Robot Manipulation in Open Environments: New Perspectives

Frank Guerin[1], Paulo Ferreira[2]

***Abstract*—The problem of performing everyday manipulation tasks robustly in open environments is currently beyond the capabilities of artificially intelligent robots; humans are required. The difficulty arises from the high variability in open environments; it is not feasible to program for, or train for, every variation. This correspondence paper presents the case for a new approach to the problem, based on three mutually dependent ideas: 1) highly transferable manipulation skills; 2) choice of representation: a scene can be modeled in several different ways; 3) top-down processes by which the robot's task can influence the bottom-up processes interpreting a scene. The approach we advocate is supported by evidence from what we know about humans, and also the approach is implicitly taken by human designers in designing representations for robots. We present brief results of an implementation of these ideas in robot vision, and give some guidelines for how the key ideas can be implemented more generally in practical robot systems.**

***Index Terms*—Robot Manipulation, Knowledge Representation, Commonsense Reasoning**

I. Introduction

The problem of robot manipulation in open environments is very important for society, e.g., to help the growing population of elderly people as young populations decline. It is also very important commercially for the next wave of robots which are moving out of constrained factory settings and beginning to tackle tasks such as picking or packing varied items in boxes, arranging items on shelves, for warehouses, supermarkets, etc. Current robotics approaches, based on pre-programming or learning from examples, seem unable to cope with the variability of open environments. Yet humans consider the same tasks to be very easy. Even when humans are constrained to use robot hardware via teleoperation, they still cope well with open environments [1], suggesting that hardware is not the primary problem. This poses an interesting scientific problem: humans very easily tackle general manipulation in open environments, but we do not know how to emulate this ability in robots. Perhaps some advantage can be gained by new approaches which borrow from what we know about how humans are doing it differently, at a cognitive level, rather than focusing narrowly on improving vision or motion planning.

Our key insight here is that for humans the task that needs to be done can influence how objects are perceived. A change of task can cause a change of perspective which results in the same scene being represented or modelled differently. Consider the objects in Fig. 1, and how one might model the scene for five different tasks: 1) If the task is to fill crates such as this, the detailed geometry of the walls of the crate can largely be ignored. Each wall could be modeled by its convex hull. 2) If the task is to lift the crate we look in detail for a gripping place for the hands, and largely ignore the detailed geometry of the contents. 3) If we are working in a constrained space and need to put down a smaller box temporarily, we look for a suitable surface, and the tops of these bottles afford such a surface; this requires a grouping of the bottle tops under top-down pressure from the task, to see that they approximate a surface. 4) If we are working in a kitchen and need to roll dough using a suitable roller, we could borrow one of these bottles. 5) If we need a weapon a bottle can also serve that purpose. In each case we take a new perspective on the same objects, modelling them in a different way: ignoring some aspects, approximating others and producing a model that facilitates the task. This cannot work purely from bottom-up perception because there are too many potential creative uses of objects that lead to a different way to model them. Nor can it work purely top-down because we cannot impose a use on an object that contradicts physical facts. We cannot make a square bottle roll. What is required is an interaction between bottom-up and top-down processes.

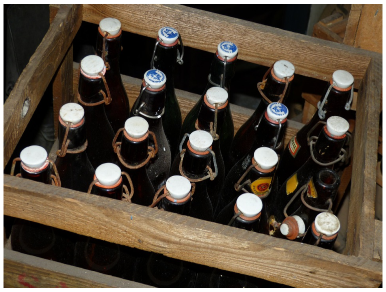


Fig. 1. A crate of bottles. (CC0 license from maxpixel.net)

This paper is structured as a series of connected claims intended to convince the reader of the new approach advocated. Here is a summary of the argument of the following sections: (II) Success in open environments means ability to cope with rare cases. (III) Humans do this very well by transfer; we can borrow ideas from how they do it. (IV) Humans transfer by allowing the task to influence perception and to impose a particular representation on a scene. We call this 'Task-driven representation'. (V) This idea is not so radical or strange and is already implicit in much robotics work. However in existing work it is the human designers who allow the tasks to

Paulo Ferreira was supported by the Brazilian agency CAPES through the program Science without Borders. We are grateful for useful ideas from the anonymous reviewers and Pawel Gajewski, Sven Schmidt-Rohr, Georg Bartels, Ales Leonardis, Subramanian Ramamoorthy.

[1]Frank Guerin, Department of Computing Science, University of Aberdeen, AB24 3UE Aberdeen, Scotland. f.guerin@abdn.ac.uk. [2]Paulo Ferreira, DELL EMC. pauloabelha@gmail.com

influence their perceptions and their engineering of a suitable representation. We need to turn this over to the robot.

