There are two major families of patch-boundary segments: those in which the two patches on either side are from surfaces that meet at the physical edge producing the boundary; and those in which the two patches are from surfaces at different depths, or in which one side is from an object and the other side is "background" (the sky, the earth, or faraway stuff). The second type is called an *occluding* segment; the first is *non-occluding*. For either sort, the two surfaces on either side of the segment may or may not belong to the same object, although exactly what constitutes an object is context-dependent. The non-occluding case is divided into two subcases, those in which the boundary between patches represents a discontinuity in surface orientation or curvature, and

those in which what changes at the boundary is the reflectance of the surface.[49] In Figure 8, the only non-occluding boundary segments are at surface discontinuities (e.g., `S11` or `S10`). In a smoother surface, say, one with a bowl-shaped depression, the surface would be segmented by the change in curvature around the lip of the bowl. A discontinuity in reflectance occurs when a surface is marked, either naturally or artificially; there are none of these in figure 8.

The patches discovered in analyzing a scene involving an object such as that of figure 2 are physical structures in three-dimensional space, but still tied to the image. For example, patches `P2` and `P5` in figure 8 are part of the same physical surface, much of which cannot be seen because it is occluded by other parts of the arch. (Half of it is on the other side of the chunk whose visible face is patch `P3`.) We won't pursue the further step of putting such surface fragments together to infer volumes.

The predicates I use in describing the content of patch representations are:

- `patch_bounds(`p`, `s`)`: s is the set of all the boundaries of patch p. The next predicate is used to describe them.

- `patch_boundary(`b, l`)`: l is a list of the segments of patch boundary b in counterclockwise order.[50]

- `boundary_seg(`s`, `b`, `p`, `q`, `d`, `c`)`: s is a segment of boundary b of patch p. (Specifically, if s is considered directed with the same orientation as the cycle of b, then p is on its left.) On the other side of s is q, which is either another patch (of the same or different surface) or `nopatch` if it's the background of the image. d is the specification of whether b is `occluding`, meaning q is further away, `occluded`, meaning q is closer, or `nonocc`, meaning that b is a

[49] What about boundaries represented by shadows? In my hypothetical visual system, shadow boundaries do not correspond to patch boundaries. They are treated as illumination differences; I'm going to neglect them in what follows

[50] Technically, the boundary is formed by the cycle denoted by l, which is identical to the cycle denoted by a rotation of l, i.e., a list obtained by moving the first element from the front of the list to the rear.

non-occluding segment. Finally, c is a specification of what kind of curve b represents in the physical surface of which b is a patch, one of `reflectance`, `surf_orient`, or `nocurve` if b is not a reflectance edge or surface-orientation change, but just an accident of viewing direction as far as p is concerned. If d =`occluded` then b always =`nocurve`.

In the list above, the predicates describe the qualitative structure of object surface patches. At some point things have to become more numerical, either one-dimensional or two-dimensional maps. The following predicates involve arguments that are "map-like."

- `patch_surf_orientation(`p`,` m`)`: m is a two-dimensional map of the surface orientation over patch p.
- `patch_reflectance_map(`p`,` m`)`: m is a two-dimensional map of the reflectance over patch p.
- `curve_description(`s`,` p`,` d`)`: d is a three-dimensional parameterization of boundary segment s, in the frame of reference of surface patch p, in which s lies. (We'll leave unspecified how frames of this sort are represented, but the next two predicates can be used to capture relevant information.)
- `patch_depth(`p`,` d`)`: d is a "qualitative" description of the relative depth of patch d. *Relative depth* is an imprecise estimate of how far away the patch is, but precise enough to order it with respect to other patches.
- `patch_direction(`p`,` h`,` z`)`: h and z specify the direction to (the centroid of) patch p from the eye, as angles in polar coordinates.

The map-like arguments describe quantities that change smoothly. Whether we view these as arrays of numbers or parameterizations of some kind is another unsettled issue. In the body of the paper, we simplify `patch_reflectance_map` drastically, replacing it with `patch_color`.[51] My propositional reconstruction of figure 8

[51] Another obvious omission is analysis of surface texture, which is just as important as reflectance, and perhaps subsumes it.

```
patch_bounds(P1, {bound_P1})
patch_bounds(P2, {bound_P2})
patch_bounds(P3, {bound_P3_1, bound_P3_2})
patch_bounds(P4, {bound_P4})
patch_bounds(P5, {bound_P5})

patch_boundary(bound_P1, [S2, S7, S3])
patch_boundary(bound_P2, [S5, S6, S7, S4])
patch_boundary(bound_P3_1, [S8, S11, S12, S9])
patch_boundary(bound_P3_2, [S1, S3, S6])
patch_boundary(bound_P4, [S10, S13, S16, S18, S17, S14, S11])
patch_boundary(bound_P5, [S15, S18, S16])
```

