



Towards Visual Semantics

Fausto Giunchiglia¹ · Luca Erculiani¹ · Andrea Passerini¹

Received: 22 March 2021 / Accepted: 24 August 2021 / Published online: 6 September 2021
© The Author(s) 2021

Abstract

Lexical Semantics is concerned with how words encode mental representations of the world, i.e., *concepts*. We call this type of concepts, *classification concepts*. In this paper, we focus on *Visual Semantics*, namely, on how humans build concepts representing what they perceive visually. We call this second type of concepts, *substance concepts*. As shown in the paper, these two types of concepts are different and, furthermore, the mapping between them is many-to-many. In this paper we provide a theory and an algorithm for how to build substance concepts which are in a one-to-one correspondence with classification concepts, thus paving the way to the seamless integration between natural language descriptions and visual perception. This work builds upon three main intuitions: (i) substance concepts are modeled as *visual objects*, namely, sequences of similar frames, as perceived in multiple *encounters*; (ii) substance concepts are organized into a *visual subsumption hierarchy* based on the notions of Genus and Differentia; (iii) the human feedback is exploited *not* to name objects, but, rather, to align the hierarchy of substance concepts with that of classification concepts. The learning algorithm is implemented for the base case of a hierarchy of depth two. The experiments, though preliminary, show that the algorithm manages to acquire the notions of Genus and Differentia with reasonable accuracy, this despite seeing a small number of examples and receiving supervision on a fraction of them.

Introduction

The Oxford Research Encyclopedia defines *Lexical Semantics* as the study of *word meanings*, i.e., *concepts* [25], where concepts are assumed to be constructed by humans through language. In the same line of thinking, this research focuses on *Visual Semantics*, namely, on how humans build concepts when using vision to perceive the world. The key assumption is that these two types of concepts are different and that, furthermore, they stand in a many-to-many relation (see Section “[Objects as Classification Concepts](#)” for the details).¹ Following the terminology from [18], we call the first type of concepts, *classification concepts*, and the latter type, *substance concepts*.² Our goal in this paper is to provide a theory and an algorithm for how to build substance concepts which are in a one-to-one correspondence with

classifications concepts, thus paving the way to the seamless integration between natural language descriptions and visual perception. Among other things, the solution we propose allows to deal with the so-called *Semantic Gap Problem* (SGP) [50]. The SGP, originally identified in 2010 and still largely unsolved, arises from the fact that, in general, there is a misalignment between what Computer Vision systems perceive from media and the words that humans use to describe the same sources. We articulate the problem we deal with is as follows.

Suppose that a person and a machine, e.g., a pair of smart glasses, are such that they see the same parts of the world under the same visual conditions. Suppose that the person has a full repertoire of words which allow her to describe what she sees according to her current point of view. Suppose, furthermore, that the machine starts from scratch

✉ Fausto Giunchiglia
fausto.giunchiglia@unitn.it

Luca Erculiani
luca.erculiani@unitn.it

Andrea Passerini
andrea.passerini@unitn.it

¹ University of Trento, Trento, Italy

¹ This assumption is consistent with the fact that the two activities of speaking and seeing involve different parts of the human brain [28].

² This terminology is motivated by the fundamentally different *function* that these concepts have. In fact, while the substance concepts are used to *represent substances* as they are perceived, the latter are used to *describe what is perceived*, i.e., *substance concepts*. This idea of seeing concepts as (biological) *functions* is based on the work in the field of *Teleosemantics*, sometimes called *Biosemanitics* [27], and in particular on the work by the philosopher R. Millikan [30, 32, 33, 35].

without any prior knowledge of the world and of how to name whatever it perceives. How can we build an algorithm which, by suitably asking the human, will learn how to recognize and name whatever it sees in the same way as its reference user?

A meaningful metaphor for this problem is that of a mother who is teaching her baby child how to name things using her own words in her own spoken language. The work in [18] provides an extensive description of the complications related to this problem, mainly related to the many-to-many relation existing between substance and classification concepts. Further complications come from the fact that, based on the definition above, the learning algorithm needs to satisfy the following further requirements:

- it must be generic, in that it should make no assumptions about the input objects;
- it must learn new objects never seen before as well as novel features, never seen before, of previously seen objects;
- it must learn from a small number of examples, starting from no examples.

The proposed Knowledge Representation (KR) solution is articulated in terms of a set of novel definitions of some basic notions, most importantly that of *object*. The theory proceeds as follows.

- We model *objects* as *substance concepts*, that we model as sets of *visual objects*, i.e., sequences of similar frames, as perceived in multiple events called *encounters*. Visual objects are stored in a *cumulative memory* \mathcal{M} of all the times they were previously perceived.
- Substance concepts are organized into a (*visual*) *subsumption hierarchy* which is learned based on the notions of *Genus* and *Differentia*. These two notions *mutatis mutandis*, replicate the notions with the same name that, in *Lexical Semantics*, are used to build *subsumption hierarchies* of word meanings [17, 29].
- The visual hierarchy is learned autonomously by the algorithm; the user feedback makes sure that the hierarchy built by the machine matches her own linguistic organization of objects. In other words, the user feedback is the means by which the hierarchy of substance concepts is transformed into a hierarchy of classification concepts. The key observation here is that the user feedback is provided not in terms of object names, as it is usually the case, but in terms of the two properties of *Genus* and *Differentia*.

The paper is organized as follows. First, we introduce objects as classification concepts, as they are used in natural language and organized in Lexical Semantics hierarchies

(Section “[Objects as Classification Concepts](#)”). This section provides also an analysis of why the very definition of classification concepts makes them unsuitable for visual object recognition. Then we define substance concepts as sets of visual objects (Section “[Objects as Substance Concepts](#)”). Then, in Section “[Building Substance Concepts](#)”, we provide the main algorithm by which substance concepts are built, while, in Section “[Object Subsumption and Identity](#)”, we describe how a hierarchy of substance concepts is built which is aligned with that of classification concepts. In this section we also provide the two basic notions of *Genus* and *Differentia* which are used to build the hierarchy. The algorithm for object learning is described in Section “[The Learning Algorithm](#)”. This algorithm has been developed for the base case of hierarchies of depth two. The extension to hierarchies of any level is left to the future work. The algorithm is evaluated in Section “[Experiments](#)”. Finally, the paper ends with the related work (Section “[Related Work](#)”) and the conclusions (Section “[conclusions](#)”).

Objects as Classification Concepts

Objects are usually named using nouns. In Lexical Semantics the meaning of nouns is provided via intensional definitions articulated in terms of *Genus* and *Differentia* [29], following an approach first introduced by Aristotle [39]. Let us consider for instance the following two definitions:

- a *triangle* is a *plane figure* with *three straight bounding sides*;
- a *quadrilateral* is a *plane figure* with *four straight bounding sides*.

In these two definitions we can identify three main components:

- *Genus*: some previously defined set of properties which is shared across distinct objects, e.g., the property of *being a plane figure*;
- *genusObj* (also called *genusObj object*): a certain representative object which satisfies the *Genus* property, e.g., the object *plane figure*. The set of objects satisfying the *Genus* properties are said to have that (same) *genusObj*;
- *Differentia*: A selected novel set of properties, different from the *Genus* properties, which are used to differentiate among objects with the same *genusObj*, e.g., the properties *having three straight bounding sides* and *having fours straight bounding sides*. These two properties define, respectively, triangles and quadrilaterals as distinct objects with the same *genusObj*.

Genus and Differentia satisfy the following four constraints:

- *Role 1 of Genus*: if two objects have different genusObj , then they are (said to be) *different*. For instance, a pyramid is not a plane figure and, therefore, is different from a triangle.
- *Role 2 of Genus*: The viceversa of Role 1 is not true, namely, we may have different objects with the same genusObj . For instance, a quadrilateral and a triangle are both plane figures but they are not the same object.
- *Role 1 of Differentia*: Two objects with the same genusObj , but different from the genusObj , are (said to be) the *same* object if and only if the *Differentia* properties do not hold of the two objects. Thus, for instance, two objects with the same genusObj and with a different *Differentia*, e.g., a triangle and a quadrilateral, are different despite being both a plane figure. Dually, two objects with the same genusObj and the same *Differentia*, e.g., two triangles, are the same object (relatively to the current selection of Genus and Differentia).
- *Role 2 of Differentia*: a genusObj and an object with that genusObj are different when the latter is characterized by a set of properties, i.e., its *Differentia*, that the genusObj does not have. Thus for instance a triangle is not the same as a plane figure, as it is just one of the many possible plane figures, e.g., triangles, quadrilaterals which share the same genusObj .

A first observation about the definitions above is that, when we say that two objects are the same object, we only mean that they satisfy the same Genus and the same Differentia. It does not necessarily mean that they are two occurrences of the same object. Thus, for instance, a right triangle and an equilateral triangle are considered as being the same object, when compared with quadrilaterals, in that they have the same number of sides. At the same time they are considered as different objects when the Differentia is taken to be the size of their angles. This observation has two immediate consequences. The first is that the process above can be iterated at any level of detail, thus creating hierarchies of any level of depth. It is a fact that, in lexical semantics, the meaning of nouns is organized as a hierarchy of increasing specificity, each layer being characterized by a new Genus and a new Differentia. In this hierarchy, an object with a certain Genus is a child of its genusObj . As a consequence, a hierarchy of depth n can be seen as the recursive juxtaposition of $(n - 1)$ hierarchies of depth 2, where the genusObj of the depth 2 hierarchy one level down is one of the children of the genusObj one level above. The root of this hierarchy is usually called *thing* [17, 29]. The second is that this process of progressive differentiation allows

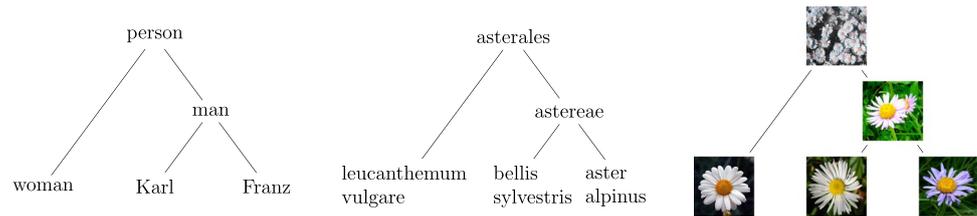
to split the set of objects under consideration into progressively smaller and smaller sets, based on the selected set of properties.