II. The Long Tail Problem

This section argues that tackling rare cases is the main stumbling block for robot manipulation in open environments, and that it merits more explicit attention, with specific techniques targeting this general problem.

Gary Marcus and Ernest Davis [2] described how the 'long tail' phenomenon affects many areas in Artificial Intelligence (AI) including language, vision, and robotics: The distribution of scenarios in the real world is such that a small number of cases are extremely frequent (head of distribution), while an enormous number of cases are exceedingly rare (tail). The sum of all the rare cases is so large that a robot will very frequently be faced with rare cases.

Knowledge engineering, pre-programming, or machine learning approaches work very well on the head of the distribution, but more and more training or engineering for different cases will only drive us slightly further down the tail; we will still encounter variations we cannot handle, with unacceptable frequency. Ersen et al. [3] analysed challenges for robot manipulation in human environments, noting that

> "it is not feasible to preprogram robots for all possible contingencies they may face in human environments, a considerable amount of knowledge needs to be figured out by the robots themselves".

Deep learning has been used with massive datasets to tackle robot grasping successfully [4], or in-hand manipulation of a block [5]. However more complex manipulation activities such as using one object on another (such as tool use in a kitchen or workshop) or altering objects before use (such as bending a wire) open up a much higher dimensional space that would need to be sampled. Deep learning has been applied to some other manipulations [6], but needs specific training for each type of manipulation. Lake et al. [7] discuss deep learning's limitations for generalisation, which suggests the need for models, something we discuss further in Sec. V. It seems doubtful that this learning approach is the final solution to tackle a wide variety of rare cases (which don't appear in training) for general manipulation in everyday tasks.

Marcus and Davis [2] point out how the long tail phenomenon explains why AI research often shows the pattern of initially rapid progress followed by slower improvements at a level which falls short of human performance. For applications such as shelf replenishment in supermarkets, or assembly in a workshop, most tasks involve a sequence of operations; even if accuracy on a single step is e.g. 85%, the chance of completing the sequence without failure is small, and one failure is often catastrophic to the sequence. Furthermore, recovering from a failure requires handling a rare case. Humans, in contrast, excel at recovering from failures in these applications.

There is another important consideration which also has the long tail problem, that is the speed of manipulation. With the robot software company ArtiMinds we looked at the business case for robots performing shelf replenishment in supermarkets, over the next ten years. We concluded that robots need to reach at least half of human speed to be economically competitive with human employees. In our analysis of human employees, from video, their speed is striking. Humans often opportunistically exploit shortcuts. E.g. when rearranging cans in a tight space: a human moves cans from one almost empty cardboard tray, to an adjacent one, all in a shelf. The human moves the can horizontally across, tilting it on the way to avoid collision with the small vertical cardboard edge of the trays. One could imagine a robot not seeing the shortcut opportunity here, and completing this rearranging task very slowly: extracting the can from the shelf and reinserting it in the adjacent tray. This would be unacceptably slow however. To be economically viable, robots will need to exploit opportunities for shortcuts, and the scenarios faced again have the long tail distribution.

This suggests that the main problem is how to tackle the rare cases. We believe it merits a new major branch of research which would develop new approaches specifically for handling rare and unexpected cases, rather than pushing existing approaches (e.g. end to end learning, or pre-programming) to handle cases further along the tail.

III. How do humans solve it?

This section makes the claim that humans solve rare cases by transferring solutions from known cases. Here we use the words transfer and analogy with the same meaning; whereas sometimes analogy is taken to mean a more high level symbolic mapping between entities, for example with abstract shapes in IQ tests, we believe the analogy mechanism is at work in lower level transfers also.

Humans cope very well with the types of rare cases that robots fail on, and they do this without needing as many training examples as AI systems. A paper on the limitations of AI planning in real environments states that:

> "An ideal system would be able to behave like humans do in these sorts of environments; in particular, it would have to exhibit creativity, devising new actions that can solve a problem or shorten a plan; use analogy to transfer solutions from other problems..." [8]

Whenever an untrained human meets a rare case, they seem to be able to find some similarity with a familiar example, and hence to transfer a solution. If a human does not have extensive training on rare examples it seems hard to account for their performance without appeal to transfer.

The idea that transfer or analogy is happening in everyday manipulation tasks is also present in Fitzgerald et al. [9], which notes that "While a robot can learn to complete a task from demonstrations, it cannot immediately transfer the learned task model to perform the task in a new environment." The required transfer is described using the language of analogy: 'a mapping between objects in the source and target environments' [9].