Table 1: Propositional reconstruction of qualitative structure of figure 8: Patch boundaries

boundary_seg(S2, bound_P1, P1, P$^?$, occluding, surf_orient)

boundary_seg(S7, bound_P1, P1, P2, occluding, surf_orient)

boundary_seg(S3, bound_P1, P1, P3, nonocc, surf_orient)

boundary_seg(S5, bound_P2, P2, P$^?$, nonocc, surf_orient)

boundary_seg(S6, bound_P2, P2, P3, occluded, nocurve)

boundary_seg(S7, bound_P2, P2, P1, occluded, nocurve)

boundary_seg(S4, bound_P2, P2, P$^?$, occluding, surf_orient)

boundary_seg(S8, bound_P3_1, P3, P$^?$, nonocc, surf_orient)

boundary_seg(S11, bound_P3_1, P3, P4, nonocc, surf_orient)

boundary_seg(S12, bound_P3_1, P3, P$^?$, occluding, nocurve)

boundary_seg(S9, bound_P3_1, P3, P$^?$, occluding, surf_orient)

boundary_seg(S1, bound_P3_2, P3, P$^?$, occluding, surf_orient)

boundary_seg(S3, bound_P3_2, P3, P1, nonocc, surf_orient)

boundary_seg(S6, bound_P3_2, P3, P2, occluding, surf_orient)

boundary_seg(S10, bound_P4, P4, P$^?$, nonocc, surf_orient)

boundary_seg(S13, bound_P4, P4, P$^?$, occluding, surf_orient)

boundary_seg(S16, bound_P4, P4, P5, occluding, surf_orient)

boundary_seg(S18, bound_P4, P4, P5, nonocc, surf_orient)

boundary_seg(S17, bound_P4, P4, P$^?$, nonocc, surf_orient)

boundary_seg(S14, bound_P4, P4, P$^?$, occluding, surf_orient)

boundary_seg(S11, bound_P4, P4, P3, nonocc, surf_orient)

boundary_seg(S15, bound_P5, P5, P$^?$, nonocc, surf_orient)

boundary_seg(S18, bound_P5, P5, P4, nonocc, surf_orient)

boundary_seg(S16, bound_P5, P5, P4, occluded, nocurve)

Table 2: Propositional reconstruction of qualitative structure of figure 8: Boundary segments

is shown in tables 1 and 2. In statements of the form `boundary_seg(`s`,` b`,` p`,` q`,` d`,` c`)` (table 2), if q does not belong the object being analyzed, I have just put "`P`$^?$".

The number of mistakes one could make in estimating the structure and parameterization of a surface are of course innumerable. A vision system, biological or artificial, can miss an important boundary, thus merging two patches into one. It can misclassify a boundary, and in particular it can become confused between `occluded` and `occluding` (a figure-ground error). Illumination can be factored out from luminance wrongly, yielding incorrect estimates of reflectance (Gilchrist, 2006).

All of these errors can, I believe, be described using edit operations on the the data structures (formulas) used to describe patches. The only exception might be flat-out hallucination, which can be handled, if it ever is, by rejecting a large class of perceptual `judgments` altogether. But, although I don't think hallucination presents any special problems, it is beyond the scope of this paper.

References

Kathleen Akins. Of sensory systems and the "aboutness" of mental states. *J. of Phil*, 93(7):337–372, 1996.

Torin Alter and Sven Walter. *Phenomenal Concepts and Phenomenal Knowledge:New Essays on Consciousness and Physicalism*. Oxford University Press, 2006.

Peter B. Andrews. Church's type theory. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Online at , 2014. URL `http://plato.stanford.edu/archives/spr2014/entries/type-theory-church/`. (Spring 2014 Edition).

David M. Armstrong. *A Materialist Theory of the Mind*. Routledge & Kegan Paul, London, 1968.

Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter

Patel-Schneider. *The Description Logic Handbook; Theory, Implementation, and Applications*. Cambridge University Press, 2003.

Ned Block. Troubles with functionalism. In C. Wade Savage, editor, *Perception and Cognition: Issues in the Foundation of Psychology, Minn. Studies in the Phil. of Sci*, pages 261–325. University of Minnesota Press, 1978.

Ned Block. On a confusion about a function of consciousness. *Behavioral and Brain Sciences*, 18(2):227–247, 1995.

Ned Block, Owen Flanagan, and Güven Güzeldere, editors. *The Nature of Consciousness: Philosophical Debates*. MIT Press, Cambridge, Mass., 1997.

Harold I. Brown. The case for indirect realism. In Edmond Wright, editor, *The Case for Qualia*, pages 45–58. MIT Press, Cambridge, MA, 2008. (A Bradford Book.).

Alex Byrne and David R. Hilbert. Color realism and color science, 2003. *Behavioral and Brain Sciences* **26**.

