Content, Computation and Externalism

ORON SHAGRIR

It is hardly disputed that the working hypothesis of cognitive science is that cognition is a form of computation; that theories in cognitive science construe human cognitive processes as a species of information processing. There has been less agreement, however, on the exact role of information or content in computational theories. Tyler Burge claims that mental content plays an *individuative* role in computational theories. On this view, computational theories of cognition are intentional in that they make essential reference to the content of mental representations.[1] In particular, Burge argues that a change in content may alter the computational identity of a cognitive system. But this view is not widely accepted. Most philosophers, though divided among themselves as to the exact role of content in theories in cognitive science, deny that content plays an individuative role in computational theories. On this majority view, computational theories make essential reference only to the syntactic properties of mental representations, and not to their content.[2]

[1] Burge (1986). See also Kitcher (1988), Davies (1991), Segal (1989, 1991), Morton (1993) and Shapiro (1997), who argue that Marr's computational theories of vision are intentional. These philosophers are divided on the question of whether (visual) content is extrinsic/broad or narrow: Burge, Kitcher, Davies and Shapiro argue it is broad, whereas Segal and Morton argue it is narrow.

[2] See Fodor (1980, 1994), Stich (1983), Egan (1995) and Butler (1998). Fodor thinks that content plays an individuative role in psychological theories, but not in computational theories (Fodor 1994; see also note 4). Egan thinks that content plays an explanatory but not individuative role in computational theories of cognition (Egan 1995; see also note 4). Stich

I concur with Burge's claim that content affects the computational identity of a cognitive system. But I believe that the arguments he has advanced in support of it are flawed. Like proponents of the majority view, Burge mistakenly assumes that the formal or syntactic structure assigned by computational theories to a cognitive system is invariant across contexts. Also mistaken is his thesis that specific content, e.g., being green, is part of computational description. I will demonstrate that content plays an individuative role in computational theories by way of determining which of the syntactic structures that the system implements is also its computational structure. My goal, therefore, is to defend a better argument for the claim that content affects the computational individuation of cognitive systems, one that avoids the errors of former arguments.

The paper opens with a brief survey of the standard view of computational individuation (section 1). Section 2 presents my argument for the claim that content impacts computational individuation. In section 3, I reassess the familiar arguments, put forward by Burge and others, for the claim that computational theories are intentional. I point out deficiencies in these arguments, and suggest ways in which they can be improved.

Before we proceed, I want to distinguish the thesis advanced in this paper (CI, for content impact) from two other theses about content and computation, and to locate CI within the ongoing debate over externalism in psychology. Here, then, are the three theses to be distinguished:

CI: Content impacts computational individuation – computational theories of cognition make essential reference to some features of content.

SE: Psychological content is broad/extrinsic – cognitive (computational) processes are defined over representations whose content is individuated, essentially, by reference

(1983) argues that content has no role - not even explanatory - in theories of cognition. These philosophers also disagree on the individuation conditions for content. Fodor (1980) and Butler (1998) argue that psychological content is narrow, whereas Fodor (1994) and Egan (1995) see it as "broad".

to features in the individual’s environment (Semantic Externalism).

CE: Computational theories of cognition are extrinsic – they make essential reference to features in the individual’s environment (Computational Externalism).

Burge argues for CE via CI and SE.[3] Indeed, it seems that if CI and SE are valid so is CE. In past, many philosophers challenged SE, but today SE has many proponents. Fodor (1994) and Egan (1995), for example, argue at length that broad content plays a major role in theories of cognition. Nevertheless, both explicitly reject CE, as they consider CI to be false.[4] This should come as no surprise, given that the arguments that have been offered for CI are unconvincing. My aim here is to correct this unfortunate situation, by advancing a different, and hopefully, stronger, argument for CI. Arguing for CI, however, is my only task. I do not defend SE here. I make some tentative suggestions with respect to CE in section 3, but these are not intended as a decisive argument.

It should be noted that I do not assume any particular theory of content. The main argument goes through with any familiar characterization of mental content, whether content is intrinsic or extrinsic, naturalized or non-naturalized. In specific examples, however, I assume that mental content is defined in terms of causal covariance, and that the “derivative” content of an artificial computing system is defined in terms of what is being represented, namely, in terms of the objects, properties and relations (or sets of objects) assigned by interpretation to the states of the system.

[3] Burge (1986). See also Kitcher (1998) Davies (1991). But see also Wilson (1994) and Bontly (1998), who argue for CE, but not via CI and SE. A more detailed exposition of Burge’s arguments is provided in section 3.

[4] More specifically, Fodor (1994) argues that broad content has an explanatory role via the primary role of intentional laws in psychological explanations. Yet, broad content has no individuative role in computational theories, whose task is to explain the intentional laws. Egan (1995) agrees with Fodor that “content does not play an individuative or taxonomic role in computational theories - a computational characterization of a process is a formal characterization” (p. 182). In her view, the explanatory role of content is connecting “the formal characterization of an internal process with the subject’s environment” (p. 182).

1. Computational individuation: The received view

Consider a physical system **P** that implements the abstract device **S** (Figure 1).[5] The *physical* system **P** is an electronic device, consisting of a pair of gates that receive and emit currents that range from 0 to 10 volts. One gate in **P** emits 5-10 volts if it receives volts larger than 5 from each of the two input channels, and 0-5 volts otherwise. The other gate emits 5-10 volts if it receives over 5 volts from exactly one input channel, and 0-5 volts otherwise. When we assign ‘0’ to emission/reception of 0-5 volts and ‘1’ to emission/reception of 5-10 volts the first gate becomes an *and-gate* and the second an *xor-gate*. Under this assignment, **P** receives pairs of digits as inputs and produces pairs of digits as outputs. The inputs of **P** are the inputs of each gate. The left hand digit of **P**’s output is the output of the *and-gate*, and the right hand digit is the output of the *xor-gate*. Overall, then, **P** can be seen as executing an algorithm for the *syntactic* function *f*:

‘0’,’0’ → ‘0’,’0’
‘0’,’1’ → ‘0’,’1’
‘1’,’0’ → ‘0’,’1’
‘1’,’1’ → ‘1’,’0’.

We can also describe what the system does in semantic terms, taking ‘0’ and ‘1’ as representations. For example, when the ‘0’ and ‘1’ are interpreted as representing numbers, **P** can be seen as computing addition (for the domain {0,1}). Under this interpretation, we can describe what **P** does in terms of the following relations between numbers:

$0 + 0 = 0$
$0 + 1 = 1$
$1 + 0 = 1$

[5] This device is introduced in Black (1990).

1 + 1 = 2

We thus have at least three different ways to describe what **P** does: a physical description (in terms of volts), a syntactic description (in terms of **S**), and a semantic description (in terms of an interpretation of the symbols ‘0’ and ‘1’). But which of these descriptions corresponds to the *computational* description of **P**? The received view is that it is the syntactic description. The computational identity of **P**, it is argued, derives from **P**’s implementing **S**. By implementing **S**, the states of **P** fall into computational types by their mapping relations to **S**. In particular, the computational identity of **P**’s input-output behavior is precisely the syntactic function *f*. On this view, the *semantic* interpretation does not correspond to the computational description of **P** since, from a computational point of view, it does not matter if we choose to interpret ‘1’ and ‘0’ as numbers or as colored hats. That is, the computational identity of **P** is the same if we interpret the ‘0’ and ‘1’ as representing numbers, colored hats or shapes. Nor does the *physical* description correspond to the computational description, because **P** is computationally equivalent to other physical systems whose physical descriptions are very different from **P**’s. The reason these different physical systems are computationally alike is that they share the same syntactic characterization **S**.

This is pretty much the received view about computational individuation. It is explicitly adopted by Block (1990), Fodor (1994), Egan (1995) and others, and I too accept its basic premise: I too believe that the computational structure of a system coincides with a syntactic structure, which it implements. On my view, however, it does not follow that content has no impact on a system’s computational individuation and description. Indeed, my objective here is to provide an argument for the claim that mental content does affect the computational identity of a cognitive system. The argument runs as follows: (1) A cognitive system, being a physical system, may simultaneously implement different syntactic structures. But (2) the computational structure of a system coincides with a *single* syntactic structure - what I call “the structure underlying the task in question”. Thus (3) to determine which syntactic structure constitutes the system’s computational structure, some other constraint must be invoked. (4) This constraint, I argue, is certain aspect of the mental content that the states of the system carry.

2. The argument: content and computation[6]

(1) A cognitive system may simultaneously implement more than one syntactic structure.