In mainstream robotics venues there is almost no work on analogy applied to robotics; people seem to view analogy as more relevant to high level cognition, including IQ tests, or advanced scientific discovery, or artistic pursuits. However we and the authors cited in this section believe it is very prevalent in everyday activities, including manipulation in open environments.

Analogy is more studied in language, and conceptual reasoning [10], [11] than in manipulation (although some manipulation examples are noted by Hofstadter [10, p. 279]). Analogy in manipulation is clear to see in infancy [12, for examples], because an infant's repertoire is small, it is easier to guess where a manipulation skill may have been transferred from. Human infants build up their manipulation skills like a branching tree where every newly acquired skill descends from some others [13]; no skill appears without precursors. Consider the task of retrieving an out-of-reach toy across a table using a rake tool. It is exceedingly difficult for infants up until about 18 months of age [14]. It is surmised that typical infants achieve success on the task by transferring from their knowledge of how to get a spoon behind a piece of food in order to scoop it up. Infants gain experience of this in the 12-18 month period when learning to self-feed with a spoon. A specific study was done on testing this transfer [15]. Infants were tested on a task of retrieving an out of reach object first with a rake and secondly with a large spoon. It was found that some infants who failed with the rake task succeeded when the tool was a spoon. Furthermore those infants who succeeded with the spoon were then able to transfer this back to succeed with the rake afterwards.

The manipulation skills of human adults build on what they have learnt in infancy, and already by two years old a child has an impressive level of widely generalisable skills that robots cannot yet match. Therefore it is worthwhile to study what humans are doing differently to robots [16, from Rod Grupen].

> "In general, skills and abilities in infants and robots are still acquired in quite different ways. Infants build layer upon layer of support skills by exploration that seems to be independent of any other purpose than to acquire comprehensive mastery of increasingly sophisticated relationships to the world. No task is required. The state of the art in robotics, however, typically starts with a target task and is reduced into pieces that are described algorithmically. Typically, a designer anticipates all the events and intermediate states and therefore, the robot is unsupported by the same breadth of contingencies that the infant spends all of its time constructing during the sensorimotor stage of development."

This quote implies a lot of transfer. When the infant attempts a task and is supported by a breadth of 'contingencies' it means that the infant is able to exploit what must have been learnt previously across a variety of different contexts. The 'support skills' that the infant acquires during its exploration are highly transferable. For example the infant learns with small and safe toys, containers, food substances, water, clothing, etc. As the infant grows to a toddler it gains access to new materials and is allowed to explore a wider range of objects and substances, but already has competence because of the ease with which the previously learnt skills are transferred to the new contexts.

The message for roboticists is that robust manipulation will require a lot of 'support skills': highly transferable skills that can help us to solve unexpected problems during a manipulation. Many of these are simple everyday skills such as lifting some object out of the way in order to complete a manipulation, supporting an object with one hand, enlarging an opening to facilitate insertion or removal. Roboticists rarely code such skills unless they are a step that the designer has foreseen as necessary. Furthermore, these skills need to be transferable to a wider variety of contexts.

The conclusion of this section is that we need highly transferable skills, and especially 'support skills': basic manipulations which are not directly related to the performance of a target task in ideal conditions, but which are likely to help in tackling problems that arise during the execution of that task in varied environments.

IV. New Perspective: Task-Driven Representation

In this section we introduce an approach to achieve transfer and analogy in manipulation skills. We will illustrate the idea of new perspectives with three concrete examples, and later explain the general idea in more abstract terms. Finally we give a brief description of an implementation of the idea.

A. Three Concrete Manipulation Examples

The following are three tasks and corresponding manipulation skills that we assume are part of the skill repertoire of a robot, then we show how they could be applied in new ways. These skills could have been acquired by learning from demonstration, or direct coding.

1) Extract a book from a bookshelf full with books: The robot may place one finger on the top edge of the book, exert downward pressure, and pull the book backwards, thereby causing it to rotate and move backwards. This preliminary operation exposes the two sides of the book, providing surfaces to grip and remove the book (see [17]).

2) Lay a cloth down flat on a surface: The robot grasps the cloth at one side (with two hands), lifts it, allowing one side to fall under gravity, then it brings that lower side in contact with one side of the surface and drags the grasped part across to the other side while lowering it, so laying it down flat.

3) Lift a pancake from a pan: The robot grasps a spatula by the handle, orients it appropriately and slides the tip along the pan surface, and under the pancake, then lifts.

