Helping a robot to understand human actions and objects: a grammatical view

Florentin Wörgötter¹ · Minija Tamosiunaite¹

Received: 5 January 2018 / Accepted: 9 March 2020 / Published online: 22 April 2020
© The Author(s) 2020

Abstract

Humans are able to perform a wide variety of complex actions manipulating a very large number of objects. We can make predictions on the outcome of our actions and on how to use different objects. Hence, we have excellent action and object understanding. Artificial agents, on the other hand, still miserably fail in this respect. It is particularly puzzling how inexperienced, young humans can acquire such knowledge; bootstrapped by exploration and extended by supervision. In this study we have, therefore, addressed the question how to structure the realm of actions and objects into dynamic representations, which allow for the easy learning of different action and object concepts. Performing different manipulation actions on a table top (e.g. the actions of “making a breakfast”), we show with our robots that this will indeed lead to some kind of implicit (un-reflected) understanding of action and object concepts allowing the agent to generalize actions and redefine object uses according to need.

Keywords Action · Human observation · Object · Semantics · Syntax

1 Introduction

Why are humans so good at interpreting actions (performed by others) and why do we also have such an excellent understanding of the use of the different objects in our world? These questions are puzzling from the perspective of cognitive science but also from a robotics viewpoint. Action- and object-understanding seems an unsurmountable problem: The same action can be performed in so many different ways! Why does a 2 years know that his/her mother makes a sandwich regardless of how, when and where this happens? Furthermore, there are so many objects existing! Hence, why does the same child—maybe a bit older—know exactly what knives and forks are good for given that there are vast amounts of different types of such items existing. For example, he/she will correctly infer, with great likelihood, that a

never-before-seen three-pronged fork is just “another fork” and nothing else. Robots cannot really do any of this.

Here, we would like to discuss a grammatical approach towards action- and object-understanding. We posit that actions and objects have their own grammar, which acts as a scaffold for our understanding and for the generative modeling and the prediction of the function of unknown actions and objects. We will focus on human manipulation actions and on simple everyday objects, but this analysis is not restricted to those.

Let us first summarize the main ideas before we support this with some theoretical and experimental analyses. In the first place, we claim that action understanding does not require object knowledge. Objects do not even have to be present to understand an action: Just think of a pantomime! Our central assumption is that actions can be understood just from the changing spatio-temporal relations between the therein-present entities (which are indeed objects—only we do not care to know which ones). The main temporal structure of a manipulation action is given by the touching and un-touching events between those. This, however, is not yet enough. In addition, we observe that, between two such (touching or un-touching) events, there are phases of motion of those entities (e.g., approach, distancing, conjoint motion, etc.). Those phases will finally clarify, which action it was.

This work was presented in part as an invited speech at the 23rd International Symposium on Artificial Life and Robotics (Beppu, Oita, January 18–20, 2018). The authors accepted the invitation for submission on June 6, 2019.

✉ Florentin Wörgötter
worgott@gwdg.de

¹ University of Göttingen, Göttingen, Germany

Detailed, quantitative knowledge about movement shapes and trajectories is thereby not needed to classify an action. Furthermore, the action does not even have to be completely finished to arrive at safe classification. Thus, this framework is predictive.

Hence, following this, there is indeed a clear sequential structure present in our actions, where motion phases and touching/un-touching events represent the structuring elements. Thus, we posit that this represents the syntax and grammar of actions [1, 2]. But what about “objects”? Objects seem far more diverse and unstructured than actions. Here, however, we can fall back onto a rather old idea: Around 1987, Biederman [3] suggested that all objects can be constructed from only a few elementary geometrical entities, which he called Geons. Later, Rivlin et al. [4] took this idea further and described objects by part graphs, where every node in such a graph represents a part (a geon) of an object, and graph edges indicate that two parts are connected to each other. The idea had great appeal, but at that time, there were no means existing to extract parts (Geons) from computer vision data from an object. Furthermore, a non-annotated part graph of this kind cannot very well describe an object, because it very much matters, too, *how* the parts are connected. Hence, it is required to annotate the graph-edges by so-called “relative pose” information, which represents the relative alignment- and attachment geometry of the connected parts. Under these constraints the annotated part-connectivity graph can, however, indeed be considered—yet again—as a syntactical structure, where certain connectivity rules may exist, constituting a “grammar of objects” (see also [5, 6]).

In the following, we will present some methods and results supporting the notion that such a grammatical view onto actions and objects is useful from the perspective of trying to make a robot understand the world.