Let us first consider the general case. Take the physical system **P**. Suppose it turns out that flip detectors of **P** are actually tri-stable. Imagine, for example, that the *and-gate* in **P** emits 5-10 volts if it receives voltages higher than 5 from each of the two input channels; 0-2.5 volts if it receives under 2.5 volts from each input channel; and 2.5-5 volts otherwise. Let us also assume that the *xor-gate* emits 5-10 volts if it receives over 5 volts in one input channel and under 5 in the other; 0-2.5 volts if it receives under 2.5 volts from each of the input channels; and 2.5-5 volts otherwise. Let us now assign the symbol ‘0’ to emission/reception of under 2.5 volts and ‘1’ to emission/reception of 2.5-10 volts. Under this assignment, both the “*and-gate*” and the “*xor-gate*” are seen as *or-gates*, and **P** as implementing an abstract machine **S’** (Figure 2) whose input-output behavior is characterized by the syntactic function *f’*:

‘0’,’0’ → ‘0’,’0’
‘0’,’1’ → ‘1’,’1’
‘1’,’0’ → ‘1’,’1’

[6] The argument here is a massive expansion and correction of an argument in Shagrir (1999). In addition, the argument here is specifically directed at cognitive systems.

‘1’,’1’ → ‘1’,’1’

It thus follows that the very same physical system **P** implements not only an abstract system **S**, but also an abstract syntactic system, **S’**. As it implements **S’**, **P** can be described, not only by the syntactic function *f*, but also by the syntactic function *f’*. In other words, the syntactic structures **S** and **S’**, and the syntactic functions *f* and *f’*, constitute different syntactic descriptions of what **P** does.[7]

One might object that **P** implements but a single syntactic structure. Since **P’**s flip-detectors are tri-stable, it could be claimed, what **P** really implements is a “deeper” syntactic structure and function from which we can derive both “shallow” structures, **S** and **S’**, and both “shallow” functions *f* and *f’*. But, as we will see shortly, it does not matter for the rest of the argument whether **S** and **S’**, as well as *f* and *f’*, are “shallow” or “deep”. Computational taxonomies are indifferent to this distinction. We could also modify the example. Suppose that **P**’s detectors are bi-stable, so that **P** “really” implements **S**. Suppose, however, that the structure **S’** is also implemented in some other physical property of the very same spatio-temporal events that comprise **P**, say, the temperature of the gates.[8] In this example, there

[7] Note that my claim is more modest than the universal realizability claims advanced by Putnam (1988) and Searle (1992). Putnam proves that “every ordinary open system is a realization of every abstract finite automaton” (p. 121). Searle argues that “for any program and for any sufficiently complex object, there is some description of the object under which it is implementing my program. Thus for example the wall behind my back is right now implementing the Wordstar program, because there is some pattern of molecule movements that is isomorphic with the formal structure of Wordstar” (pp. 208-209). But my claim here is neither that every physical system implements an abstract computing system, nor that a given physical computing system implements any abstract computing system. My claim, rather, is that there are physical systems that can be seen as implementing more than one abstract computing system. Thus even if the universal realizability claim is false, as Chalmers (1996) and Copeland (1996) argue, my more modest claim can be correct. Indeed, nothing in arguments of Chalmers and Copeland invalidates (1).

[8] If, for example, the inputs and outputs of the gates are positive and negative charges, the temperature of the gate may depend on the absolute values of the currents. Thus we can implement one syntactic structure in the current values (as ‘+’ and ‘-‘), another, in the temperatures.

may be no deeper syntactic function implemented by **P** at all. It is thus clear that **S** and **S'** provide different individuation conditions for the computational identity of events and states of **P**.

Now what about the case where the physical systems are *cognitive* systems? There is no reason why a cognitive system, as a physical system, cannot simultaneously implement more than one syntactic structure. None of the considerations referred to are dependent on **P**'s being cognitive. It may certainly be the case that our visual system, for example, implements more than a single syntactic structure. Assume, for instance, that one syntactic type is associated with a neuron's firing between 0-5mv and another with its firing 5-10mv, and that as a result, the visual system can be seen as implementing a syntactic structure **S**. It is thus possible, as the argument about **P** demonstrates, that if a syntactic type is associated with the same neuron's firing 0-2.5mv and another with its firing 2.5-10mv, the visual system can also be seen as implementing a different syntactic structure **S'**.[9] In fact, the more complex the system, the greater the chances that it simultaneously implements more than one syntactic structure, as it has more processing units and connections whose values can be carved up in different ways. So if a very simple system such as **P** simultaneously implements more than a single syntactic structure, it is at least possible that a cognitive system, which is much more complex, will simultaneously implement multiple syntactic structures.

Let us now examine the implications of the multiplicity of syntactic implementation to *computational individuation*. As we saw, it is universally accepted that the computational identity of the system has something to do with the syntactic structure it implements. In particular, it has never been denied that two systems that implement different syntactic

[9] **S** and **S'** here are presumably more complex than the simple structures implemented by **P**. It would have also been more realistic to implement syntactic types in spiking rates, or in firing of 50mv, but I prefer to keep the neural example close to the artificial one.

structures fall under different computational types.[10] I do not take issue with these claims. The question, though, is how to understand the phrase “implement different syntactic structures”: does a computational taxonomy of a system takes into account all the syntactic structures the system implements, or just the syntactic structure underlying the task in question? By “the syntactic structure underlying the task in question” I mean the syntactic structure associated with a task we take the system performs. Assume, for example, that a task of **P** is to compute addition. In this case, the syntactic structure associated with **P**’s performing addition is **S**. **P** also implements **S'** and perhaps many other structures. Nevertheless, the syntactic structure associated with, and so underlying, **P**’s performing addition is neither **S'** nor **S&S'**, but **S**. The question, then, is whether a computational taxonomy counts all the syntactic structures **P** implements, or just the syntactic structure underlying **P**’s performing addition. I contend that the latter is the case:

(2) A computational taxonomy of a cognitive system takes into account just the syntactic structure underlying the cognitive task in question.

Suppose I use **P**, which implements **S**, to compute addition, and you use a different physical system, **P’**, which also implements **S**, to compute addition. We would surely take **P** and **P’** to be computationally equivalent, regardless of whether or not **P’** also happens to implement **S’**. Indeed, if it turns out that **P’**, unlike **P**, does not implement **S’**, we would say that there are other contexts, where **P** and **P’** are used for other purposes, and in which they are not computationally equivalent. But we will certainly would not doubt that **P** and **P’** are computationally equivalent in their current task (i.e., performing addition). This demonstrates

[10] Burge and Davies challenge the opposite claim: that two systems that are computationally different must implement different syntactic structures. They hold that systems that carry different content are computationally different, even if they implement the same syntactic structure (see section 3). However, they seem perfectly comfortable with the assertion that two systems that implement different syntactic structures are computationally different.

that the computational identity of a system is coupled with the syntactic structure underlying a specific task the system performs.

Let us turn to cognition. I hope to show that computational theories of cognition do and must take into account just syntactic structures underlying the cognitive tasks in question. Consider first the case of two cognitive modules whose underlying syntactic structures are different, although the modules implement the same class of syntactic structures. Imagine (following Davies 1991) that a component of the visual system, called visex, computes a representation of depth of the visual scene from binocular disparity. There also exist, however, remote creatures whose auditory system has a component, called audex, whose microphysical structure is identical to that of visex. Audex, however, computes a representation of certain sonic properties. Also assume that the syntactic structure underlying visex is **S**, and the one underlying audex is **S'**. Whether this assumption is valid at all will be discussed in section 3. The question we ask here is whether computational taxonomies count visex and audex as identical or distinct. And the answer, I believe, is no. Computational theories of visex and audex would provide different computational descriptions. That is, a computational theory would describe visex through the syntactic structure underlying its visual task, namely **S**, and audex through the syntactic structure underlying its auditory task, namely **S'**. Indeed, vision theorists do, in point of fact, cite only the syntactic structures underlying the specific visual tasks they are studying. The possibility that our visual system simultaneously implements other syntactic structures, that these structures might even underlie other cognitive tasks, in other scenarios, is of little interest to vision theorists, and does not alter the computational description they put forward. That visex also implements **S'** only indicates that there are other scenarios in which theories describe visex by **S'**. This could be the case, for example, where visex has been transplanted into brains of other creatures, and now implements **S’** to support, say, an auditory task. But with regard to the tasks in question, the computational descriptions of visex and audex are different.