A second observation is that the above definitions are given in natural language and are meant to make precise the meaning of words. These linguistic definitions are designed to generate what we call *classification concepts*, namely, concepts which are amenable for classification [13]. In addition, in fact, the very existence of lexical semantics hierarchies provides evidence of their suitability for this task. This type of definitions is well grounded in the everyday practice, in particular when used to name and describe things, for instance during interactions among humans. However, they do not work as well while one is in the middle of the recognition process, namely, while she is trying to identify the object she is looking at. How many times were you able to recognize someone or something based only on a natural language description, without the help of a photo or anything which could point to specific spatial properties?

Let us clarify this observation with an example. Assume you see at a certain distance two *things* moving towards you. Initially you will not recognize what these things are but, when they are close enough, you will be able to recognize two *persons*, seen from the back. The day after, you see again two persons, which may or may not be those recognized the day before: hard to say, they did not come close enough. In any case, this second time these two persons get close enough for you to finally recognize your friends *Karl* and *Frank*. What allowed you to distinguish *Karl* from *Frank* is that the former has white hair while the latter has black hair and mustaches. Later on, walking towards you, you will recognize a *woman*. You will have been able to recognize her as a *person* different from the two previous *men* because she has long hair and a skirt. Of course you will know the terms you have used to describe what you will have seen, i.e., *person*, *man*, *woman*, *Karl* and *Frank*, as someone will have taught them to you, for instance during your childhood. In KR, the simple scene described above can be formalized by saying that *Karl* and *Frank* are *instances*, while *person*, *man* and *woman* are (*classification*) *concepts* and by stating the following facts: *man(Karl)* (to be read as *Karl is a man*), *man(Frank)*, $\text{man} \sqsubseteq \text{person}$ (to be read as *man is subsumed by person*) and $\text{woman} \sqsubseteq \text{person}$, the latter two facts stating that all men and all women are persons. The resulting hierarchy, as formally defined via the logical subsumption symbol \sqsubseteq , is provided in Fig. 1 (first left) where the classification concepts there represented are defined, for instance, as (partial quote from [29])

- *person*: individual, someone, somebody;
- *woman*: an adult female person, as opposed to man;
- *man*: an adult male person, as opposed to woman;
- *Karl*: an instance of a man;

Fig. 1 (Left): a classification concept hierarchy; (Center): a classification concept hierarchy for daisies; (Right:) the center hierarchy where words are substituted with images representing the corresponding daisies



- *Franz*: an instance of a man.

Notice how the above definitions and the properties they involve (e.g., being adult, male or female, being an instance) are completely unrelated to the process by which recognition was carried out, which was in terms of a continual analysis of visual information, at increasing levels of precision.

The previous example is representative of the situation where the observer has complete knowledge of the objects being perceived and the partiality of information is caused by some contextual factors. Consider now the hierarchy of classification concepts in the center of Fig. 1, which names and classifies daisies, whose images are in the corresponding place in the hierarchy in the right of Fig. 1. A possible lexical semantics definition of these daisies is as follows:

- *asterales*: an order of flowering plants containing eleven families, the most notable being asteraceae (known for composite flowers made of florets);
- *leucanthemum vulgare*: flower native to Europe and the temperate regions of Asia, commonly referred as marguerite;
- *astereae*: a tribe of plants, commonly found in temperate regions of the world, also called daisy or sunflower family;
- *bellis sylvestris*: Southern daisy, perennial plant native to central and northern Europe;
- *aster alpinus*: blue alpine daisy, plant commonly found in the mountains in Europe.

Most readers, in particular those who are not florists, even if coming to know about the hierarchy above, e.g., because being described it, will be unable to recognize the various types of daisy. As a consequence they will not be able to build it starting from images (e.g., the ones on the right in Fig. 1), simply because they will not be able to recognize the features which allow to distinguish among the various types of daisy. Most likely, in many cases, the hierarchy will be collapsed to a single node while, in others, the light purple daisy will be separated from the others, just because of its colour.

In general, classification concepts do not seem well suited for the process of object recognition. This despite the fact that it is common practice to use them in supervised machine learning, where the user feedback is, often if not always,

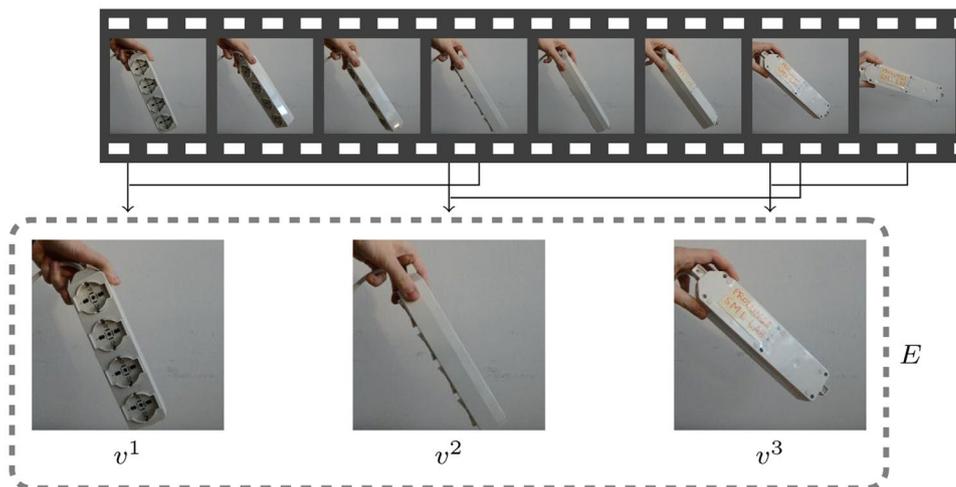
provided in terms of words whose meaning is defined via lexical semantics hierarchies. Evidence of this difficulty is provided by the SGP, whose original formulation describes it as (quote from [50]) “... *the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation.*”. The main motivation for the SGP seems to be that classification concepts model objects as *endurants*, i.e., as being always wholly present, at any given moment in time, with their proper parts being present in a certain spatial configuration and satisfying certain properties (e.g., color, shape, position, activity) [14]. Typical examples of endurants are all the physical objects, e.g., those mentioned above. Of course, the spatial configuration may change, or the object might not be accessible visually (as in the first example above), or the observer might not be able to discriminate some of its relevant properties (as in the daisies example above), but this has no impact on how classification concepts are defined.

Classification concepts, while serving well the purpose of describing what was previously perceived, are largely unrelated to the process by which the objects are perceived and, in particular, to the fact that their perception is constructed *incrementally*, via a set of *partial views* which progressively enrich what is visually known. To this extent, notice how *person*, *man*, and *Karl* are correctly represented in Fig. 1 as three different classification concepts. However, in the little story above, these three classification concepts actually describe the same piece of reality, seen at different times, at different levels of detail and from different points of view.

Objects as Substance Concepts

The key intuition underlying the work described here is to model objects as *perdurants*, where, quoting from [14] ... *perdurants ... just extend in time by accumulating different temporal parts, so that, at any time they are present, they are only partially present, in the sense that some of their proper temporal parts (e.g., their previous or future phases) may be not present.* Typical examples of perdurants are events and activities. Taking an object as a perdurant amounts to saying that we never have a full (visual) picture of the object but that its visual representation is built progressively, in time. Notice how this is exactly what happens in our everyday life.

Fig. 2 Example of an encounter. For better visualization, each visual object is represented, here and later, as its first frame



We call *substance concepts* the representation of objects as perdurants.

The starting point in the definition of substance concepts is the crucial distinction between what we perceive as being in the real world, that we call *substances* and their corresponding mental representations, i.e., their *substance concepts*. Following R. Millikan, we take substances as those things (quote from [33]), “... about which you can learn from one encounter something of what to expect on other encounters, where this is no accident but the result of a real connection ...”. [18] provides a detailed discussion of what substances are and of how they generate substance concepts in the mind of observers, based on the work on *Teleosemantics* [27], and in particular on the work by Ruth Millikan [29–34]. In the following, substances should be intuitively thought as those things which, when perceived in the most detail, will generate the perception of individuals, e.g., *Karl*, *my cat*, but that, under different conditions, will generate more generic or even very different substance concepts, e.g., *a moving object*, *an animal*. The key observation is that, while substances are crucial in our informal understanding of perception in that they allow us to focus on the process of how objects are perceived, they play no role in the formal model that we define below. With this in mind in the following: (i) we avoid defining what a substance is (no such definition could be meaningfully grounded in human experience); and (ii) we consider substances only in their *causal role* on the generation of a concept, a role that is constrained within the events during which a substance is perceived. We call such events, *encounters* and (iii) we qualify this causal role in terms of two properties that substances have, as introduced below, and that we call *Space Persistency* and *Time Persistency*. Notice, however, that both *Space Persistency* and *Time Persistency*, as all the definitions provided in this paper, are given as properties of substance concepts.

We assume that encounters are represented as *spatio-temporal worms*, i.e., temporal sequences of *frames*, where f_S^i is a frame for a substance S , each frame being encoded via a set of *low-level visual features*.³ We represent encounters, by exploiting the *Space Persistency* of substances, namely, the fact that, in time, substances change very slowly their spatial position. Because of space persistency, during an encounter, any two adjacent frames will be very similar, while this will not necessarily be the case with two non adjacent frames. We model *Space Persistency* in terms of *Frame Similarity (Dissimilarity)*, written $f_{S_1} \simeq f_{S_2}$ ($f_{S_1} \not\simeq f_{S_2}$). Given *Frame Similarity*, we define *visual objects*, where v_S is a visual object for a substance S , as sequences of adjacent frames where the last frame is *similar* to the first, and *encounters* E_S as sets of visual objects, i.e.,:

$$E_S = \{v_S^1, \dots, v_S^n\}. \tag{1}$$

Figure 2 reports an example of an encounter consisting of eight frames organized in three visual objects. Notice how having multiple similar frames in the same visual object makes it quite robust to local contextual variations. The first time a substance S is perceived as a new *object*, that object will consist of a single encounter; but this object will be enriched by subsequent encounters. We model this situation by taking objects to be the set of all the different visual objects collected by the different encounters. Let E_S^1, \dots, E_S^m be a set of encounters. Then we have:

³ Notationally, we use superscripts to mean elements of a sequence, and (optionally) the subscript S , to mean elements obtained from one or more encounters E_S with the substance S , as in f_S^i , v_S^i , and O_S^i . Different subscripts mean elements generated in possibly different sets of encounters. We omit the subscript whenever the substance we are referring to is clear from the context.