Peter Carruthers. *Consciousness: Essays from a Higher-Order Perspective*. Clarendon Press, Oxford, 2005.

Peter Carruthers. Natural theories of consciousness. *European J. of Phil*, 6(2): 203–222, 1998.

David Chalmers. On implementing a computation. *Minds and Machines*, 4(4): 391–402, 1994.

David J. Chalmers. A computational foundation for the study of cognition. *J. of Cognitive Science*, 12(4):323–357, 2011. URL `http://consc.net/papers/computation.html`.

W. G. Chase, editor. *Visual Information Processing*. Academic Press, New York, 1973.

W. G. Chase and Herbert A. Simon. Perception in chess. *Cognitive Psych*, 4:55–81, 1973a.

W. G. Chase and Herbert A. Simon. The mind's eye in chess. In Chase (1973).

Alonzo Church. A formulation of the simple theory of types. *J. of Symbolic Logic*, 5:56, 1940. 68.

Paul M. Churchland and Patricia S. Churchland. Recent work on consciousness: philosophical, theoretical, and empirical. *Seminars in Neurology*, 17(2):179–186, 1997.

M.J. Cresswell and G.E. Hughes. *A New Introduction to Modal Logic*. Routledge, London, 1996.

Robert Cummins. *Meaning and Mental Representation*. The MIT Press, Cambridge, Mass., 1989.

Antonio R. Damasio. *The Feeling of What Happens: Body and Emotion in the Making of Consciousness*. Harcourt, Inc, San Diego, 1999. (A Harvest Book.).

Daniel C. Dennett. What RoboMary knows. In Alter and Walter (2006), pages 15–31.

Daniel C. Dennett. *Brainstorms*. Bradford Books/MIT Press, Cambridge, Mass., 1978a.

Daniel C. Dennett. A cure for the common code? In *Brainstorms* Dennett (1978a), pages 90–108.

Daniel C. Dennett. *Consciousness Explained*. Little, Brown and Company, Boston, 1991.

David R. Dowty, Robert E. Wall, and Stanley Peters. *Introduction to Montague Semantics*. Dr. Reidel Publishing Company, Dordrecht, 1981.

Gerald Edelman and Giulio Tononi. *Consciousness: How Matter Becomes Imagination*. The Penguin Press, London, 2000.

Jerry Fodor. *The Language of Thought*. Thomas Y. Crowell, New York, 1975.

Jerry Fodor. *Psychosemantics: The Problem of Meaning in the Philosophy of Mind*. Bradford Books/MIT Press, Cambridge, Mass., 1988.

Jerry A Fodor. Fodor's guide to mental representation: The intelligent auntie's vade-mecum. *Mind*, 94(373):76–100, 1985.

David A. Forsyth and Jean Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, Englewood Cliffs, NJ, 2002.

Alan Gilchrist. *Seeing Black and White*. Oxford University Press, 2006.

Paul Graham. *ANSI Common Lisp*. Prentice-Hall, Englewood Cliffs, NJ, 1996.

C. Larry Hardin. *Color for Philosophers: Unweaving the Rainbow*. Hackett, Indianapolis, 1993. (Expanded edition.).

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, 2009. Second edition.

Geoffrey E. Hinton. Learning multiple layers of representation. *TRENDS in Cognitive Sciences*, 11(10):428–434, 2007.

Jerry Hobbs and Robert C. Moore, editors. *Formal Theories of the Commonsense World*. Ablex Publishing Corporation, 1985.

W.H. Ittleson. *The AMes Demonstrations in Perception*. Princeton University Press, 1952.

Frank Jackson. *Perception: A Representative Theory*. Cambridge University Press, 1977.

Frank Jackson. Epiphenomenal qualia. *Phil. Quart.*, 32:127–136, 1982.

Frank Jackson. What Mary didn't know. *The J. of Philosophy*, 83(5):291–295, 1986.

Bela Julesz. *Foundations of Cyclopean Perception*. University of Chicago Press, 1971.

Amy Kind. Qualia. In *Internet Encylopedia of Philosophy*, 2008. URL `http://www.iep.utm.edu/qualia/`. Accessed at .

Robert Kirk. *Raw Feeling: A Philosophical Account of the Essence of Consciousness*. Oxford University Press, Oxford, 1994.

Stephen Kosslyn. *Ghosts in the Mind's Machine: Creating and Using Images*. Norton, New York, 1983.

Yann LeCun, B Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, and L.D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1:541–551, 1989.

William G. Lycan. *Consciousness*. MIT Press, Cambridge, Mass., 1987.

William G. Lycan. Consciousness as internal monitoring. *Phil. Perspectives*, 9:1–15, 1995a. (Reprinted as Lycan 1997).

William G. Lycan. A limited defence of phenomenal information. In Metzinger (1995), pages 243–58.