Or consider another pair of cognitive systems, visex and viSex. This time we compare systems whose tasks and underlying syntactic structures are the same, while the classes of syntactic structures they implement are different. This case is analogous to that of the artificial systems **P** and **P’**. Visex is the module that computes depth from disparity in Mary’s visual system, and viSex is the module that computes depth from disparity in Paul’s visual system. The same syntactic structure, **S**, underlies both visex and viSex. But there are also slight physical differences between visex and viSex, which result in slight differences in the syntactic structures they implement. It turns out, for example, that visex but not viSex also implements **S’**. Now, again, there is little doubt that a computational theory of vision will come up with the same computational descriptions for both systems – the theory will use **S** to describe both visex and viSex. Indeed, computational theories of vision identify those syntactic structures underlying the visual systems of *all* individuals. It is quite possible that Mary’s visex, but not Paul’s viSex, simultaneously implements another syntactic structure, say **S’**, because one of its neurons is tri-stable. But this possibility does not alter the computational description of these systems, as visual systems.

I have argued that theories of cognition describe systems as computationally equivalent or distinct relative to the cognitive task in question. In the specific scenario described above, they would count visex and audex as computationally distinct, and visex and viSex as computationally equivalent. I now suggest that this mode of taxonomy is no mere convenience, but is imperative. The alternative, which takes into account all the syntactic structures the system implements, undermines the prospects of cognitive science to quantify over different individuals. For it may be the case that systems that physically systems always implement different classes of syntactic structures. Were computational theories of vision to describe visex and viSex differently, it might ensue that no theory of cognition could make any generalizations. To insist that the computational descriptions of visex and viSex are different is thus to endanger the hope underlying cognitive science, namely, that computational theories can generalize over different individuals.

To sum up, I began by showing that artificial systems are considered computationally equivalent or distinct relative to a task they perform. I then demonstrated that computational theories of cognition classify systems according to the syntactic structures underlying the cognitive tasks they perform. Finally, I argued that theories of cognition must choose this mode of computational taxonomy if they hope to sustain a science of cognition. This suffices to establish that computational taxonomies classify cognitive systems into types according to the syntactic structure underlying the task they perform, and not according to the class of all syntactic structures they implement.

It might be objected that I am confusing metaphysical and epistemological considerations. The objection is that I am identifying computing cognitive systems relative to the explanatory agenda of the observers, though this agenda is not an essential component of the identity conditions of computing systems. The identity conditions of a computing system consist solely of all the syntactic structures the system implements. In response to this line of argument, I must stress that none of the claims I have advanced so far make the identity conditions of cognitive systems observer-relative. First, I do not claim, as does Searle (1992), that syntax is not “intrinsic” to physics. My claim is only that a physical system may simultaneously implement more than one syntactic structure. We can safely assume that all these implemented structures are intrinsic. Second, when I say that the computational identity of a system is task-relative, I by no means imply that the task is observer-relative. Observers are free to choose whether or not they want to explain how visex extracts depth from disparity. But their choice does not change the fact that visex does extract depth from disparity, just as their choice does not change the fact that visex does not digest food. And finally, linking syntactic structures with the task performed by visex is similarly not a matter of observers' choice. Observers are free to characterize visex using any of the syntactic structures visex implements. But the freedom to choose does not mean that the structure coincides with the extraction of depth from binocular disparity. If observers wish to provide a syntactic description of **P**'s performing addition, they must use **S**, and not **S'**, to describe **P**. Likewise, if observers wish to provide a syntactic characterization of visex extracting depth from disparity, they must choose the syntactic structure underlying this visual task. In short, nothing in my claims makes the computational identity of a cognitive system observer-relative. Observers may or may not choose to provide a computational characterization of a cognitive system. But should they choose to do so, they have only one option: they must characterize the system by means of the syntactic structure underlying the cognitive task the system performs.

It might still be contended that my argument depends on a certain conservatism about task descriptions: a cognitive system is seen as falling under one task description, and therefore, is computing only one function, namely, the function associated with this one task description. But, it could be argued, a cognitive system has many task descriptions. And moreover, in describing a cognitive system as a computing system one cannot weed out any of the other task descriptions. There are always other functions, associated with the other task descriptions, that the system could be interpreted as computing.[11] It thus seems that a

[11] This claim is advanced by Cummins (1989): "It is clear that any system that simulates *f* is bound to simulate a lot of other functions as well" (p. 106), and by Haugeland (1978): "Of course, simply specifying the interpretation does not convince us that the object really plays

cognitive system computes not just the function associated with one cognitive task, but also the functions associated with other tasks. Hence, a computational taxonomy of a cognitive system should take into account not just the syntactic structure underlying a given cognitive task, but also all the syntactic structures underlying all relevant tasks. That vision theorists take into account only the syntactic structure underlying a specific visual task reflects the fact that they are motivated by epistemological and pragmatic considerations. But these considerations do not reflect the deep individuation conditions of the visual system. The identity conditions of a computing visual system consist of all the syntactic structures that the system implements.

However, I do not insist that a cognitive system performs but a single task, or falls under one task description. My claim, rather, is that a computational taxonomy of a cognitive system *individuates* the states of a system relative to one particular cognitive task that the system performs. Let me explain. On the one hand, I agree that we cannot force a unique semantic description of what a cognitive system does. Moreover, I have argued that we cannot even always establish a unique syntactic description of what the system does. I thus agree that a cognitive system has many task descriptions. But on the other hand, I insist that a computational taxonomy of a cognitive system does not take into account the implemented syntactic structures associated with all these task descriptions, but only the syntactic structure underlying the cognitive task in question. More specifically, my claim is that a computational theory of vision does and must individuate the states of a system, *qua visual system*, relative to a specific visual task that the system performs.

How do we decide between these two alternatives? How do we tell that a

chess… With a little ingenuity, one can stipulate all kinds of bizzare “meanings” for the behavior of all kinds of objects; and in so far they are just stipulations, there can be no empirical argument about whether one is any better than another.” (pp. 253-4).

computational taxonomy takes into account the syntactic structure underlying a cognitive task, and not the other syntactic structures? The question here is not what constitutes computation. We are not investigating whether cognitive scientists use the term 'computation' properly.[12] What interests us is the conception of computation underlying cognitive science, hence, we should be asking what is the *essential* feature of a cognitive system to which cognitive scientists refer when they describe this system as a computer. And the answer, I argued, is that the essential feature is, and cannot be other than, the syntactic structure underlying the cognitive task in question. In describing the visual system as a computing system, the syntactic structure underlying the visual task is the only relevant feature, and must be the sole relevant feature if we wish to sustain a science of cognition. Perhaps another conception of computation is possible, on which taxonomies consider all syntactic structures. But my point is that this alternative conception is not used, and has no use, in cognitive science.

An analogy might help here. Suppose a physicist were to describe a piece of wood and a piece of metal that are at the same temperature as physically equivalent. Some would surely insist that the physicist is wrong: the pieces are not really physically equivalent (just as visex and viSex are not really computationally equivalent). They are simply at the same temperature (just as visex and viSex have the same underlying syntactic structure). Equivalence, they would argue, is measured by the totality of a system's physical properties (just as computational equivalence is measured by the totality of syntactic structures the system implements). My reply to this contention is that the analogous question we should be focusing on is not whether the physicist is using the term 'physically equivalent' correctly (just as our focus is not whether vision theorists use the term 'computation' correctly). The

[12] Elsewhere (Shagrir 1999), however, I argue that taking in account only one syntactic structure is in accord with the true nature of computation.

question we should be asking (the answer to which is temperature) is by virtue of which feature of the wood and the metal does the physicist describes them as equivalent (just as our focus is on the notion of computation as it is used in cognitive science). On this approach, we observe that vision theorists describe, and indeed must describe, visex and viSex as computationally equivalent, and inquire into the essential feature of visex and viSex by virtue of which vision theorists describe them as equivalent. We then conclude that visex and viSex are computationally equivalent because the structure underlying their visual task is of the same syntactic type. The syntactic structure underlying the cognitive task, therefore, *is and must be* the fundamental measure of computational identity in cognitive science.

It follows from (1) and (2) that:

(3) The computational identity of a cognitive system is not determined solely by the syntactic structure(s) the system implements.

If a cognitive system, as a physical system, may implement more than a single syntactic structure, as (1) asserts, and if the computational taxonomy counts only the syntactic structure underlying the task in question, as (2) asserts, then there must be an additional constraint that determines which of the implemented structures constitutes the system's computational structure. Moreover, if a cognitive system simultaneously implements several different syntactic structures – if its intrinsic physical/neural properties simultaneously realize different syntactic structures – then surely these intrinsic physical/neural properties cannot, by themselves, serve to explain why the computational identity of the system is given by one syntactic structure rather than another.[13] But what else could determine computational identity? What could explain why the computational identity of the system is conferred by one syntactic structure rather than another? I contend that:

[13] "Intrinsic" properties are non-relational properties, namely, properties that a system has by virtue of itself.