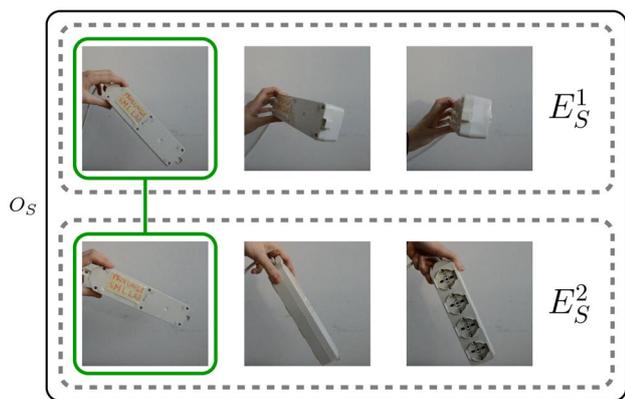


Fig. 3 Single object consisting of two encounters. The green line connects two similar visual objects

$$O_S = \cup_i E_S^i = \{v_S^1, \dots, v_S^n\} = \{v_S^i\}. \tag{2}$$

This situation is well represented in Figure 3 where each row is a different encounter.

Building Substance Concepts

Objects as substance concepts get cumulatively built in time. Let E_S^1, \dots, E_S^m be a sequence of encounters. Let O_S be an object defined as in Eq. (2). Then O_S is incrementally constructed as follows:

1. ADDOBJECT(\mathcal{M}, E_S^1)
2. UPDATEOBJECT($\mathcal{M}, O_S^i, O_S^{i-1} \cup E_S^i$), $i = 2, \dots$ (3)
3. O_S IS O_S^i , $i = 1, \dots$

where:

- ADDOBJECT creates a new object O_S^1 in the *cumulative memory* \mathcal{M} of the objects perceived so far;
- O_S^i is an object as perceived after any given number i of encounters; and
- $O_S^{i-1} \cup E_S^i \in \mathcal{M}$ is the cumulative memory of O_S^i ;
- UPDATEOBJECT updates the current memory O_S^{i-1} of an object with the visual objects coming from E_S^i , thus constructing O_S^i ;
- The construct IS in Item 3 is the formal statement assessing that we take objects as the cumulative memory of what has been perceived so far.

A first observation is that item 3 implicitly states that substance concepts evolve in time, i.e., that they are perdurants. In this perspective, O_S^{i-1}, O_S^i, E_S^i and also O_S , are all partial views of S , all contributing to the construction of O_S . This process of object construction may eventually terminate if

the appearance of an object does not change. However, an object may also keep evolving. Thus, for instance, the current encounter with *Frank* may contain visual objects which are quite dissimilar from the ones encountered earlier on, for instance because of the different age (e.g., 15 vs. 35).

A second observation is that the process described in Eq. (3), and in particular the decision of which between step 1 and step 2 must be applied, depends on the ability to recognize whether the current encounter is a partial view of an object already recognized. However, how to decide? Let us write $O_{S_1} = O_{S_2}$ to mean *Object Identity*, namely, that the two substance concepts are two (partial) views of the *same* object, rather than two views of two *different* objects. This may, in fact, be the result of two different sequences of encounters with the same object. Let us also write $O_{S_1} \neq O_{S_2}$ to mean *Object Diversity*. Then, Item 2 will be applied only for that object O_S such that $O_S^{i-1} = E_S^i$, while Item 1 will be applied whenever $O_S^{i-1} \neq E_S^i$ for all objects in \mathcal{M} .

The complications arising in the decision on Object Identity depend on two main factors. The first is that the *correlation between substances and substance concepts is many-to-many*.⁴ To reiterate an example from the previous section, the same substance can be perceived as *Karl*, as a *man* or as a *person* while, vice versa, the same substance concept, e.g., *man* can be recognized from multiple individuals. In other words, we need to decide at which level, in the visual subsumption hierarchy, the current encounter for the same substance should be assigned. The second issue is that, independently of the level of the subsumption hierarchy, the decision on Object Identity must be made taking objects to be *endurants*, as represented by classification concepts, being classification concepts what is used by humans in their everyday interaction and classification activities. Object Identity is a much richer notion than visual similarity as it involves considerations, such as language, culture, function of the objects, and much more, see, e.g., [20, 22, 36]. Among other things notice how we have $O_S = O_S^i, i = 1, \dots$, this meaning that Object Identity is invariant in time. As a consequence, there is a *many-to-many correspondence between substance concepts and classification concepts*, as also extensively exemplified in [18].

The double many-to-many mapping from substances to substance concepts and then from substance concepts to classification concepts is the main cause of the inherent ambiguity which appears in the identification of objects. This phenomenon is well known in Computational Linguistics and it is the cause of the so-called *lexical gaps*, namely, concepts which are lexicalized in one language but not in other languages [16]. Things are made even worse when,

⁴ This corrects the wrong statement, made in [18], that there is one-to-one mapping between substances and substance concepts

even within the same language, one considers the subjective behaviour of individuals; see the two examples in Section “[Objects as Classification Concepts](#)”. Notice that the problem is not that of constructing a hierarchy of meanings; in Section “[Object Subsumption and Identity](#)” we show how this can be done based on the visual similarity of objects as defined as in Eq. (2). The problem is that such a hierarchy will almost inevitably suffer from the SGP and, therefore, will not achieve the goal of aligning classification concepts and substance concepts. The solution we propose is articulated in the following main assumptions:

1. We assume that the fact that two objects are visually similar is a *necessary condition* for object identity. This assumption is well grounded in our everyday experience and also made in the mainstream Computer Vision research. To this extent, we introduce the notions of *Visual Object Similarity (Dissimilarity)*, written $v_{S_1} \simeq v_{S_2}$ ($v_{S_1} \not\simeq v_{S_2}$) and of *Object Similarity (Dissimilarity)*, written $O_{S_1} \simeq O_{S_2}$ ($O_{S_1} \not\simeq O_{S_2}$). Notice that we need to define what visual similarity is, given that, as discussed above, the same object can appear in many different ways; this will be discussed in Section “[The Learning Algorithm](#)”.
2. We assume, as also implicit in Millikan’s quote, that substances have a property of *Time Persistency*, namely, some form of time invariance in how they appear across encounters. This assumption allows us to compare, up to a point, visual objects coming from different encounters. Notice that how space and time persistency operate is specific to the objects being considered, no matter whether instances or concepts. Thus, for instance, *Karl* will keep having white hair while *Frank* will keep having black hair and mustaches. Analogously, humans, such as all animal species, are characterized by a *homeostatic mechanism* which causes them to possess a certain set of common traits (e.g., their shape, how they move) that often, but not always, make them look similar [18]. The key consideration here is that, once an observer has subjectively decided what is the object that she is trying to recognize from a substance S , the criteria for object identity do not change. In other words, time persistency applies not only to the perceived object but also to the perceiving subject.
3. We organize objects in a visual subsumption hierarchy, exactly like the one used in lexical semantics, but with the key difference that *Genus* and *Differentia* are computed in terms of the substance concepts’ visual properties, as represented by the visual objects. This allows to deal with the problem of the many-to-many mapping between substances and substance concepts.
4. *Last, but not least*, we deal with the many-to-many mapping between substance concepts and classification concepts by relying on the key role of the *user supervi-*

sion. This transformation is crucial to the integration of human vision, where objects keep evolving in time (i.e., they are perdurants), and language-based reasoning, which thinks of objects as being completely described in any moment in time (i.e., they are endurants). *Genus* and *Differentia* can be computed in a completely unsupervised manner, via object similarity. However, the user feedback, which is given *only* on *Genus* and *Differentia*, guarantees that the machine-built hierarchies largely coincide, modulo recognition mistakes, with the hierarchies that a user would build. Notice how this supervision is unavoidable, that it is exactly the same type of supervision that a mother would give to her child, and that it is subjective, evidence being also that different languages conceptualize different objects [16].

As a last remark, notice that in the visual hierarchy mentioned in item 3, all nodes are labeled only by substance concepts. Instead, in lexical semantics hierarchies, nodes are labeled by (classification) concepts, e.g., *man*, and instances, e.g., *Frank*. In other words, as correctly pointed out by R. Millikan [33], but see also [18], from a perception point of view, the usual KR distinction between concepts (usually modelled as sets of instances) and instances does not apply.

Object Subsumption and Identity

As from Eq. (2), objects, represented as substance concepts, are sets of visual objects. The idea is to exploit this fact to build a hierarchy of objects based on visual similarity. As from the discussion at the end of the previous section, this hierarchy gives us only the necessary conditions for object identity. In the following we assume that the two hierarchies of substance concepts and classification concepts coincide assuming that the user feedback is used to validate the choices made. The algorithm in Section “[The Learning Algorithm](#)” will show how this is done in practice by suitably asking feedback to the user.