2 Methods and results

2.1 Actions

We use 3D point-cloud data that are first segmented by computer vision methods [7, 8] into objects and object parts. Figure 1 shows on top in color code such a segmentation performed on several action frames from a movie of a cutting action. All these parts are then consistently tracked along the movie, but we do not perform any recognition process, except for the “Hand”. Thus, objects have just abstract classes: “Main” (M) is the object first touched by the hand. “Primary” (P) the one from which Main first untouches. “Secondary” (S) the one first touched by Main. The big table in Fig. 1 shows on top the temporal sequence of how any two such objects will touch “T” (or untouch “N”), where

“U” mean that the relation is still unknown. Touching and untouching events present the main temporal action structure. In addition to this, we define a set of 11 static and six dynamic relations that can exist between two objects. Those could be, static: “right”, “left”, “in front of”, etc.; dynamic: “getting closer”, “moving together”, etc. The sub-tables below show how these relations change in the course of an action. This tabular action representation is called: *Extended Semantic Event Chain* (ESEC [9]).

We have performed an analysis of 32 different manipulation actions, as shown in Fig. 2. Those can be grouped into different super-classes (top), like actions for destroying or for rearranging things. The same color code is used in the bottom panel that shows a decision tree. The specific combination of object-object relations that changes from column to column allows recognizing actions after several columns, due to the sequential column structure that they have. At most 7 columns are needed to recognize any action. This tree also shows that cutting (e.g. Fig. 1) can be recognized already at the moment when the fourth column of the ESEC emerges, hence, far before the actual action has ended. In summary, we found that on average, these actions are recognized only after 45.06% (SD=6%) of their total completion time. Hence, we can predict all manipulation actions before they are half-finished.

2.2 Objects

To arrive at object connectivity graphs we use methods described in detail in [10], where we can only give a short summary here leaving all equations out. First, we describe each object-part by a so-called “part-signature” (Fig. 3a). For this, we normalize the maximal distance d between points in the point cloud to one and determine the surface normals at each point. Then, we calculate for all inter-point distances Δd (distances from every point to every other point) the sum of the angular differences $\Delta\alpha$ of all surface normals. As a result, we observe the every part gets its own characteristic part-signature (Fig. 3a). Next, we determine how parts are attached and aligned with respect to each other. For this, we represent each part with its *variance blob* V . The variance blob represents the variance of the point distribution of a part in all possible directions normalized to one. Visualization of these variance blobs can be achieved using spherical plots, as shown in Fig. 3b. In these plots, the variance is denoted by the radius of the surface from the origin. In case of constant variance, this results in a constant radius in all directions, thus in a perfect sphere. A variance blob represents the variance in the Euclidean space of the point cloud. This is why the aspect ratio (the elongation) of a part is directly visible in the elongation of its variance blob. Exact shape

Fig. 1 Top: several frames of a movie of a cutting action and the corresponding 3D scene segmentation (here rendered as colored and arbitrarily numbered image segments). Only five frames are shown. Bottom: Extended Semantic Event Chain (eSEC) table representing 14 frames (columns, yellow). A new column is added to the eSEC whenever at least one touching or static or dynamic relation between the objects in an action changes. Thus, columns represent temporal chunks (of unequal length) given by change events. Objects are: H=Hand, M=main, P=Primary-, S=Secondary Object, T=touch-

ing, N=untouching, U=unknown. We define here the following types of static spatial relations: “Above” (Ab), “Below” (Be), “Top (To), “Inside” (in), and the following dynamic spatial relations: “Moving Together” (MT), “Halting Together” (HT), “Fixed-Moving Together” (FMT), “Getting Close” (GC), “Moving Apart” (MA) and “Stable” (S). X means that an object was broken apart, Q and O refer to objects that are too far away for defining clear relations. White part: touching/untouching events, green: static relation changes, blue: dynamic relation changes

detail (like round, cylindrical, etc.) is ignored. Remember, we had this represented by the part signature (see Fig. 3a).

To calculate the alignment between parts A and B (AL_{AB}), we use the histogram intersection similarity V_{AB} between V_A and V_B (see [10] for details). This is done by calculating the intersection of each bin of variance blob A with each corresponding bin in variance blob B. Finally, we calculate the alignment number AL_{AB} as the sum of the bins of the intersection blob. Note: $AL_{AB} = AL_{BA}$. In addition to this, the attachment number AT_{AB} reflects at which location of part A part B is connected. For this, we calculate the vector connecting centroid A to centroid B: r_{AB} . Then, we retrieve the value on the variance blob V_A in direction of r_{AB} . Since V is normalized, the values of single bins scale reciprocally with the total number of bins. This is why we multiply the result by the total number of bins to get AT_{AB} . Note: $AT_{AB} \neq AT_{BA}$ and we always calculate both. Figure 3c shows how AL remains essentially constant, where AT changes much for three objects, which consist of the same two parts A and B. Figure 4 shows the

final object connectivity graph for one example (artificial) object.