(4) The computational identity of a cognitive system is affected by the content of its states.

Let us first observe that the way we talk about what is being computed implies that content impacts computational identity. Systems often compute functions. But the functions a system computes are always characterized in semantic terms – that is, in terms of the content of its states. We say that a system computes 34+56 (where the function '+' is defined over numbers, not numerals), or the next move of the white queen on the chess board, or the shape of a distal object. What does this characterization tell us? Claim (2) implies that the computational identity of a system is conferred by the syntactic structure underlying a task the system performs. This task is often presented as the function the system computes. But as I just pointed out, this function is characterized in semantic terms. This implies that there is a close relationship between a system's computational identity and the semantic characterization of the task in question.[14] To put it differently, we saw – via (2) – that the computational identity of the system is related to a task the system performs. We also saw that the characterization of a task in syntactic or physical terms does not explain the relationship between the task and computational identity, as other syntactic functions are also implemented. It is therefore no accident that we characterize the task in semantic terms. Such characterization indicates that computational identity is affected by content: content determines, at least partly, which of the implemented syntactic structures is the computational structure of the system.[15]

[14] There are systems whose task cannot be described as computing functions, e.g., interactive systems. However, the tasks of these systems are also characterized in semantic terms, e.g., controling the traffic. So these systems also fall in the scope of the current argument.

[15] We also have other evidence that content determines computational identity. It is notorious that the standard definitions of computation are altogether unsuccessful at distinguishing computation from other physical dynamics. It therefore stands to reason that computing systems (but not other physical dynamics) are systems whose processes are partly

The main argument for (4) proceeds by elimination. Recall that we are trying to ascertain what features determine which syntactic structure is the computational structure of the system. We have already eliminated the possibility that the syntactic or the intrinsic physical/neural properties of a cognitive system play this determinative role. We can also eliminate the so-called neural transducers that surround the system. The neural transducers are surely important, but it seems that computational taxonomies are indifferent as to how the transducers mediate the information flowing to and from the cognitive system as long as their inputs and outputs are the same. This is readily apparent if we leave unchanged the information entering the cognitive system fixed, but alter the mechanisms that carry the information. Assume, for example, that we replace the mechanisms that transduce the light waves hitting the retina into representations of light intensities with other transducing mechanisms that yield the same result, but have a different structure. It is clear that the computational identity of the visual system will not vary. The computational identity remains the same as long as what is being transduced is the same. Different transducers will make a computational difference only if what they transduce is different. Thus, the only features of transducers that could be relevant are the inputs of the sensory transducers, their behavioral consequences, and the information being transduced. But the inputs of the sensory transducers and their behavioral consequences are often, if not always, "analog". It is therefore hard to see how these inputs and outputs could contribute to identifying the preferred syntactic structure. After all, our earlier toy example clearly demonstrates that we can match the analog inputs and outputs – receiving and emitting 0-10mv – to all pertinent syntactic structures[16]. We are therefore left with two sorts of features that could explain the

individuated by the content of the representations over which they are defined. See Shagrir (1999) for further discussion of the problem and a proposed solution.

[16] If the values are "digital" we can apply the original argument to the system that consists of

choice of computational structure: environmental features correlated with the intrinsic physical/neural properties of the cognitive system, and phenomenal features (conscious experiences) correlated with these neural properties. But both kinds of features are precisely the ones we associate with the content of the system's states. Thus content impacts computational individuation.[17]

This argument is reinforced by our non-cognitive example. States of the same physical system **P**, we saw, can fall under different syntactic types, say **S** or **S'**. As we further observed, states of **P** can fall under different computational types when **P** is used for different tasks. Yet nothing in the intrinsic properties of **P** compels us to favor one syntactic structure over another. There must be another constraint, *external to* **P**, that determines which syntactic structure constitutes the computational identity of **P** in a given context. In this example, there need not be any transducers. Thus, the only factor that could somehow make the difference is features of (derived) content of the states of **P**. Assuming there are significant affinities between the artificial and the natural, we can infer that content impacts the computational identity of a cognitive system.[18]

the cognitive system *plus* its transducers (see the duck/rabbit example in note 18).

[17] At this point we can rule out the option that defines content by "narrow" functional-computational role. For it follows from (1)-(3) that the computational role of the system is at least partially defined by features of content. Yet, it might be the case that "wide" functionalism is correct: the computational structure is partly determined by the causal relations of the intrinsic neural states with distal objects, whereas content is defined by these causal relations *plus* the inner relations defined by the chosen syntactic structure.

[18] Some might object that **P** can be considered as a computing system only if its inputs and outputs are connected to input and output devices such as a keyboard and monitor. I do not think that a computing system must have extra input/output devices. But even if it does, the extra devices do not provide a unique syntactic structure. Imagine that **P** generates (on the screen) the legendary duck/rabbit image, which can be seen either as a duck or a rabbit. Imagine it turns out that the structure (or algorithm) underlying the generation of a duck image is **S**, but the one underlying the rabbit image is **S'** (I am indebted to Jack Copeland for suggesting the example). What is the computational identity of **P**? I would say it is **S** when **P**'s task is to compute a duck, and **S'** when its task is to compute a rabbit, and perhaps another structure, **S''** or **S&S'**, when its task is to compute a duck/rabbit image. Thus adding

I have argued that content impacts computational individuation. One might wonder how is it at all possible for content to determine computational identity. Here is one way. Assume that our cognitive system implements two syntactic structures **S** and **S'**. Suppose that in **S** a neuron firing 0-5mv is correlated with one syntactic type and its firing 5-10mv is correlated with another syntactic type, whereas in **S'** the neuron firing 0-2.5mv is correlated with one syntactic type, and its firing 2.5-10mv is correlated with another syntactic type. I contend that **S** will be preferred over **S'** if it turns out that the neuron fires 0-5mv whenever one type of objects (e.g., green objects) is detected, and it fires 5-10mv whenever another type of objects (e.g., red objects) is detected. I am thus suggesting that the content correlated with the neural properties determines, at least partially, which syntactic structure constitutes the computational identity of our cognitive system. The computational structure of the system is the syntactic structure that reflects the correlation between neural properties and semantic properties.

I foresee two objections to this proposal. One objection that might be raised is that the proposal conflicts with familiar examples which arguably show that a change in content – say, from colors to shapes – does not alter the computational identity of the system. The claim is that the computational identity of the system will still be **S** also in a case where firing 0-5mv covaries with round objects, and firing 5-10mv, with rectangular objects. One response to such examples is to insist that such changes in "specific" contents do indeed alter the computational identity of a cognitive system.[19] But this is certainly not my view. My proposal is not compromised by these examples, as I do not claim that *every* change in content alters computational identity. The features that make a computational difference, in my view, are formal features, that is, set-theoretic relations and other high-level mathematical

the extra devices does not always help to establish computational identity.

[19] This strategy is taken by Burge and by Davies (see section 3).

relations among the represented objects. Consider again the neuron firing at 0-10mv. My proposal is that **S** will be preferred to **S'** because emitting 0-5mv covaries with one *class* of objects and emitting 5-10mv with a different *class*. The relevant formal property here is "belongs to" (being part of the same class), that is, that all the covaried distal objects share the formal property of being part of the same class. The specific non-formal property (e.g., being green) that these covaried objects have in common can vary (e.g., from being green to being rectangular) without any variance in taxonomical choice. But were the formal properties of these objects to vary – from a situation where objects covaried with emission of 0-5mv belong to the same class, to one they belong to different classes – then so would the taxonomical choice.[20]

A second objection to my proposal might be that it conflicts with the multiple realizability of computational structures. It is elementary that we could also implement the structure **S** in netware where neurons flip at 2.5mv (rather than at 5mv), and also in many other types of hardware. My proposal seems to threaten the idea that all these systems are computationally equivalent, for it seems to suggest that having the cut-off at 2.5mv rather than 5mv makes a computational difference. But this is not the case. My suggestion is perfectly compatible with multiple realization. I am not saying that changing the cut-off point alters computational identity of the system. Rather, I am saying that content explains the alteration in computational identity in cases where a different cut-off point alters

[20] Frances Egan (private communication) wonders why we should consider set-theoretic features as aspects of mental content. My reply is that these are features of content because they are higher-order mathematical structures of the objects in the represented domain (Following Gila Sher (1996), we can call these features of content "formal contents"). But even if the set-theoretic features are not content-related, content still determines computational identity. For example, we can alter computational identity by altering the specific content of some of the tokens of the syntactic type associated with emission of 0-5mv. This change in content will alter the set-theoretic features that are correlated with the intrinsic neural properties, and so will alter the computational identity of the system.