As from Section “[Objects as Classification Concepts](#)”, a lexical semantics hierarchy can be seen as the iteration of many depth 2 hierarchies, each with its own *Genus* and *Differentia*. Therefore, without lack of generality, in the following we focus on hierarchies of depth 2. The main goal below is to restate the conditions for *Genus* and *Differentia*, informally stated in Section “[Objects as Classification Concepts](#)” for classification concepts, in terms of formally defined conditions on substance concepts. Let us assume that we are given a genus object `genusObj`. In the general case the construction of `genusObj` will happen recursively from the top node, i.e., *thing*. Then, let us define `sameGenus(O_{S_1}, O_{S_2})` and `Different(O_{S_1}, O_{S_2})` as two

binary boolean functions which discriminate over objects, based on their visual objects. We enforce the four roles in Section “Objects as Classification Concepts” by enforcing the following three constraints:

$$\neg \text{sameGenus}(O_{S_1}, O_{S_2}) \longrightarrow O_{S_1} \neq O_{S_2}, \tag{4}$$

$$\begin{aligned} & \text{sameGenus}(O_{S_1}, O_{S_2}) \longrightarrow \\ & (O_{S_1} = O_{S_2} \longleftrightarrow \neg \text{Different}(O_{S_1}, O_{S_2})), \end{aligned} \tag{5}$$

$$O_{S_G} \subseteq O_{S_1} \cap O_{S_2}. \tag{6}$$

where O_{S_G} is the `genusObj` of O_{S_1}, O_{S_2} . Notice how the specifics of `Genus` and `Differentia` are left open, we only require that they are both computed out of the visual objects in input, i.e., O_{S_1}, O_{S_2} and that they satisfy the three constraints above. This is on purpose as it gives us freedom in many dimensions, e.g., of the specifics of the learning algorithms used, of how visual similarity and/or object identity are defined, and also of how `sameGenus` and `Different` are defined in any different layer of the hierarchy under construction. The algorithm in Section “The Learning Algorithm” will instantiate the missing information selecting, for each decision point, one among the many possible options.

Let us concentrate on the constraints. They satisfy the following intuitions. *First*, they satisfy the four criteria defined in Section “Objects as Classification Concepts”. Equation (4) formalizes *Role 1* and *Role 2* of `Genus` while Equation (5) formalizes *Role 1* of `Differentia`. Equation (6) formalizes *Role 2* of `Differentia`; in fact from Eq. (6) we have $O_{S_G} \subseteq O_{S_1}$. To have $O_{S_G} \neq O_{S_1}$, O_{S_1} must have at least a visual object $v_{S_1}^i \notin O_{S_G}$. Then there are two cases, either $v_{S_1}^i$ is such that `Different` holds, in which case we are done (from Eq. (5)), or this is not the case, in which case $O_{S_G} = O_{S_1}$, namely, that visual object is irrelevant to the identity of O_{S_1} . Notice how this latter case does not rise if we take `genusObj` to be exactly the intersection. Equation (6) captures the intuition that the visual objects which are not considered belong to both objects by chance. Thus for instance, *Karl* and *Frank* might happen to have had, when observed, a red sweater. However, red sweaters are not a characteristic of *men*. *Second*, Eq. (4) captures the fact that `sameGenus` provides necessary but not sufficient conditions for object identity. *Third*, Eq. (5) provides necessary and sufficient conditions for two objects to be different, but under the assumption that `sameGenus` holds. Namely, `Different` can be applied only after having discarded all the objects which do not satisfy `sameGenus`.

The three constraints above allow us to build the desired subsumption hierarchy. Let us write $O_{S_j} \sqsubseteq O_{S_i}$ ($O_{S_i} \sqsupseteq O_{S_j}$) and say that O_{S_j} is *subsumed* by O_{S_i} (O_{S_i} *subsumes* O_{S_j}) to mean that the visual objects of O_{S_j} are a subset of those vis-

ual objects of O_{S_i} which are relevant for the computation of `Different` (see discussion above on Eq. (6)). We also write $O_{S_j} \sqsubset O_{S_i}$ and talk of *strict* subset and subsumption to mean $O_{S_j} \sqsubseteq O_{S_i}$ and $O_{S_j} \neq O_{S_i}$, and similarly for $O_{S_i} \sqsupset O_{S_j}$.

Let us assume that O_{S_2} and O_{S_2} have the same `genusObj`, O_{S_G} , namely, that `sameGenus`(O_{S_1}, O_{S_2}) and, therefore, `sameGenus`(O_{S_G}, O_{S_2}), `sameGenus`(O_{S_G}, O_{S_1}) hold. Clearly, $O_{S_G} \sqsubseteq O_{S_1}$ and $O_{S_G} \sqsubseteq O_{S_2}$. This makes the premise and, therefore, the consequence of Eq. (5) hold of all three objects. We have the following cases (for compactness, below we write `D` to mean `Different`):

1. $\text{D}(O_{S_1}, O_{S_2}), \text{D}(O_{S_1}, O_{S_G})$ and $\text{D}(O_{S_2}, O_{S_G})$: we have $O_{S_G} \sqsubset O_{S_1}$ and $O_{S_G} \sqsubset O_{S_2}$, namely, the situation where all three objects are different;
2. $\text{D}(O_{S_1}, O_{S_2}), \neg \text{D}(O_{S_1}, O_{S_G})$ and $\text{D}(O_{S_2}, O_{S_G})$: we have $O_{S_G} = O_{S_1}$ and $O_{S_G} \sqsubset O_{S_2}$;
3. $\text{D}(O_{S_1}, O_{S_2}), \text{D}(O_{S_1}, O_{S_G})$ and $\neg \text{D}(O_{S_2}, O_{S_G})$: we have $O_{S_G} = O_{S_2}$ and $O_{S_G} \sqsubset O_{S_1}$;
4. $\neg \text{D}(O_{S_1}, O_{S_2}), \text{D}(O_{S_1}, O_{S_G})$: we have $O_{S_G} \sqsubset O_{S_1}$ with $O_{S_1} = O_{S_2}$;
5. $\neg \text{D}(O_{S_1}, O_{S_2}), \neg \text{D}(O_{S_1}, O_{S_G})$: we have $O_{S_1} = O_{S_2} = O_{S_G}$.

Two observations. The first is that, under the assumption that O_{S_1} and O_{S_2} have the same `genusObj` O_{S_G} , `sameGenus` and `Different` provide us with *necessary* and *sufficient* conditions for both *object identity* and *object subsumption*, and therefore, they provide us with the means for building the depth 2 sub-hierarchy under consideration. In fact as from the (only if) directions of clauses 2, 3, 4, 5, two objects are the same if they have the same `Genus` and `Different` does not hold of them. Thus, taking into account the necessary conditions provided by Eq. (4) we have:

$$\begin{aligned} O_{S_1} = O_{S_2} \longleftrightarrow & \text{sameGenus}(O_{S_1}, O_{S_2}) \wedge \\ & \neg \text{Different}(O_{S_1}, O_{S_2}) \end{aligned} \tag{7}$$

Furthermore, as from clauses 1, 2, 3, 4, we have that `genusObj` is the parent node of the objects of which it is the `genusObj`, namely:

$$O_{S_1} \sqsubset O_{S_G} \longleftrightarrow \text{Different}(O_{S_1}, O_{S_G}) \tag{8}$$

The concluding remark is that, so far, we have only dealt with hierarchies of depth two, but the reasoning above can be replicated to build hierarchies of any depth. Let us assume that we have a new object O_{S_3} with $\neg \text{sameGenus}(O_{S_3}, O_{S_1})$ and, thus, $O_{S_1} \neq O_{S_3}$, $O_{S_2} \neq O_{S_3}$, and $O_{S_G} \neq O_{S_3}$. At the same time, O_{S_3} can share some visual objects with O_{S_1} or O_{S_2} which make `Different` false. Thus, for instance, a *plane* is not a *bird*, but they both *fly*. Given, any two objects there is *always* a `genusObj`, also when these objects are very different, and this is the key fact which allows for the construction

of subsumption hierarchies of any depth. Notice how we may end up with a `genusObj` which is the empty set, this being the limit case where the `genusObj` is *thing*: a generic object is something which has been perceived but with no associated visual objects.

The Learning Algorithm

We first provide a computational interpretation of the definitions introduced above and then we introduce the algorithm, which should be seen as a first prototype and representative of a wide class of algorithms. Any algorithm will do as long as it satisfies the constraints for `Genus`, `Differentia` and the `genusObj`. Let us analyze the definitions one by one.

Frames. We encode frames using an *unsupervised* deep neural network [5], trained to perform a combination of self-supervised and clustering tasks. Using an unsupervised network allows to produce embeddings which are not explicitly biased towards classes of objects, while, at the same time, complying to the assumption that machines extract features from what they perceive, autonomously, as humans do. We define *frame similarity* as the Euclidean distance between frame encodings.

Visual objects. We define them as contiguous sequences of frames, and we represent them as the average between the frame encodings. We assume for robustness that visual objects are of a fixed limited length. Visual object are perceived by a procedure, named `PERCEIVE`, which returns an encounter as a set of visual objects, as from Eq. (1). We model *visual object similarity* as a diversity threshold on the distance between visual objects:

$$v^i \simeq v^j \stackrel{\text{def}}{=} d(v^i, v^j) < \theta \tag{9}$$

Objects. We define objects as from Eq. (2), i.e., as sets of visual objects extracted from sets of encounters. We define *object similarity*, analogously to visual object similarity, as a diversity threshold on the distance between objects:

$$O_{S_1} \simeq O_{S_2} \stackrel{\text{def}}{=} d(O_{S_1}, O_{S_2}) < \theta \tag{10}$$

where the distance between objects is defined as the minimal distance between their respective visual objects:

$$d(O_{S_1}, O_{S_2}) = \min_{v^i \in O_{S_1}} \min_{v^j \in O_{S_2}} d(v^i, v^j) \tag{11}$$

By keeping the same threshold as for visual object similarity, we have that object similarity holds when two objects have at least two similar visual objects:

$$O_{S_1} \simeq O_{S_2} \iff \exists v^i \in O_{S_1}, \exists v^j \in O_{S_2} : v^i \simeq v^j. \tag{12}$$

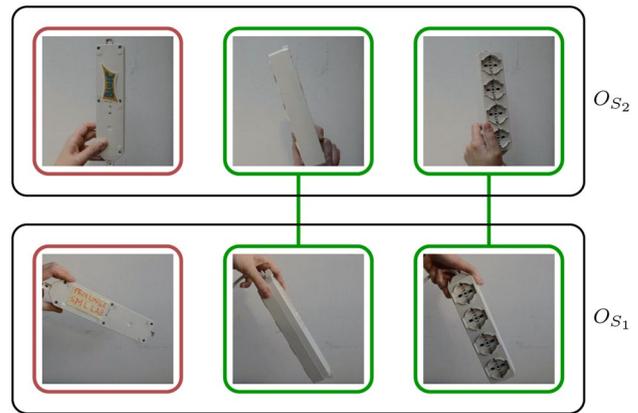


Fig. 4 Two distinct objects which share the same `genusObj`. The green lines connect similar visual objects, while the visual objects representing the `Differentia` are highlighted in red

Genus. We implement `sameGenus` as a Boolean function which computes object similarity:

$$\text{sameGenus}(O_{S_1}, O_{S_2}) \stackrel{\text{def}}{=} O_{S_1} \simeq O_{S_2} \tag{13}$$

In other words, we take object similarity to be a sufficient condition for `sameGenus` to hold. This is appropriate for the base case of hierarchies of depth two, with the implicit assumption that objects with different genus are all instances of a generic *thing* object. We leave the generalization to deeper hierarchies to future work. A hint on how to perform such generalization can be found in the Conclusion Section.