William G. Lycan. *Consciousness and Experience*. MIT Press, Cambridge, Mass., 1996.

William G. Lycan. Consciousness as internal monitoring. In Block et al. (1997), pages 755–771.

John McCarthy. Recursive functions of symbolic expressions and their computation by machine. *Comm. ACM*, 3(4):184–195, 1960.

Drew McDermott. *Mind and Mechanism*. MIT Press, Cambridge, Mass., 2001.

Drew McDermott. Artificial intelligence and consciousness. In Philip David Zelazo, Morris Moscovitch, and Evan Thompson, editors, *The Cambridge Handbook of Consciousness*, pages 117–150. Cambridge University Press, 2007.

Drew McDermott. Computationally constrained beliefs. *J. of Consciousness Studies*, 20(5–6):124–50, 2013.

Thomas Metzinger. *Being No One: The Self-Model Theory of Subjectivity*. The MIT Press, Cambridge, MA, 2003.

Thomas Metzinger, editor. *Conscious Experience*. Ferdinand Schoningh (English edition published by Imprint Academic), 1995.

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

Marvin Minsky. *Semantic Information Processing*. MIT Press, Cambridge, Mass, 1968.

Richard Montague. *Formal Philosophy: Selected Papers of Richard Montague*. Yale University Press, New Haven, 1974. Edited by Richmond Thomason.

Robert C. Moore. A formal theory of knowledge and action. In Hobbs and Moore (1985), pages 319–358.

Thomas Nagel. What is it like to be a bat? *Philosophical Review*, 83:165–180, 1974. [[pp?]] Also in (Nagel 1979).

Christopher Peacocke. *Sense and Content: Experience, Thought, and their Relations*. Clarendon Press, Oxford, 1983.

Benjamin Pierce. *Types and Programming Languages*. The MIT Press, 2002.

Maurice Pope. *The Story of Decipherment*. Thames and Hudson, London, 1999.

Zenon Pylyshyn. *Computation and Cognition: Toward a Foundation for Cognitive Science*. MIT Press, Cambridge, 1985.

Willard V.O. Quine. *Mathematical Logic*. Harvard University Press, Cambridge, MA, 1940.

Diana Raffman. On the persistence of phenomenology. In Metzinger (1995), pages 293–308.

Georges Rey. Dennetts unrealistic psychology. *Philosophical Topics*, 22(1/2: The Philosophy of Daniel Dennett):259–289, 1994.

John C. Reynolds. Polymorphism is not set-theoretic. Technical Report 296, INRIA Sophia-Antipolis, 1984.

David Rosenthal. Two concepts of consciousness. *Phil. Studies*, 49:329–359, 1986.

Chaitanya R. Sanna, Wen-Hsiung Li, and Liqing Zhang. Overlapping genes in the human and mouse genomes. *BMC Genomics*, 9:169, 2008.

Murray Shanahan. *Embodiment and the Inner Life*. Oxford University Press, 2010.

Sidney Shoemaker. The inverted spectrum. *J. of Philosophy*, 74(7):357–381, 1981.

Sidney Shoemaker. The inverted spectrum. In Block et al. (1997), pages 643–662.

Sydney Shoemaker. Self-knowledge and "inner sense": Lecture I: The object perception model. *Philosophy and Phenomenological Research*, 54(2):249–269, 1994a.

Sydney Shoemaker. Self-knowledge and "inner sense": Lecture II: The broad perceptual model. *Philosophy and Phenomenological Research*, 54(2):271–290, 1994b.

Sydney Shoemaker. Self-knowledge and "inner sense": Lecture III: The broad perceptual model. *Philosophy and Phenomenological Research*, 54(2):291–314, 1994c.

Michael Spivey. *The Continuity of Mind*. Oxford University Press, 2007.

Stalnaker. What might nonconceptual content be? In E. Villanueva, editor, *Concepts (Philosophical Issues)*. Ridgeview, Atascadero, 1998. 1998.

Walid Taha and Tim Sheard. MetaML and multi-stage programming with explicit annotations. *Theoretical computer science*, 248(1):211–242, 2000.

James W. Thatcher, Eric G. Wagner, and Jesse B. Wright. Data type specification: Parameterization and the power of specification techniques. In *Proc. Tenth Annual ACM Symposium on Theory of Computing*, pages 119–132, 1978.

Giulo Tononi. Consciousness as integrated information: A provisional manifesto. *Biol. Bull*, 215:216–242, 2008.

Tim van Gelder. The dynamic hypothesis in cognitive science. *Behavioral and Brain Sciences*, 21:615–665, 1998. (With commentary and rebuttal).

James Watson, Tania Baker, Stephen Bell, Alexander A.F. Gann, Michael Levine, and Richard M. Losick. *The Molecular Biology of the Gene, 6th Edition*. Cold Spring Harbor Laboratory Press, 2007.