computational identity. Let me explain. As we saw, it is possible that one syntactic structure, **S**, is correlated with having the cut-off at 5mv, and another, **S'**, with having it at 2.5mv. As we also saw, a computational taxonomy picks out either **S** or **S'** (or neither, but not both). Thus the individuation depends on the location of the cut-off point. But what does determine the location? My suggestion is that the location is a function of the relations between intrinsic neural properties and formal properties of the represented objects. Features of content determine whether the cut-off is at 5mv or 2.5mv. Thus, in our example, where the syntactic structures correlated with each cut-off are different, locating the cut-off at 2.5mv rather than at 5mv makes a computational difference. The computational identity will be **S'** and not **S**. Yet, it does not follow that there is no room for multiple realization. In the cases relevant to multiple realization – where the syntactic structure correlated with the cut-off at 5mv is the same as the syntactic structure correlated with the cut-off at 2.5mv – locating the pertinent cut-off at 2.5mv instead of 5mv does not make a computational difference. Thus, the same syntactic structure **S** could be implemented in netware whose neurons flip at 2.5mv as well as in other hardware.[21]

[21] My idea, in other words, is that "formal content" (see note 20) plays a role in determining the location of the cut-off of neural/physical properties, and so in determining the syntactic structure which constitutes the computational structure of the system. This leaves two open questions: whether every difference in formal content involves a computational difference, and whether every computational difference involves a difference in formal content. One might assume I would answer the first question is the negative, in light of my contention that **S** could be implemented in netware whose neurons flip at 2.5mv. But I would actually answer in the affirmative, at least when we consider the system *as a whole*. Two systems whose computational structure is **S** carry the same formal content even if the neurons flip at 5mv in one system and at 2.5mv at the other. In particular, if the whole system is a neuron firing at 0-10mv, then the formal content will be the same, regardless of whether the cut-off is at 5mv or 2.5mv. The formal content in both cases is comprises of *two distinct classes*.

One might further suppose that I would answer the second question in the negative. For it seems that it is also possible that two different computational structures could have the same formal content. If the formal content of neurons firing at 0-10mv can be the same regardless of the cut-off point, we could also have two systems, one whose computational structure is **S**, the other whose structure is **S'**, whose states have the same formal content.

The argument is now complete. If (1) a cognitive system may implement more than one syntactic structure, then since (2) the computational identity of the system is given by the syntactic structure the system implements in performing its cognitive task, (3) there must be another constraint that determines which syntactic structure is relevant to the computational identity of the system. And this constraint, I have argued, involves the content of the system's states. Thus (4) mental content impacts the computational identity of cognitive systems.

3. Computation and externalism

Psychological externalists have also argued that content impacts the computational identity of cognitive systems. The most renowned argument is Burge (1986), who seeks "to correct the impression, often conveyed in recent philosophy of psychology, that intentional theories are regressive and all of the development of genuine theory in psychology has been proceeding at the level of purely formal, 'syntactical' transformations (algorithms) that are used in cognitive systems" (p. 29). Burge has, in fact, employed CI to advance his argument for the

This possibility is worrisome, because it seems to conflict with my suggestion that formal content determines computational structure. For it appears that formal content cannot always determine which syntactic structure, **S** or **S'**, is the computational structure.

But there is no contradiction here. My proposal is that formal content determines which of the syntactic structures implemented by the *same* system is its computational structure, that is, the suggestion is that formal content determines computational identity via the relations of content with the neural properties. Within the same system, the formal content of **S** must differ from that of **S'**.

Moreover, I would answer the second question in the affirmative. It is true that the formal content of neurons firing at 0-10mv could be the same regardless of the location of the cut-off. But in this case, the syntactic structure implemented by firing 0-10mv is also the same – {'0','1'}. Note that in our early toy example, the location of the cut-off at 2.5 or 5 volts impacts higher-order syntactic relations. When the cut-off is at 5 volts the system performs the operation *and*, whereas when the cut-off is at 2.5 volts the system performs the operation *or*. But this computational difference entails a difference in formal content. The *and-gate* implemented by one system mirrors one set-theoretic relation (i.e., intersection of two distinct classes), whereas the *or-gate* implemented by the other system mirrors quite another set-theoretic relation (i.e., union of two distinct classes).

claim that computational theories of cognition make essential reference to features in an individual’s environment (CE). His central argument consists of the claims that computational theories of vision are intentional in that they make essential reference to visual content (p. 31) (CI), and that “intentional content – is individuated in terms of the specific distal causal antecedents in the physical world” (p.32) (SE). From this pair of assertions Burge goes on to conclude that “individualism is not true for the [computational] theory of vision” (p. 34) (CE).

But I believe that the arguments Burge, Kitcher, Davies and others have put forward for CI – the arguments that rest on thought-experiments and on exegesis of Marr’s theories of vision – are unconvincing. These arguments are intended to support the view that content affects computational individuation, but in fact play into the hands of those who consider CI, and thus CE, false. The deficiency of these arguments, in my view, is their assumption that the computational description of a cognitive system includes not only the syntactic structure the system implements, but also specific content. In what follows, I will point out the problems in these arguments, and show how they can be corrected. In so doing, I hope to demonstrate that CE is a viable philosophical position.

3.1 Thought Experiments

Burge (1986) and Davies (1991) advance thought experiments to support CI. Using these experiments, they hope to show that a change in physical environment can alter the computational identity of cognitive systems. Let us focus here on the visex/audex thought experiment (Davies 1991). In this experiment, visex is a component of the visual system that computes a representation of depth of the visual scene from information about binocular disparity, whereas audex, a component in the auditory system of some creatures, computes the representation of certain sonic properties. Visex and audex have the same intrinsic microphysical structure. Were we to remove a particular audex from its normal environment and plugged it into a visex slot, it would now compute depth from disparity. But what can we conclude from this thought experiment? Is the difference between audex and visex, when each is embedded in its normal environment, a *computational* difference?

Davies thinks it is. He argues that the computational theory of visex "generalizes over visexes qua components of the visual system" (1991, p. 482), whereas the computational theory of audex generalizes over audexes qua components of the auditory system. But since the visual content of visex and the auditory content of audex are arguably different, the states of visex and audex fall under *different* computational types. And since the content of visex/audex is arguably extrinsic, the computational difference results from environmental differences.

Egan (1991, 1995) argues that Davies's conclusion does not follow. Egan and Davies share some ground. They agree that the content of a perceptual state is partly determined by the extrinsic physical environment. Consequently, they agree that the contents of the states of visex and audex, in their normal environments, are different. They thus agree that the intentional descriptions of the processes in visex and audex are different. Egan, however, reminds us that we have to keep the intentional and the syntactic descriptions of visex/audex separate. She then argues that the computational description of visex/audex coincides with its syntactic description, and not with its intentional description. Nothing in the thought experiment contradics this. But since the syntactic descriptions of visex and audex are alike, it certainly seems that the states of visex and audex fall under the same computational types. Egan therefore understands the thought experiment as vindicating her claims that content does not play any individuative role in computational theories of cognition, and that computational theories of cognition are individualistic.

My position on the thought experiment is that Davies and Egan considered only a limited number of scenarios. In these scenarios, the syntactic structures underlying visex and audex are the same. The debate between them thus turns only on whether the different contents of visex and audex make a computational difference (Davies) or not (Egan). I believe that Egan is right about these scenarios. Visex and audex are computationally equivalent because their underlying syntactic structures are the same. There are also, however, other scenarios, that Davies and Egan do not even consider, scenarios in which the syntactic structures underlying visex and audex are different. That is, it is possible that the syntactic structure underlying visex, which executes a visual task, is **S**, and the syntactic structure underlying audex, which executes an auditory task, is **S'**. This might be the case if, for example, it turns out that neural receptors in visex fire at 0-5mv upon detecting one kind of light intensity, and at 5-10mv upon detecting another. Whereas, in an auditory environment, the same receptors fire at 0-2.5mv upon detecting one kind of sound intensity, and at 2.5-10mv upon detecting another. Consequently, it may turn out, as we saw above, that "neural gates" in visex function as *and-gates* whereas their counterpart gates in audex function as *or-gates*. The syntactic structures underlying the visual task of visex and the auditory task of audex will therefore differ.