Differentia. We implement `Different` as a boolean function which holds for two objects with the same `genusObj` if there is no visual object, aside the ones in their `genusObj`, which makes the two objects similar, namely:

$$\text{Different}(O_{S_1}, O_{S_2}) \stackrel{\text{def}}{=} \nexists v^i \in O_{S_1} \setminus \text{genusOf}(O_{S_1}). \\ \nexists v^j \in O_{S_2} \setminus \text{genusOf}(O_{S_2}) : v^i \simeq v^j \tag{14}$$

where `genusOf` is a function which computes the `genusObj`, as:

$$\text{genusOf}(O_{S_1}) \stackrel{\text{def}}{=} \{v^i \in O_{S_1} \mid \exists O_{S_2}, \exists v^j \in O_{S_2} : \text{SG}(O_{S_1}, O_{S_2}) \wedge v^i \simeq v^j\} \tag{15}$$

where the function `SG(OS1, OS2)` returns true if the user in the past gave supervision (see below), telling the algorithm that `OS1` and `OS2` share a `genusObj` while being different. Figure 4 shows two objects with the same `sameGenus` (green lines) but for which also `Different` holds (the two red visual objects are different).

User feedback. We use two functions `ASKSAMEGENUS` and `ASKDIFFERENT` which ask the user, when available, about `sameGenus` and `Different` between an encounter

E and an object O stored in memory. If the user is not available, they return $\text{sameGenus}(O, E)$ and $\text{Different}(O, E)$, respectively. Notice how the user intervention is exploited *exactly and only* in the computation of sameGenus and Different , to consolidate object similarity into object identity, as from the previous section. The user feedback, collected by ASKSAMEGENUS and ASKDIFFERENT , is exploited by a function $\text{UPDATESIMILARITY}(\mathcal{M})$ whose goal is to adjust the diversity threshold θ , see Eq. (9), based on the knowledge available so far. The threshold is computed using the strategy, proposed in [11], each time a new supervision is provided by the user. These supervisions are stored as a set:

$$\mathcal{K} = \{ \langle \delta_i, y_i \rangle \mid 1 < i < |\mathcal{K}| \}$$

where $\delta_i = d(O_i, E_i)$ is the distance between object-encounter pairs, coupled with a boolean value $y_i = \text{ASKSAMEGENUS}(O_i, E_i)$ containing the supervision of the user. The value of θ is computed solving the following problem:

$$\theta = \underset{\lambda}{\operatorname{argmax}} \sum_{i=1}^{|\mathcal{K}|} \mathbb{1}((\delta_i < \lambda) \oplus \neg y_i) \quad (16)$$

where $\mathbb{1}$ is the indicator function mapping *True* to 1 and *False* to 0, and \oplus is the exclusive OR. [11] provides a strategy for how to efficiently solve this problem by performing a number of evaluations equal to $|\mathcal{K}|$.

Algorithm 1 Build subsumption hierarchy

```

1: procedure BUILDSUBSUMPTIONHIERARCHY
2:    $\mathcal{M} \leftarrow \emptyset$ ;
3:   while True do
4:      $E \leftarrow \text{PERCEIVE}()$ 
5:      $O \leftarrow \text{GETMOSTSIMILAROBJECT}(E, \mathcal{M})$ 
6:     if  $\text{ASKSAMEGENUS}(O, E)$  then
7:       if  $\text{ASKDIFFERENT}(O, E)$  then
8:          $\text{ADDOBJECT}(\mathcal{M}, E)$ 
9:       else
10:         $\text{UPDATEOBJECT}(\mathcal{M}, O, O \cup E)$ 
11:     else
12:        $\text{ADDOBJECT}(\mathcal{M}, E)$ 
13:      $\text{UPDATESIMILARITY}(\mathcal{M})$ 

```

The algorithm building the subsumption hierarchy is implemented as the infinite loop shown in Algorithm 1. This algorithm is a direct implementation of the recursive construction of objects given in Eq. (3) via the test for object equality and subsumption as from Eqs. (7), (8). We use a function $\text{GETMOSTSIMILAROBJECT}$ which, given an object and a cumulative memory \mathcal{M} of all the objects perceived so far, returns the object which is most similar. The implementation of this function is based on the consideration that there are two possible cases. In the first, that same object

was previously seen, and therefore, this is the object to be selected. In the second, the object was not previously seen, in which case there may be no objects sharing visual objects (no objects with the same genusObj) or there may be one or more similar objects, possibly including the genusObj , which share the genusObj with the new object. Based on this intuition $\text{GETMOSTSIMILAROBJECT}$ returns the nearest already seen instance that satisfies the similarity constraint of Eq. (12), if existing, otherwise it returns the most similar genusObj , computed as described above. Notice that we make the further simplifying assumption to ask the user, via ASKSAMEGENUS , only for the most similar element. Thus the model is not guaranteed to keep a hierarchy always in line with the desires of the user. Ideally one should ask for supervision for every similar object. This choice was made to limit the effort required to the user. We will deal with this problem in further research following the line of thought already started in [11].

For what concerns lines 6–12 of the algorithm, we have the following: (i) in Line 8, it creates a new object because Different holds for an object with the same Genus (as from Eq. (8), this is the case when subsumption holds); (ii) in Line 10, it extends an already existing object for which sameGenus holds but Different does not (as from Eq. (7), this is the case when we have object identity); in Line 12, it creates a new object corresponding to an instance of a new genusObj (as from Eq. (4)). It is easy to see how the hierarchy satisfies, for any given sequence of encounters, the five conditions provided in the previous section.

Each iteration of the algorithm requires a forward step in the embedding network followed by a nearest-neighbour search in the memory of stored encounters. A simple linear search is sufficient for real-time interaction for reasonable sized data sets, compatible with the need for supervision by a single person. On the other hand, the approach can easily scale to arbitrary data sets by leveraging techniques, such as tree-decomposition [6] or locality-sensitive-hashings [56, 57] for efficient exact and approximate nearest-neighbor search. These approaches can be complemented with prototype selection strategies [3, 15, 63] to combine time and memory efficiency.

As a concluding remark, notice how the algorithm satisfies the requirements listed in the introduction: (i) the hierarchy is built autonomously and it becomes a hierarchy of classification concepts thanks to the user feedback, (ii) no assumption is made about the input objects, (iii) the algorithm learns objects never seen before and (iv) it incrementally learns how to recognize objects starting from no objects.

Fig. 5 Sample frames from the data set. The left and right columns represent two wallets that only differ by the card they contain. The discriminative view is only present in the bottom row (highlighted in red)



Experiments

The algorithm was implemented in PyTorch and the implementation can be freely downloaded at <https://github.com/lucaerculiani/towards-visual-semantic>. In all experiments we have used a moving average of size fifty and stride fifteen to create the visual objects. The embedding network is a standard VGG-16 architecture [49] which was pre-trained on the YFCC100M data set [53] in an unsupervised fashion with the DeeperCluster algorithm (see [5] for the details of how the network was pre-trained).

Data set

The main difficulty in setting up the experiments was that no existing data set matches the conditions we needed to properly evaluate our framework. This setting requires a collection of objects that can be grouped on the basis of their visual appearance. Inside each group, all objects must have some partial views that make them *indistinguishable* from the other elements of the group (the *Genus*), while at the same time having other views that enable the discrimination of single objects (the *Differentia*). No public data set enforces this constraint. As a consequence we have created our own data set, which consists of a collection of video sequences of various objects, recorded while rotating or being deformed against a blank background, making sure that each video contained only a *partial* view of the object. We have made the simplifying assumptions of a blank background, which is clearly limiting for a real world application. This assumption is motivated by the focus on the recognition of *Genus* and *Differentia*, rather than on the distinction between objects and background.

The resulting data set⁵ contains videos for five different types of objects: a coffee pod, a multiplug, a pencil case, a

smartphone and a wallet. For each object type we recorded videos for two different instances, that were different only for a certain view (as in Fig. 4). For each object instance we recorded five videos that contain the discriminative view, and five that do not. Videos were recorded at 60 fps and lasted between 1 and 5 s. Figure 5 shows some sample frames for wallets. The left and right columns represent two distinct wallets (that should be recognized as having the same *Genus*), that only differ by the card they contain. The top row shows (excerpts of) videos that do not contain the discriminative view (and should thus be predicted as not having *Differentia*), while the bottom row shows videos of the same objects where the *differentia* is visible (the red frame in each sequence).