After arriving at the object connectivity graphs, we can now learn possible object-to-function associations. During training, we provide a conventional small neural network with example graphs and manually associate to those the potential function of the belonging object. We have focused here on simple objects (tools!) that relate to the shapes of our human hands like “borer”, “ladle”, “blade”, “hammer”, etc. [10]. The network will in the test phase produce probability values (in percent) for a certain function. Figure 5 shows several objects and their corresponding highest probability function. Interestingly the network sometimes produces high values for several function like for the hammer in the bottom right panel, which the network associates to the “hammer”- but also to a “poker”-function.

Humans often do the very same and “abuse” an object, like in the case of this hammer where you might consider grasping it at the hammer-head to use the stick (hammer handle) as a poker in your fireplace. The use of skulls for

Fig.2 Action categories (top) and sequential unraveling of the different actions (bottom) along the time line (vertical arrow) of the emerging ESEC columns. Actions: 1: Lay; 2: Simple push/pull; 3: Stir; 4: Lever; 5: Hit/Flick; 6: Poke; 7: Bore/Rub/Rotate; 8: Knead; 9: Push from x to y; 10: Cut/Chop/Scissor cut/Squash/Scratch; 11: Draw; 12: Scoop; 13: Push apart; 14: Break/Rip-off; 15: Uncover by Pick and place; 16: Uncover by Push; 17: Push together; 18: Put over; 19: Push over; 20: Pick and place; 21: Take and invert; 22: Shake; 23: Rotate align; 24: Take down; 25: Push down; 26: Put inside; 27: Put on top; 28: Push on top; 29: Pour from a container onto the ground when the liquid first un-touches the container then touches the ground; 30: Pour from a container on the ground when the liquid can touch the container and the ground at the same time; 31: Pour from a container to another container when the liquid first un-touches the container then touches another container; 32: Pour from a container to another container when the liquid can touch the container and another container at the same time. Colors correspond to action categories as shown above

Fig.4 Left: test object with parts A, B, C. Right: object connectivity graph (black disk are nodes, lines are edges, dashed mean that those parts are not physically connected.) Intersection histograms: V_{AB} , V_{AC} , V_{BC} . Values for AL are $AL_{AB}=0.76$, $AL_{AC}=0.99$, $AL_{BC}=0.77$. AT=0 stands for “no contact”, other than that, we leave out AT to save space

drinking has been reported from ancient Germanic tribes who allegedly used the skulls of their enemies to quaff their mead.

3 Conclusions

The current study showed that it is possible to capture the semantics of human actions and the involved objects by leveraging from a grammatical perspective. This may also help to improve robots in their interactions with the world and with us.

Fig.3 Arriving at an object graph. **a** Part signatures histograms (color represents counts) for three example parts (top). Δd is the normalized inner distance between points, $\Delta \alpha$ is the value of the angular difference between surface normals. **b** Alignment AL between

parts A and B for the four hammer-like objects shown on the left side. Colored “balls” represent the Variance Blobs (see text) of the parts and their histogram intersection from which AL is calculated. **c** Alignment and attachment numbers for three example objects

Fig. 5 Example of object-to-function associations for different objects

Acknowledgements Open Access funding provided by Projekt DEAL. This work has been funded by the European Commission, H2020 Program Grant 732266 “Plan4Act”.

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. Aksoy EE, Abramov A, Dellen B, Ning K, Dörr J, Wörgötter F (2011) Learning the semantics of object action relations by observation. *IJRR* 30:1229–1249
2. Yang Y, Fermüller C, Aloimonos Y (2013) Detection of manipulation action consequences. In: proceedings of CVPR, pp 2563–2570
3. Biederman I (1987) Recognition-by-components: a theory of human image understanding. *Psychol Rev* 94(2):115–147
4. Rivlin E, Dickinson SJ, Rosenfeld A (1995) (1995), Recognition by functional parts. *Comp Vis Image Underst* 62(2):164–176
5. Fergus R, Perona P, Zisserman A (2003) Object class recognition by unsupervised scale-invariant learning. In: proceedings of CVPR, pp 264–271
6. Felzenszwalb PF, Girshick RB, McAllester D, Ramanan D (2010) Object detection with discriminatively trained part-based models. *IEEE Trans PAMI* 32(9):1627–1645
7. Stein S, Schoeler M, Papon J, Wörgötter F (2014) Object Partitioning using Local Convexity. In: proceedings of CVPR, pp 304–311
8. Schoeler M, Papon J, Wörgötter F (2015) Constrained planar cuts—object partitioning for point clouds. In: proceedings of CVPR, pp 5207–5215
9. Ziaetabar F, Aksoy EE, Wörgötter F, Tamosiunaite M (2017) Semantic analysis of manipulation actions using spatial relations. In: proceedings of ICRA, pp 4612–4619
10. Schoeler M, Wörgötter F (2015) Bootstrapping the Semantics of Tools: Affordance analysis of real world objects on a per-part basis. *IEEE TAMM* 8(2):84–98

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