Note that I do not claim that visex and audex implement different *classes* of syntactic structures. They do, in fact, implement the same class of structures. The claim is rather that the syntactic structure underlying visex (as a visual module) is different from the one underlying audex (as an auditory module). Moreover, I do not deny that, in these scenarios, there are also differences in the neural surroundings of visex and audex. Such differences exist simply because some mechanisms transduce light waves into electro-chemical properties upon entering visex, while other mechanisms transduce sound waves into electro-chemical properties upon entering audex. *But these differences must exist in any case*. They also exist in the cases discussed by Davies and Egan, where the syntactic structures underlying visex and audex are the same. In other words, the question here is not whether transducers are different: we know they are. The question is whether the underlying syntactic structure of a physical module remains unchanged even when the module is located in different neural and distal environments. Davies and Egan both assume that the syntactic structures underlying visex and audex are the same. But, as has been just showed, this assumption is false. There are scenarios in which the syntactic structures underlying visex and audex are different.

I have argued that Davies's argument is unconvincing, as Davies considers only scenarios where the syntactic structures underlying visex and audex are the same. In these scenarios, as Egan convincingly points out, visex and audex are actually computationally equivalent. There are, however, other scenarios where the syntactic underlying structures are different. And based on these scenarios, we can reconstruct an argument for CE, as follows:

E1: There are scenarios in which the syntactic structures underlying visex and audex are different.

E2: On these scenarios, the computational identities of visex and audex are different. That is, visual computational theories will use **S** to characterize visex, and auditory computational theories will use **S'** to characterize audex.

E2 follows from the same considerations that led to claim (2) of my argument, namely, the claim that computational taxonomies pick out only syntactic structures underlying the cognitive task in question.

E3: The computational identity of at least some cognitive systems – e.g., cognitive modules – may vary across contexts.

Imagine that visex is moved to an audex slot. It is possibe, according to E1, that the underlying structures of visex and audex will be different. On this scenario, according to E2, the computational identities of visex and audex will differ too.

E4: The computational identity of some cognitive systems (e.g., modules) is at least partly determined by features external to them.

Consider a scenario in which visex and audex are computationally different. Visex and audex have exactly the same intrinsic physical properties, so their states fall under the same neural/physical types, but different computational types. The computational difference must thus be determined by features external to visex and audex.

CI: Content affects the computational identity of at least some cognitive systems.

This claim basically follows from the same considerations that led to claim (4) of my argument.[22] But let us also see how content affects the computational individuation of visex and audex. Assume that the computational identity of visex is given by **S** and that of audex by **S'**. This is the case, we assumed, if neural receptors in visex fire at 0-5mv upon detecting one kind of light intensity, and at 5-10mv upon detecting another. Whereas, the same receptors, in an auditory environment, fire at 0-2.5mv upon detecting one kind of sound intensity, and at 2.5-10mv upon detecting another. What makes the computational difference here is not the difference in specific content (i.e., light vs. sound properties), but the difference in formal (i.e., set-theoretic) structure: for example, that a neuron's firing at 0-5mv is correlated with one class in the visual field, but with two distinct classes in the auditory field. Were the pertinent *formal* relations in the distal fields the same, the computational identity of visex and audex would also be the same. Visex and audex could both be characterized via **S**, even though their specific contents are different. This is the case in the

[22] It will actually take some work to apply these considerations to the case at hand because visex and audex are also surrounded by other *cognitive* components.

scenarios considered by Davies and Egan. In these scenarios, a neural property is correlated with one property/relation in the visual field just in case this neural property is correlated with a single property/relation in the auditory field. Thus in these scenarios visex and audex are computationally equivalent.

Now what about CE, the claim that computational theories of cognition make essential reference to features in the individual's environment? I am not sure how to infer CE directly from E1-E4 and CI. The problem is that it could be argued that content is defined by relations to other cognitive states outside visex/audex that are inside the individual, or by phenomenal aspects correlated with brain properties that are outside visex/audex. But if the contents of visex and audex are determined by features in the embedding external environments (SE), *as both Davies and Egan apparently assume*, then, given CI, it is broad content, and thus distal features, that makes the computational difference between visex and audex. Thus if SE is assumed, we can conclude that:

CE: Computational theories of cognition make essential reference to features in the individual’s environment.

3.2 Marr's theories of vision

Let us now consider the arguments that have been at the center of the debate over psychological externalism, namely, those surrounding David Marr’s computational theories of vision. Marr (1982) distinguishes the computational level of description from the algorithmic. The major task at the computational level is to provide “an analysis of how properties of the physical world constrain how problems in vision are solved” (Hildreth and Ullman, 1989, p. 582),[23] whereas the algorithmic level provides a description of the process

[23] In Marr’s terms: “In the [computational] theory of visual processes, the underlying task is

itself. Take, for example, Marr's computational theory of stereopsis (stereo vision). The first part of this theory is concerned with measuring disparity, that is, the relative angular discrepancy in the positions of objects on the two retinal images.[24] Marr formulates three constraints on the match between the left and right images, and argues that when the constraints are satisfied, and the image contains a sufficient amount of detail, the match between the two images is physically correct and hence unique. The continuity constraint, for example, asserts that the disparity between the images varies smoothly. This constraint results from the contingent physical fact that matter is cohesive – separated into objects whose surfaces are smooth, in the sense that surface variation is small compared to the distance from the observer. Things in the world that give rise to sharp intensity changes, e.g., objects' boundaries, are spatially localized and occupy a small fraction of the area of an image. It therefore follows that disparity does not normally exhibit too many discontinuities.

Burge (1986) has used this and other examples to claim that CI: "The top (computational) levels of the theory [of vision] are explicitly formulated in intentional terms. And their method of explanation is to show how the problem of arriving at certain veridical representations is solved" (p. 35). Burge also argues (pp. 32ff.) that SE: The "information or content of the visual representations is always individuated by reference to the physical objects, properties, or relations that are seen" (p. 34). Burge then concludes that CE: The taxonomy of computational states essentially refers to the physical environment.[25]

to reliably derive properties of the world from images of it; the business of isolating constraints that are both powerful enough to allow a process to be defined and generally true of the world is a central theme of our inquiry." (p. 23).

[24] This task is also known as the stereo-matching problem (pp. 111-116). The second part of the theory concerns the use of disparity to estimate the relative distances of the objects from the viewer (pp. 155-159).

[25] Burge's argument actually has additional three steps, though it is obvious that CE follows from SE and CI alone. The other three steps constitute a thought-experiment that dramatizes the possibility of the computational identity of the visual system changing upon a change in

Some philosophers challenge Burge's arguments for SE.[26] Our concern here, however, is with CI. CI has also been Egan's concern. Egan (1992, 1995) argues that Burge has confused intentional with computational descriptions and conflated methodological maxims with identity conditions. She agrees with Burge that the content of perceptual states is individuated with reference to distal physical stimuli (SE). But she rejects Burge's conclusion that these stimuli also play an individuative role in Marr's computational theories (CE). Egan argues that while Marr also describes computational processes intentionally – asserting, for example, that early visual processes compute the representations of salient properties of distal objects, such as their boundaries – it is crucial to separate this intentional description from the computational description of the processes. From a *computational* point of view, Marr explicitly asserts that early visual processes are described by the mathematical formula $\nabla^2 G * I(x,y)$, where ∇^2 is the Laplacian, G is the Gaussian, * is the convolution operator, and I(x,y) are the real-valued arguments of the function (Marr, 1982, pp. 336-338). And on Egan's view, this proves that CI is false.

Thus, Egan argues that Marr's distinction between the computational and algorithmic levels does not correspond to the standard distinction between the semantic (intentional) and formal (syntactic) levels. Rather, Marr's computational and algorithmic levels both provide formal descriptions of the system. The computational level provides a formal description of the function (input-output relations) computed, whereas the algorithmic level provides a formal description of the mediating process. On Egan's view, Marr separates the two levels for methodological reasons only. Marr's top-down approach rests on the assumption that external physical constraints make it more efficient, and perhaps essential, to characterize the input-output function first. Nevertheless, she argues, we must be careful not to confuse the

the physical environment. This thought experiment is discussed below.

[26] See Segal (1989, 1991), Shapiro (1993) and Butler (1998).

role of environment in discovering computational description with the formal nature of this description: “the top [computational] level should be understood to provide a *function-theoretic* [formal] characterization” (Egan, 1995, p. 185).

I cannot offer a complete analysis of Marr’s theory here, nor I can discuss in detail the interesting interpretations of this theory proffered by Burge, Egan and many others. I will, however, outline an alternative interpretation which lies somewhere between the views of Burge and Egan. On this interpretation, Egan is correct in attributing to Marr the view that the computational description of the visual system is, at least ideally, a formal one. She is certainly correct in claiming that Marr’s primary motivation is methodological, not individuative. And she is right to point out that Burge has at best shown that content (i.e., distal stimuli) plays an explanatory role in Marr’s computational theories, but has not demonstrated that content plays an individuative role in these theories. Despite this, it is possible to construct a better argument for Burge’s claim that content plays an individuative role in Marr’s computational theories.