Experimental Results

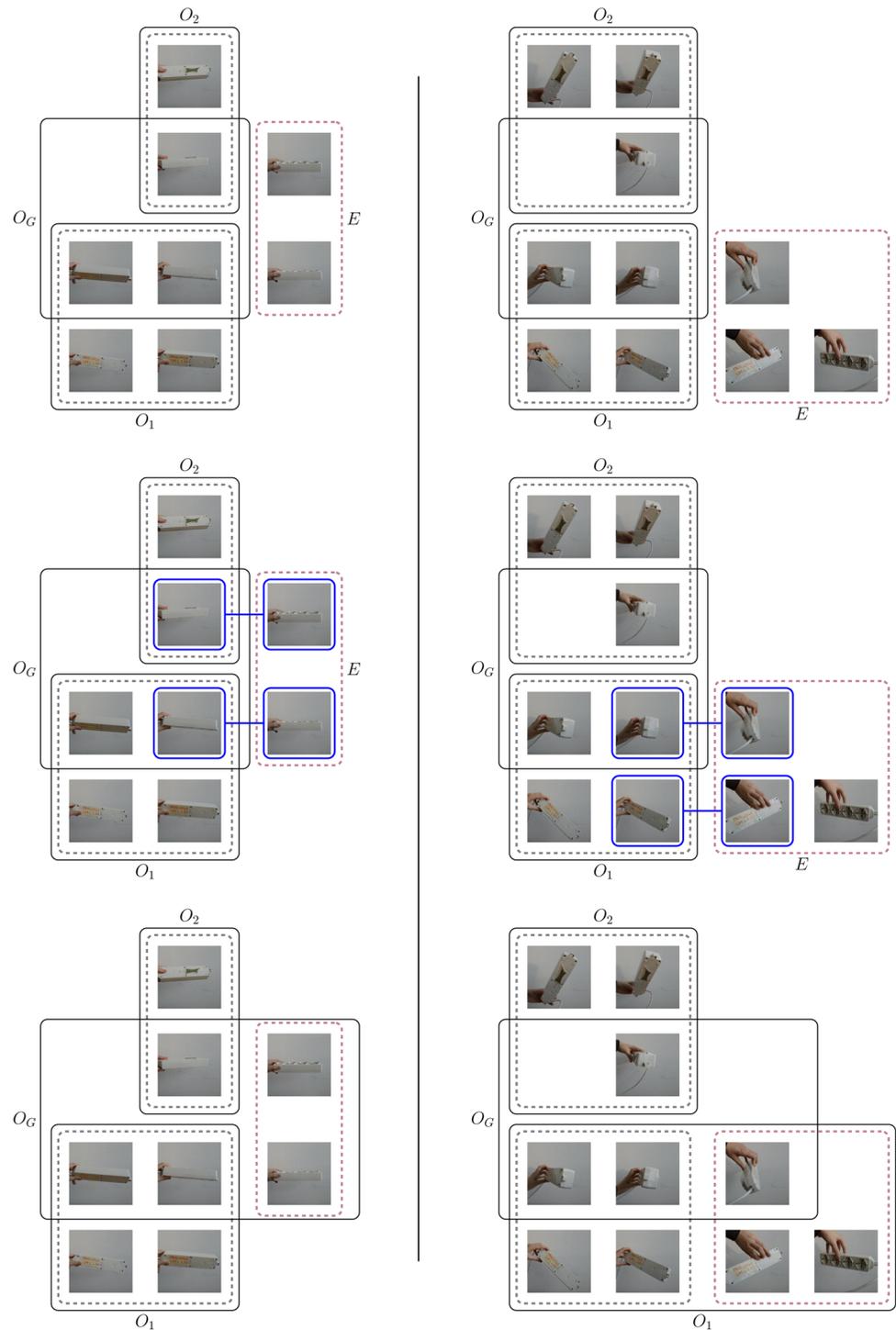
In the following we report first qualitative and then quantitative results in terms of capacity of the learning algorithm to recover the notions of *Genus* and *Differentia* of the user. Below, we say that the answer is correct when this is the case, incorrect otherwise.

Qualitative Results

Figure 6 shows two cases of encounters that were correctly processed by the algorithm with no user intervention. Each column represents the sequence of steps made to process a new encounter (the visual objects in the purple box), namely, perception, recognition and memorization, and the two columns represent cases giving rise to different choices made by the algorithm. The encounters already present in memory are represented by gray dashed boxes, and the corresponding objects by black boxes. A box covering visual objects from multiple objects represents the *genus* of that group of objects. The linked couples of blue visual objects represent items that were recognized as similar by the machine. In the left column, a new encounter is correctly recognized as having the same *Genus* of two objects already stored in memory. The *genusObj* is updated by incorporating the visual objects of the encounter. This is all the algorithm can

⁵ The data set is freely available at https://figshare.com/articles/dataset/small_re-identification_dataset_with_classes/14706003, where both raw data and precomputed embeddings can be downloaded.

Fig. 6 Examples of two correct choices made by the algorithm. The left column depicts a case in which the machine correctly identified the Genus of the new encounter (the encounter does not contain enough information for instance-level recognition). The second column represents a case in which the new encounter was correctly identified as an already seen object. The new visual objects were added to the matched object. In addition, the `genusObj` was updated with the subset of visual objects matching it

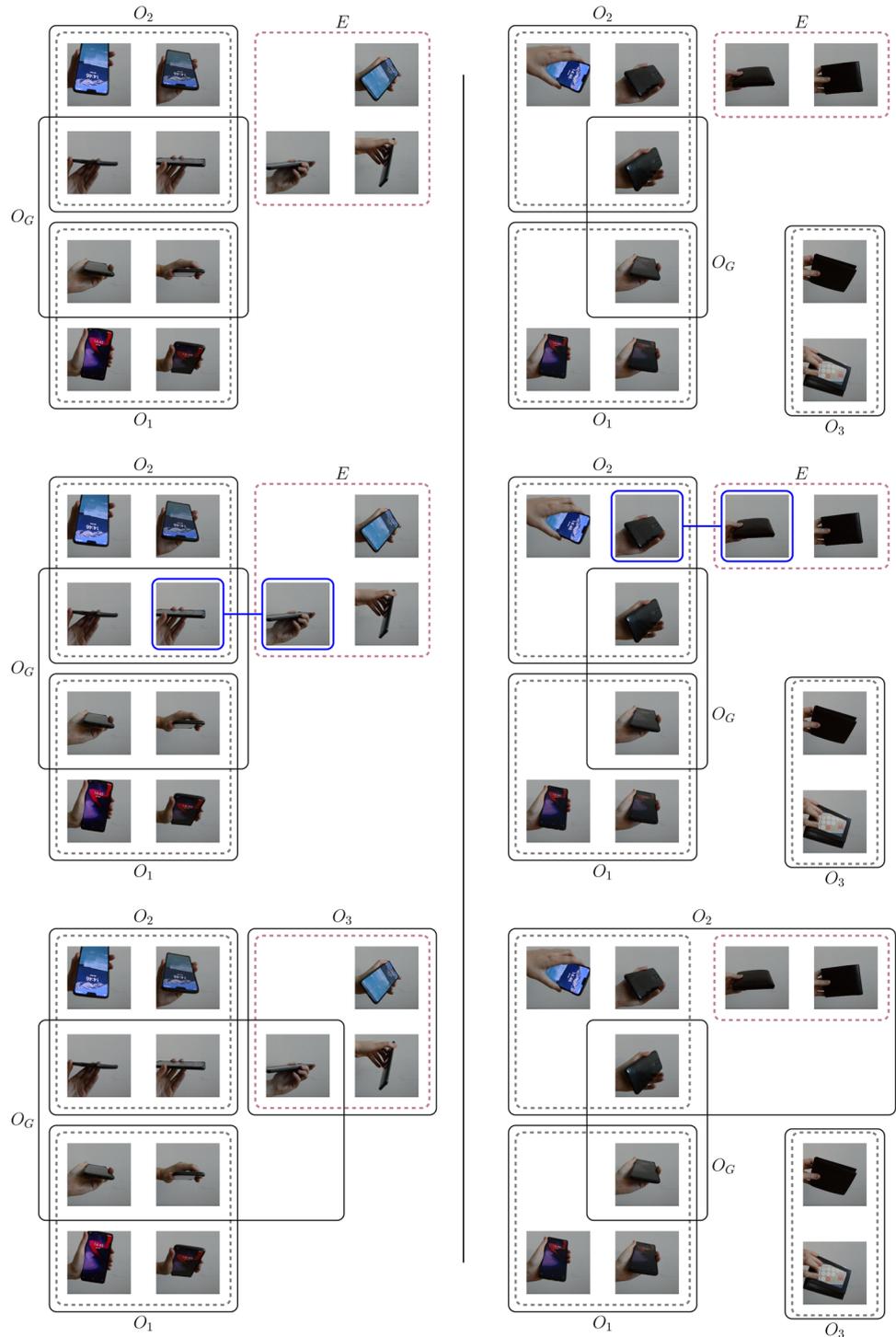


do, as the encounter does not have enough visual information to allow for an instance-level recognition. In the right column, a new encounter is correctly recognized as being the same as an object stored in memory. The visual objects of the encounter are added to the retrieved object, while its `genusObj` is updated with the visual objects that are found similar to it. Note how updating the object enriches

its representation by including a viewpoint that was never observed before (the one showing the sockets).

Figure 7 shows two cases of encounters in which the algorithm made choices which are not aligned with the user perspective. In the left column, the algorithm manages to recognise the Genus of the new encounter, but fails to realize that the encounter is actually the same as one of the objects it

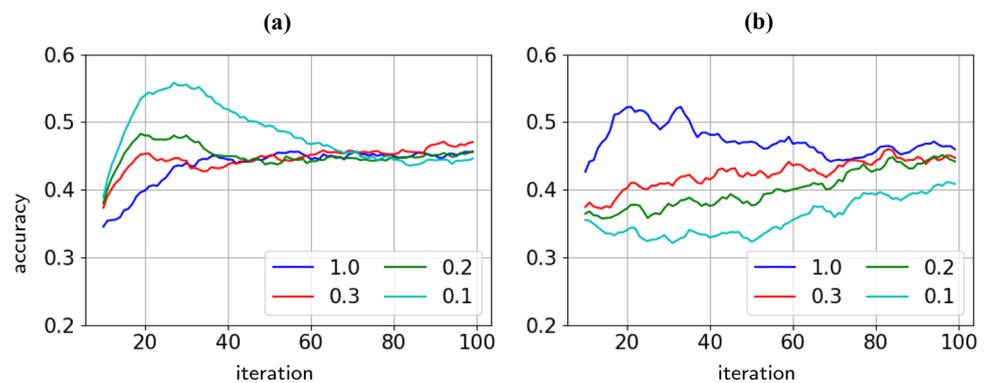
Fig. 7 Examples of two incorrect choices made by the algorithm. The left column depicts a case in which the machine recognized the correct *genusObj* for the new encounter but not the correct instance. This led to the creation of a new separate object with that *genusObj*. The second column represents a case in which the new encounter (containing a wallet) was mistakenly assigned to a completely different object (a smartphone)



has seen before. In so doing it wrongly creates a new object in memory. This error can be avoided if user feedback is available to answer an ASKDIFFERENT query (line 7 of Algorithm 1). The right column represents a case in which the new encounter was mistakenly identified as a completely different object. In this case, the availability of user supervision can prevent the algorithm from performing the wrong match