Let us start with methodology. Marr’s principal methodological claim is that investigation of a visual system should proceed top-down: from the computational level, through the algorithmic level, to the implementation level. Marr often declares that “unless the computational theory of the process is correctly formulated, the algorithm will almost certainly be wrong” (1982, p. 124). This claim is, of course, controversial.[27] But our concern is not with the validity of the top-down strategy, but its motivation. One might think that the primary motivation of the top-down strategy is the inaccessibility of internal processes. But this is not Marr’s view. Marr explicitly states that “a*lthough algorithms and mechanisms are empirically more accessible*, it is the top level, the level of computational theory, which is

[27] See, for example, Churchland and Sejnowski (1992).

critically important from an information-processing point of view... an algorithm is likely to be understood more readily by understanding the nature of the problem being solved than by examining the mechanism (and the hardware) in which it is embodied" (my emphasis, p. 27). The problem that motivates the top-down strategy is that the physical mechanisms implement too many algorithms, and so there are many ways to abstract from hardware or netware. Even if all the implemented algorithms are transparent, scientists have no way of knowing which of the implemented algorithms is the appropriate computational description of the visual mechanism[28]. Trying to describe the visual process by means of neurons is like "trying to understand bird flight by studying only feathers: it just cannot be done. In order to understand bird flight, we have to understand aerodynamics" (Marr, p. 27). It cannot be done, that is, not because the structure of feathers is inaccessible, but because it is simply much easier to extract the structure relevant to flying when we have general knowledge of the structures that explain flying.

On my reading of Marr, then, the motivation for the top-down strategy is not that the mechanism is inaccessible. The motivation arises from the fact that the visual system, as a neural system, simultaneously implements a variety of algorithms. The methodological problem is to determine which of the implemented algorithms is the algorithm underlying the visual task. The idea behind the top-down strategy is that it is much easier to extract the underlying algorithm when we know something about the structures with which the visual system could solve the visual problem. The goal of Marr's computational theories is to utilize facts about the physical environment in order to arrive at the strategy of solving a visual problem. *Vision* demonstrates how scientists can employ their knowledge about the physical

[28] For Marr, a formal description of a process can consist of analog values – i.e., the intensity values I(x,y). This possibility dramatically multiplies the potential number of formal descriptions as we locate the cutoffs wherever we wish.

world to constrain possible solutions to visual problems. And Marr's remarkable achievement in *Vision* is to show that in many cases, such as in the theory of stereopsis, the physical world constrains unique solutions to visual problems. Indeed, it is this remarkable achievement that motivates Marr's top-down strategy. It is methodologically more sound, Marr contends, to arrive at the algorithm via environmental constraints than to attempt to extract it from neurons.[29]

This is as far as methodology goes, but it goes far enough to have implications for questions about individuation. For if the visual system, as a neural system, simultaneously implements different algorithmic structures, then we need a constraint to determine which of the implemented algorithms constitutes the computational structure of the system. But this constraint – as we saw throughout the argument in section 2 – is not to be found in the intrinsic physical features of the system. Rather, this constraint involves features of content. Hence content affects computational identity.

On the standard reading of Marr, then, the methodological and the individuative roles of content are separable. Both Burge and Egan assume that the methodological role of content is in solving a strictly *epistemological* problem. On this view, content is useful for *discovering* the underlying algorithm. But this methodological role of environmental facts has no implications for the question of whether content also plays an individuative role in computational theories (Burge) or not (Egan). On my reading, the methodological and individuative roles of content are interrelated, the latter driving the former. To reiterate, the individuative role of content is to determine which of the implemented formal structures is

[29] Marr (pp. 122-124) demonstrates, for example, that the algorithms for the stereo matching problem that were arrived at without the computational analysis of the stereo problem do not compute the right thing. It is also important to note that even though the computational theory has been established, there can still be different algorithms satisfying the (computational) constraints. Marr is perfectly aware of this possibility, as he himself mentions two different algorithms for stereo-matching which satisfy the same constraints (p. 27).

the computational structure of the visual system. The methodological problem is to reading this formal structure, and no other, off the implementing neural mechanism. Marr's theories exploit features of content to constrain the possible computational structure of the visual system. Specifically, Marr assumes that our visual system is a representational biological system that came into being through a process of evolution. As such, there must be some meaningful relations between physical facts about the world (e.g., illumination conditions, cohesiveness of matter, etc.) and facts about the visual system (e.g., the continuity constraint). And these semantic relations between the representing visual states and the represented distal physical facts are constraints on possible solutions of identifying the formal structure in question.

Thought experiments are another way to highlight the role of content in computational theories of vision. Thus Burge (1986, pp. 34-36) asks us to assume that the physical conditions have changed; for example, that the visual system is now located in a spiky universe that violates the continuity constraint. Assume that in this environment there are "different physical conditions and perhaps different (say optical) laws regularly causing the same non-intentionally, individualistically individuated physical regularities in the subject's eyes and nervous system" (p. 34). In this environment, Burge concludes, the computational identity of the visual system will be different too. But Burge's conclusion does not yet follows. As Egan argues (1995, p. 191), the computational identity of the visual system will remain the same even in a spiky universe as long as the function-theoretic relations are invariant across environments. More specifically, Burge and Egan agree that, in this example, the formal structure (input-output function and algorithm) underlying stereo-matching will remain the same in the spiky universe. They also agree that the visual content, being extrinsic, can change with the environment. They disagree only on whether this variable content does (Burge) or does not (Egan) enter into the computational identity of the system.

Now I tend to agree with Egan that, since the computational description is a formal description, on this scenario the computational identity of the visual system will not vary. However, there are *other* scenarios, which neither Burge nor Egan consider, in which the computational identity of the visual system does vary. It is at least conceptually possible that the formal structure (input-output function and/or algorithm) underlying what we see also varies. This can happen if, in the spiky environment, the proximal stimuli received by the visual system differ from those received in its usual environment. This may also happen in our world in the cases in where the visual system receives stimuli that generate optical illusions. My point is that our visual system can entertain many different stimuli. In its usual environment, it encounters only some of them. But in laboratories or other uncommon environments, it may indeed encounter other stimuli. In these environments, many features of the visual system can be different: the distal causal stimuli, the proximal stimuli and even conscious experience. But another feature can vary as well: the formal structure underlying the visual system. It might turn out that the syntactic structure underlying the visual task is different in a spiky universe; in this case, the computational identity of the visual system can could well vary too. Thus Burge has a point after all. The thought experiments, when properly construed, indeed show that a change in content across contexts sometimes alters the computational identity of the visual system. They show, in other words, that on the computational theories of vision, visual content affects computational identity.

I have argued that CI is true in the context of Marr's theories of vision. But what about the other questions asked about individuation: (1) Is SE true in the context of theories of vision? Is visual content extrinsic? (2) Is CE true in the context of vision? Do computational taxonomies of the visual system make essential reference to distal environmental facts? Neither question can be answered conclusively. It is true that Marr's computational theories employ distal physical facts to identify which of the implemented structures is the computational structure of the visual system. It is also true that vision theorists classify visual content by distal stimuli. But, as many have already pointed out, practice alone falls short of demonstrating that the individuation of content, and so of computational states, makes an essential reference to distal stimuli.[30] Yet, if visual content in Marr's theories is extrinsic, as both Burge and Egan assume, then Marr's computational theories of vision are also extrinsic. If visual content is extrinsic (SE), then computational theories of cognition are extrinsic too (CE).

4. Summary

In section 2 I put forward an argument for the thesis that content impacts the computational individuation of the states of cognitive systems. This argument undermines the central assumption of the computational theory of mind, namely, that cognitive processes, as computational processes, are non-intentional. Computational processes may, perhaps, be non-intentional in the sense that their formal descriptions do not explicitly mention specific content. However, content does determine – in ways discussed in section 3 – which of the implemented formal structures constitutes the system's computational structure. In this important sense, computational processes are intentional through and through.