(and spoiling the representation in memory of the retrieved object). The algorithm would in this case create a new object initialized with the encounter. Note, however, that in case the encounter was indeed an instance of an already stored object (but different from the one retrieved by the machine), or shared a Genus with it, asking feedback for the most similar object only as done in Algorithm 1 would not suffice

Fig. 8 Accuracy of prediction for Genus (a) and Differentia (b), respectively, for increasing number of iterations and different amounts of supervision (curves for different values of α)



to discover it (see discussion in Section “The Learning Algorithm” on the limitations of this choice). Indeed, the algorithm should progressively ask for feedback on a sequence of objects (of decreasing similarity) until the user confirms the match, which could end up being too demanding for the user. A possible solution is that asking the user to provide names for objects and genres, thus making the mapping between substance and classification concepts explicit. This, however, does not solve the problem entirely, as without full supervision the memory would contain objects without names. Note also that a purely name-driven supervision cannot work, for the very reasons that have been discussed when contrasting substance concepts with classification concepts (e.g., the user could provide the name of a genus when the new encounter also has a differentia). We plan to investigate in future work a hybrid similarity-driven and name-driven retrieval strategy in conjunction with the extension of the method to hierarchies of arbitrary depth.

Quantitative Results

What described above provides a qualitative view of the behaviour of the algorithm, which largely depends on the availability of user feedback. We have also ran a quantitative evaluation showing how recognition performance over time is affected by the amount of supervision. The experiment is organized as follows. Sequences are showed one after the other and at each iteration the user provides supervision with probability α . We have run experiments for different values of α with $\alpha \in \{1.0, 0.3, 0.2, 0.1\}$ and where $\alpha = 1.0$ is the setting where supervision is always available. In all settings the model is provided with a supervision for each of the first five sequences to bootstrap the estimation of the diversity threshold θ . We ask the model to predict `sameGenus` and `Different` at each iteration before receiving feedback from the user (if available, otherwise the algorithm prediction is used to update the memory). The results of the experiment are depicted in Fig. 8.

Figure 8a presents the accuracy computed for the prediction of Genus. A prediction is correct if the new encounter

shares a Genus with an object in memory and the algorithm retrieves the correct `genusObj`, or the algorithm correctly identifies the encounter as a completely novel object. The plotted results are computed as the mean accuracy of the prediction over two thousand different runs, each with a different order of the sequences, smoothing the curves using a moving average of length five. Surprisingly enough, in the first half of the experiment the smaller the supervision the better the accuracy. This is due to the fact that at the beginning, most sequences contain new objects, thus the more the supervision the higher the bias of the model to predict a new sequence as unseen. This bias progressively fades away proceeding with the experiment, and all models end up achieving similar results on average. These results suggest that even a very limited amount of supervision is sufficient to learn a reasonable value for the diversity threshold, which is what the algorithm needs to retrieve objects with the same genus if stored in memory (see Eqs. 9 and 12).

Figure 8b shows the accuracy of the prediction of `Different`, over the subset of sequences for which the Genus is predicted correctly. The task here is more complex, as the algorithm needs to gather enough supervision to characterize the genus object (`genusObj`) so as to identify the `Differentia` (see Eqs. 14 and 15). Indeed, in this case the greater the amount of supervision, the better the model is capable of recognizing whether the new sequence contains enough information to identify the correct instance. Apart for the setting with least supervision ($\alpha = 0.1$), for which the performance gap with respect to the fully supervised case stays rather large, the different models end up achieving comparable performance.

Overall, these preliminary results indicate that the algorithm is capable of progressively acquiring the notions of Genus and Differentia with reasonable accuracy despite seeing a small number of examples and receiving supervision on a fraction of them. These results are a proof-of-concept of the feasibility of the approach. The main limitation of this work is the fact that we are restricting ourselves to hierarchies of depth 2. Extending the learning algorithm to deeper hierarchies is indeed our main target for future

research, as discussed in the conclusion of the paper. This will also require building a much bigger data set, having as main reference the Imagenet data set [44]. Building this data set and the corresponding hierarchy of classification concepts is a research task in itself.

Related Work

As already hinted in the introduction, this work is grounded in the work done in Teleosemantics. At the same time, the distinction between substance concepts and classification concepts resembles the two types of concepts proposed by Millikan and Fodor, see also their debate on Recognitional Concepts [12, 32]. In fact, substance concepts map quite naturally to Millikan's recognitional concepts while classification concepts seem to be a good conceptualization of Fodor's work on the structure of semantic theories [23]. The work provided here suggests that we need both types of concepts, which functionally serve different problems, the crucial issue being how to keep them aligned.

This work constitutes a major shift from mainstream KR and Computer Vision in four dimensions. First, it treats objects as perdurants, where classification concepts are, instead, endurants. In other words, we assume that we have two (very different) representations for anything we perceive, e.g., a *person*. Objects are assumed to be represented only partially and to evolve in time building (modulo forgetting) richer and richer but never complete visual representations. Second, it uniformly models instances and classification concepts as substance concepts, and, therefore, as sets of visual properties. Thus, substance concepts, i.e., objects, are visual representations of both classification concepts and instances. Third, object visual similarity is *not* taken to be the same as object identity, this latter notion applying only to classification concepts. Fourth the user is never asked about the name of an object but only about *Genus* and *Differentia*.

The work proposed in this paper is a (first step towards the) solution of the SGP. The previous work so far has been on how to integrate feature-level information with semantic level information. Thus, some early work has proposed to encode semantic information via ontologies [21], others propose to use tags or similar high-level features [10, 26], others propose to involve users using active learning [52], most recently it has been proposed that the semantic gap should be handled in DNNs when aggregating multi-level features [37]. The common denominator is that this work exploits classification concepts, rather than substance concepts, and that it does not build the subsumption hierarchy. All these proposals do not provide a general solution to the SGP. A fair amount of work has also been done trying to model objects in a way which is compliant to how humans

think about objects. Most of this work, motivated by Robotic applications has concentrated on identifying the function of objects see, e.g., [4, 8, 40, 51, 58]. But because of its very purpose, this work models objects as classification concepts.

The visual hierarchy proposed in this work naturally reminds of the work on hierarchies done in Computer vision and, in particular the work on ImageNet [7]. The introduction of the ImageNet data set and its associated challenge [44] has boosted image classification towards (and even beyond) human-level performance. While most research has focused on fine-grained classification of (subsets of) leaf classes, hierarchical classification has also been directly addressed [59]. However, this work assumes a *predefined* hierarchy given in advance, as well as a fixed set of examples to learn from. Furthermore, our focus is on the classification of videos of objects rather than static images. It is part of our future work to collect a data set of object videos that resembles the ImageNet data set in terms of size and depth of the hierarchy. A promising direction consists in leveraging the recent developments in terms of Embodied AI simulators [9], that, however, need to be adapted in terms of quantity, diversity and granularity of the concepts that can be represented.

In recent years, there has been a growing interest towards open world [1, 43], continual and lifelong learning [38]. Most approaches focus on the sequential learning of novel classes via class-specific training sessions, trying to avoid catastrophic forgetting [19] by, e.g., parameter regularization [24, 61], model capacity expansion [45, 60] or task rehearsal [48, 54]. Alternatives accounting for unsupervised [41] or task agnostic [62] settings have also been recently explored. However, the underlying assumption is always the presence of a predefined (possibly not explicit) set of classes, that are progressively presented to the algorithm. Even the open-world classification setting [1, 43], where the learner should be able to tell if an entity does not belong to the set of known classes (the so-called open-set classification [2, 47]), requires a specific training session to incorporate novel classes. On the other hand, few-shot learning methods [42, 46, 55], that address the scarcity of data using similarity-based or meta learning approaches, are typically closed-world and offline, with well-separated training and testing phases. A fully online incremental and agnostic setting where the hierarchy of objects emerges from the combination of encounters and feedback from the user is beyond the scope of these approaches.

Conclusion

In this paper we have provided the first steps towards a general theory of visual semantics. The ultimate goal is to understand the general mechanisms by which it is possible

to align the meaning of words with the perception of the objects named by those words. The main foundational contribution of this paper is the distinction between substance concepts and classification concepts, the first being modeled as perdurants the latter as endurants, and the mechanism by which these two different types of concepts must be aligned. This latter result has highlighted the crucial role of humans, not so much to tell machines what objects are, machine can learn this by themselves, but to make sure that what they learn is coherent with how humans perceive the world.

The future work will proceed in many directions. The first will be the extension of this work to hierarchies of any depth. For what concerns the algorithm, the main requirement is the addition of a system to build a new `genusObj` on top of an existing `genusObj`. The general idea we foresee is that of ditching the global threshold mechanism in favor of a different similarity threshold (or metric) for each `genusObj`. The second extension is that of combining similarity-based retrieval with name-based retrieval, to quickly but reliably identify the position(s) in the memorized hierarchy where to put the new encounter. We also plan to reduce the burden of the user by introducing online active learning strategies (see [11] for an initial solution in a flat instance-level recognition task).

Finally, the algorithm should be adapted to deal with tougher visual contexts including variable background, occlusions, noisy feedback etc.

Acknowledgements This paper was supported by the “*WeNet - The Internet of Us*” and the *TAILOR* projects, funded by the European Union (EU) Horizon 2020 programme under GA number 823783 and 952215, respectively.