This argument, if it is valid, calls for a reevaluation of the relations between computation and cognition. The task of the present paper, however, was more modest:

[30] For a recent argument to this effect see Butler (1998).

reexamination of the relationship between the claim that cognition is computation and claims about externalism in psychology. The most important implication of my argument is that upholding the claim that cognition is computation is no barrier to upholding the claim that computational theories of cognition are extrinsic (CE). Some have argued that computational processes, being described in formal terms, must be intrinsic. But, as I have attempted to show, thought experiments and Marr’s *Vision* provide adequate support for CI. Moreover, they provide adequate support for the claim that the computational identity of cognitive modules, and perhaps even of the whole visual system, can vary across contexts. The paper does not provide a complete argument for CE. If content is extrinsic (SE), however, CE follows immediately. But I did not argue for SE here.[31]

Department of Philosophy
The Hebrew University of Jerusalem,
Jerusalem, 91905
Israel

Oron Shagrir

References

Block, Ned: 1990. “Can the Mind Change the World?” In Boolos George (ed.), *Meaning and Method: Essays in Honor of Hilary Putnam*. Cambridge: Cambridge University Press. 137-170.

Bontly, Thomas: 1998. “Individualism and the Nature of Syntactic States”, *British Journal of Philosophy of Science* 49: 557-574.

Burge, Tyler: 1986. “Individualism and Psychology”. *Philosophical Review* 95: 3-45.

[31] Many colleagues contributed insightful comments. I am especially grateful to Hanoch Ben-Yami, Jack Copeland, Yuval Dolev, Frances Egan, Uri Hason, Yakir Levin, Itamar Pitowsky, Michael Roubach, Gila Sher and Steve Yalowitz. I am also grateful to anonymous referees of *Mind* for corrections and thoughtful suggestions. An early version of this paper was presented at the 1997 annual meeting of the Society for Philosophy and Psychology, and at the Philosophy Colloquia at the Hebrew University of Jerusalem, Ben Gurion University and Haifa University. I thank the participants for much stimulating discussion. This research was supported by The Israel Science Foundation founded by The Israel Academy of Sciences and Humanities.

Butler, Keith: 1998. “Content, Computation, and Individualism”. *Synthese* 114:277-292.

Chalmers, J. David: 1996. “Does a Rock Implement Every Finite-State Automaton?” *Synthese* 108: 309-333.

Churchland Patricia and Sejnowski Terrence: 1992. *The Computational Brain*. Cambridge, Mass.: MIT Press.

Copeland, B. Jack: 1996. “What is Computation?” *Synthese* 108: 335-359.

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

Davies, Martin: 1991. “Individualism and Perceptual Content”. *Mind* 100: 461-484.

Egan, Frances: 1992. “Individualism, Computation and Perceptual Content”. *Mind* 101: 443-459.

– 1995. “Computation and Content”, *Philosophical Review* 104: 181-204.

Fodor, Jerry: 1980. “Methodological Solipsism Considered as a Research Strategy in Cognitive Psychology”. *Behavioral and Brain Sciences* 3: 63-73.

– 1994. *The Elm and the Expert*. Cambridge, Mass.: MIT Press.

Haugeland, John: 1978. “The Nature and Plausibility of Cognitivism”. *Behavioral and Brain Sciences* 2: 215-226. Reprinted in Haugeland John (ed.): 1984. *Mind Design*. Cambridge, Mass.: MIT Press. 243-281.

Hildreth, Ellen and Ullman, Shimon: 1989. “The Computational Study of Vision”. In Posner Michael (ed.), *Foundations of Cognitive Science*. Cambridge, Mass.: MIT Press. 581-630.

Kitcher, Patricia: 1988. “Marr’s Computational Theory of Vision”. *Philosophy of Science* 55: 1-24.

Marr, David: 1982. *Vision.* San Francisco: Freeman.

Morton, Peter: 1993. “Supervenience and Computational Explanation in Vision Theory”. *Philosophy of Science* 60: 86-99.

Putnam Hilary: 1988. *Representations and Reality*. Cambridge, Mass.: MIT Press.

Searle, John: 1992. *The Rediscovery of the Mind*. Cambridge, Mass.: MIT Press.

Segal, Gabriel: 1989. “On Seeing What is Not There”. *Philosophical Review* 98: 189-214.

– 1991. “Defense of a Reasonable Individualism”. *Mind* 100: 485-493.

Shagrir, Oron: 1999. “What is Computer Science About?” *The Monist* 82:131-149.

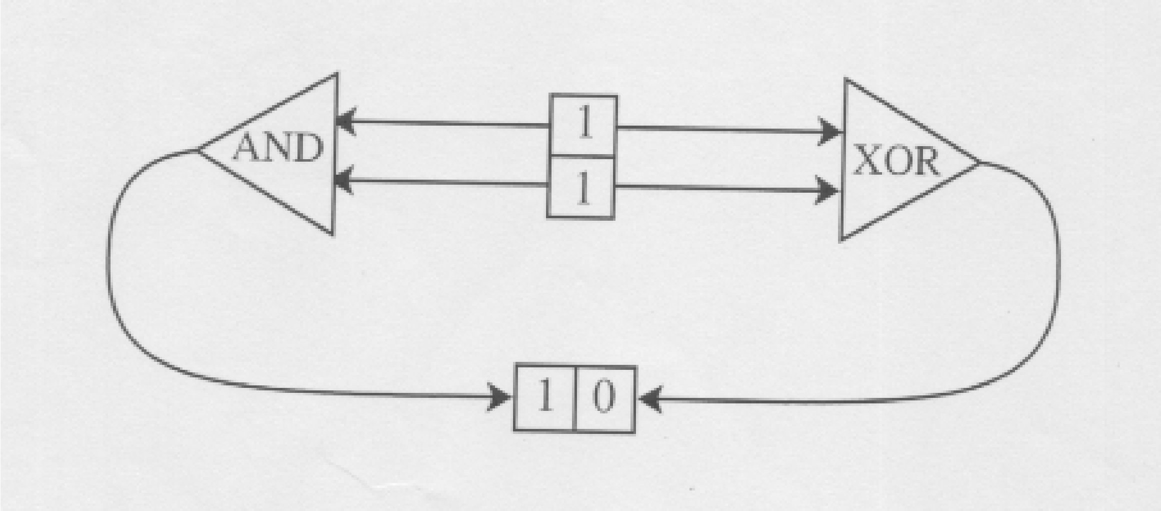
Shapiro, Lawrence: 1993. “Content, Kinds, and Individualism in Marr’s Theory of Vision”. *Philosophical Review* 102: 489-513.

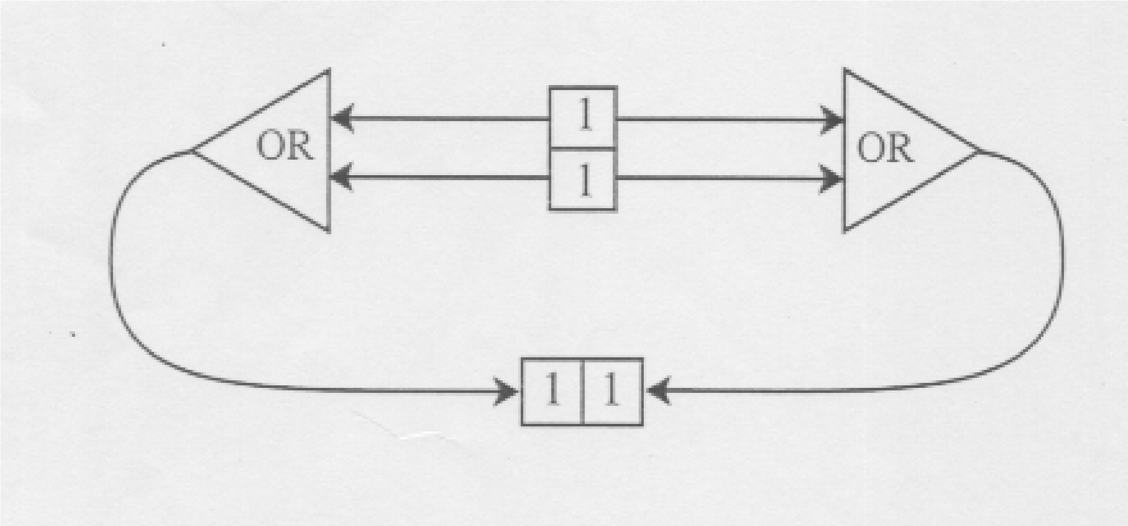
– 1997. “A Clearer Vision”. *Philosophy of Science* 64:131-153.

Sher, Gila: 1996. “Did Tarski Commit “Tarski’s Fallacy”?” *Journal of Symbolic Logic* 61: 653-686.

Stich, Stephen: 1983. *From Folk Psychology to Cognitive Science*. Cambridge, Mass.: MIT Press.

Wilson, A. Robert: 1994. “Wide Computationalism”. *Mind* 103: 351-372.


1
1
AND
XOR
1
0


1
1
OR
OR
1
1