Funding Open access funding provided by Università degli Studi di Trento within the CRUI-CARE Agreement.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Bendale A, Boulton TE. Towards open world recognition. In: CVPR. 2015;pp. 1893–1902.
2. Bendale A, Boulton TE. Towards open set deep networks. In: CVPR. 2016;pp. 1563–1572.
3. Bien J, Tibshirani R. Prototype selection for interpretable classification. *The Annals of Applied Statistics* 2011;5(4). <https://doi.org/10.1214/11-AOAS495>
4. Bogoni L, Bajcsy R. Interactive recognition and representation of functionality. *Comput Vis Image Underst.* 1995;62(2):194–214.
5. Caron M, Bojanowski P, Mairal J, Joulin A. Unsupervised pre-training of image features on non-curated data. In: ICCV. 2019:2959–2968.
6. Clarkson KL. Nearest-neighbor searching and metric space dimensions. In: *Nearest-Neighbor Methods for Learning and Vision: Theory and Practice*. MIT Press 2006.
7. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. 2009:248–255. IEEE.
8. DiManzo M, Trucco E, Giunchiglia F, Ricci F. Fur: understanding functional reasoning. *Int J Intell Syst.* 1989;4(4):431–57.
9. Duan J, Yu S, Tan HL, Zhu H, Tan C. A survey of embodied ai: From simulators to research tasks 2021.
10. Elahi M, Deldjoo Y, Moghaddam FB, Cella L, Cereda S, Cremonesi P. Exploring the semantic gap for movie recommendations. In: *Proceedings of the Eleventh ACM Conference on Recommender Systems*. 2017:326–330.
11. Erculiani L, Giunchiglia F, Passerini A. Continual egocentric object recognition. *ECAI* 2020.
12. Fodor J. There are no recognitional concepts: not even RED. *Philos Issue.* 1998;9:1–14.
13. Fumagalli M, Bella G, Giunchiglia F. Towards understanding classification and identification. *Pac Rim Int Conf Artif Intell.* 2019:71–84.
14. Gangemi A, Guarino N, Masolo C, Oltramari A, Schneider L. Sweetening ontologies with dolce. In: *International Conference on Knowledge Engineering and Knowledge Management*. 2002:166–181. Springer.
15. Garcia S, Derrac J, Cano J, Herrera F. Prototype selection for nearest neighbor classification: taxonomy and empirical study. *IEEE Trans Pattern Anal Mach Intell.* 2012;34(3):417–35. <https://doi.org/10.1109/TPAMI.2011.142>.
16. Giunchiglia F, Batsuren K, Freihat AA. One world—seven thousand languages. In: *Proceedings 19th International Conference on Computational Linguistics and Intelligent Text Processing*. 2018:18–24.
17. Giunchiglia F, Batsuren K, Bella G. Understanding and exploiting language diversity. In: *IJCAI*. 2017:4009–4017.
18. Giunchiglia F, Fumagalli M. Concepts as (recognition) abilities. In: *FOIS*. 2016:153–166.
19. Goodfellow IJ, Mirza M, Xiao D, Courville A, Bengio Y. An empirical investigation of catastrophic forgetting in gradient-based neural networks. In: *ICLR* 2014.
20. Guarino N. The role of identity conditions in ontology design. In: *International Conference on Spatial Information Theory*. pp. 221–234. Springer 1999.
21. Hare JS, Lewis PH, Enser PG, Sandom CJ. Mind the gap: another look at the problem of the semantic gap in image retrieval. In: *Multimedia Content Analysis, Management, and Retrieval* 2006. vol. 6073, p. 607309. International Society for Optics and Photonics 2006.
22. Hirsch E. *The concept of identity*. Oxford: Oxford University Press; 1982.

23. Kats J, Fodor J. The structure of a semantic theory. *Language*. 1963;39(2):170–210.
24. Kirkpatrick J, Pascanu R, Rabinowitz N, Veness J, Desjardins G, Rusu AA, Milan K, Quan J, Ramalho T, Grabska-Barwinska A, et al. Overcoming catastrophic forgetting in neural networks. *PNAS* p. 201611835, 2017.
25. Lalumera E. Concepts are a functional kind. *Behav Brain Sci*. 2010;33(2–3):217–8.
26. Ma H, Zhu J, Lyu MRT, King I. Bridging the semantic gap between image contents and tags. *IEEE Trans Multimed*. 2010;12(5):462–73.
27. Macdonald G, Papineau D, et al. *Teleosemantics*. Oxford: Oxford University Press; 2006.
28. Martin A, Chao LL. Semantic memory and the brain: structure and processes. *Curr Opin Neurobiol*. 2001;11(2):194–201.
29. Miller GA, Beckwith R, Fellbaum C, Gross D, Miller KJ. Introduction to wordnet: an on-line lexical database. *Int J Lexicogr*. 1990;3(4):235–44.
30. Millikan RG. *Language, thought, and other biological categories: new foundations for realism*. New York: MIT Press; 1984.
31. Millikan RG. Biosemantics. *J Philos*. 1989;86(6):281–97.
32. Millikan RG. A more plausible kind of “recognitional concept”. *Philos Issue*. 1998;9:35–41.
33. Millikan RG. *On clear and confused ideas: An essay about substance concepts*. Cambridge: Cambridge University Press; 2000.
34. Millikan RG. *Varieties of meaning: the 2002 Jean Nicod lectures*. New York: MIT Press; 2004.
35. Millikan RG. *Language: a biological model*. Oxford: Oxford University Press; 2005.
36. Noonan H. *Identity*. Dartmouth, Aldershot USA 1993.
37. Pang Y, Li Y, Shen J, Shao L. Towards bridging semantic gap to improve semantic segmentation. In: *International Conference on Computer Vision (ICCV)*. 2019:4230–4239.
38. Parisi GI, Kemker R, Part J, Kanan C, Wermter S. Continual lifelong learning with neural networks: a review. *Neural Netw*. 2019;113:54–71.
39. Parry WT, Hacker E. *Aristotelian Logic*. New York: State University of New York Press; 1991.
40. Pechuk M, Soldea O, Rivlin E. Function-based classification from 3d data via generic and symbolic models. In: *Proceedings of the National Conference on Artificial Intelligence*. vol. 20, p. 950. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 2005:1999.
41. Rao D, Visin F, Rusu A, Pascanu R, Teh YW, Hadsell R. Continual unsupervised representation learning. In: Wallach, H., Larochelle, H., Beygelzimer, A., d’Alché-Buc, F., Fox, E., Garnett, R. (eds.) *Advances in Neural Information Processing Systems 32*, pp. 7647–7657. Curran Associates, Inc. 2019. <http://papers.nips.cc/paper/8981-continual-unsupervised-representation-learning.pdf>
42. Ravi S, Larochelle H. Optimization as a model for few-shot learning 2017.
43. Rudd EM, Jain LP, Scheirer WJ, Boulte TE. The extreme value machine. *TPAMI*. 2018;40(3):762–8.
44. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, Berg AC, Fei-Fei L. Imagenet large scale visual recognition challenge. *Int J Comput Vis*. 2015;115(3):211–52.
45. Rusu AA, Rabinowitz NC, Desjardins G, Soyer H, Kirkpatrick J, Kavukcuoglu K, Pascanu R, Hadsell R. Progressive neural networks. 2016. [arXiv:1606.04671](https://arxiv.org/abs/1606.04671).
46. Santoro A, Bartunov S, Botvinick M, Wierstra D, Lillicrap T. Meta-learning with memory-augmented neural networks. In: *ICML*. 2016:1842–1850.
47. Scheirer WJ, de Rezende Rocha A, Sapkota A, Boulte TE. Toward open set recognition. *TPAMI*. 2013;35(7):1757–72.
48. Shin H, Lee JK, Kim J, Kim J. Continual learning with deep generative replay. In: *NIPS*. 2017:2990–2999.
49. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings 2015*.
50. Smeulders A, Worring M, Santini S, Jain R. Content-based image retrieval at the end of the early years. *IEEE Trans PAMI*. 2000;22(12):1349–80.
51. Stark L, Bowyer K. Achieving generalized object recognition through reasoning about association of function to structure. *IEEE Trans Pattern Anal Mach Intell*. 1991;13(10):1097–104.
52. Tang J, Zha ZJ, Tao D, Chua TS. Semantic-gap-oriented active learning for multilabel image annotation. *IEEE Trans Image Process*. 2011;21(4):2354–60.
53. Thomee B, Shamma DA, Friedland G, Elizalde B, Ni K, Poland D, Borth D, Li LJ. Yfcc100m: the new data in multimedia research. *Commun ACM*. 2016;59(2):64–73.
54. van de Ven GM, Tolia AS. Generative replay with feedback connections as a general strategy for continual learning. 2018. [arXiv:1809.10635](https://arxiv.org/abs/1809.10635).
55. Vinyals O, Blundell C, Lillicrap T, Wierstra D. Matching networks for one shot learning. In: *NIPS*. 2016:3630–3638.
56. Wang J, Zhang T, Song J, Sebe N, Shen HT. A survey on learning to hash. *IEEE Trans Pattern Anal Mach Intell*. 2017;99:1.
57. Wang J, Shen HT, Song J, Ji J. Hashing for similarity search: a survey. 2014. [arXiv:1408.2927](https://arxiv.org/abs/1408.2927).
58. Woods K, Cook D, Hall L, Bowyer K, Stark L. Learning membership functions in a function-based object recognition system. *J Artif Intell Res*. 1995;3:187–222.
59. Yan Z, Zhang H, Piramuthu R, Jagadeesh V, DeCoste D, Di W, Yu Y. Hd-cnn: Hierarchical deep convolutional neural networks for large scale visual recognition. In: *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*. 2015:2740–2748. ICCV ’15, IEEE Computer Society.
60. Yoon J, Yang E, Lee J, Hwang S. Lifelong learning with dynamically expandable networks. In: *ICLR 2018*.
61. Zenke F, Poole B, Ganguli S. Continual learning through synaptic intelligence. In: *ICML*. 2017:3987–3995.
62. Zeno C, Golan I, Hoffer E, Soudry D. Task agnostic continual learning using online variational bayes 2018. [arXiv:1803.10123.pdf](https://arxiv.org/abs/1803.10123)
63. Zhang X, Zhu Z, Zhao Y, Chang D, Liu J. Seeing all from a few: ℓ_1 -norm-induced discriminative prototype selection. *IEEE Trans Neural Netw Learn Syst*. 2019;30(7):1954–66.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.