Now we look at some problem situations a robot may encounter in a domestic environment, and how the above skills could be transferred to provide a solution. Consider a pizza box that needs to be lifted out of a chest freezer, where the pizza boxes are stacked flat (with the largest surface facing upwards). Without a suction cup there are no available surfaces to get a standard gripper around. If the robot attempts to take a perspective similar to the situation of skill 1) then it can search for a surface to exert perpendicular pressure on and then pull. Here the robot is attempting to create a mapping between the scenario of book extraction and pizza box extraction. The objects and their important components (in this case surfaces) are the entities to map, and the motions applied to components must be adapted appropriately. It can find one exposed side to exert pressure on, and pull that side upwards, thereby rotating the box and exposing sides for grasping. We envisage that a system would explore several candidate mappings internally, and apply reasoning and physics simulation to select the most promising candidate before executing it.

This same skill 1) could also be adapted to apply to other situations: a cylindrical can in a tightly packed shelf can be pulled like the book; a long rectangular box set against a corner between wall and floor does not expose gripping surfaces, but could be pulled horizontally by pressing on one end and pulling, hence exposing surfaces; boxes such as DVD cases, within a larger cardboard box may fall during manipulation, lying flat in the cardboard box and requiring the same operation as the pizza box; to remove a slice of bread from a sliced loaf in a bag the same general operation of applying pressure on one edge and pulling is employed.

Note that a designer could code a very generic skill to apply a 'pull to tilt' to almost any object. The above examples then become a more straightforward application of that skill, and there is no requirement for a new cognitive capability from the robot.[1] However this approach presumes that the designer has thought in advance how each skill may be usefully generalised; i.e. the designer must consider that the surface to be pulled could be on top or on the side, that the pull could be horizontal or vertical, etc. This is not the approach we are advocating. We do not think it feasible that a designer can foresee every useful generalisation that the robot might profit from. Instead the robot itself should have the ability to create novel mappings between scenarios at run-time. We would like to see a more human-like approach: the robot is taught with demonstrations on specific instances, but we expect some intelligent ability to transfer from the robot.

We will proceed to look at adaptations of the two remaining skills. A robot is tidying a house, picking up books from the floor to stack them neatly on a table. Suppose that a thin magazine has been picked by its spine. the other end will droop down. The motion that would put a book on a stack will not work now because the drooping edge obstructs the motion and may become folded. If the robot can adapt skill 2) to this situation it can allow the magazine to hang under gravity's pull, and bring the lower side in contact with one edge of the stack and drag the grasped part across to the other side while lowering it, so laying it down on the stack. Originally the books were abstracted as rectangular blocks, and this was applied to the magazine, but then the perspective switches to treat it like a cloth. Skill 2) can also be applied to situations such as laying a chain or belt out on a worktop, or even a long rigid body if it is difficult to generate the torque to rotate it to be horizontal before placing on a surface.

Finally consider a robot who needs to lift a slice of cheese that is flat on a chopping board. With typical robot fingers it is not easy to lift it by the edge as human fingers could. The robot may try to apply the lifting pancake skill 3). With no suitable thin spatula present the robot may substitute a knife. Here the cutting edge of the knife is not important, and the knife will not be oriented with this edge pointing downward. Instead the width of the blade is important, affording a suitable surface to support the cheese. The pan of 3) maps to the chopping board, the spatula tip maps to the knife tip; the knife can slide along the board and be inserted between cheese and board.

B. Task-Driven Representation

We see above how a skill can be generalised to a new situation by seeing it as similar to the old one, e.g. we choose to see cylindrical cans in a shelf as similar to rectangular books in a shelf. Once we see the new situation as similar to the old we can map the old skill to suitably abstracted components of the new situation. This shows how one skill can apply to many situations. In the other direction one situation can be seen in many different ways so that different skills could be applied. In the book example above we focused on the upper surface of one target book. However suppose that we have a low bookcase which is open at the top, and suppose a robot is carrying a serving tray and needs to set it down somewhere temporarily to free the manipulators for some other urgent task. The robot could see the top surfaces of a row of books as all forming one approximately flat surface that affords support for a tray. Another task might demand that the robot exploit the gap between two books to insert an item.

The cheese scenario above is also open to different perspectives: If a "pierce with fork" skill is in the robot's repertoire then it may choose to see the cheese as a peirceable material and lift it that way. Alternatively we may view the cheese as flexible material; if a rectangular slice is pushed together horizontally from the two shorter sides it is likely to bend upwards in the middle, affording an easy grasping place.

Any real world situation can be represented at multiple levels of granularity, e.g. each object as a simple convex hull, or a complex detailed fine grained mesh model. To have a practical representation to work with we abstract at a suitable level. At any particular level of granularity we have multiple ways to abstract the situation, depending on what approximations we want to make, distorting some aspects (e.g. treat a curved surface as flat) and ignoring other aspects (e.g. a handle or inner space might not be relevant). Task-driven representation means that we allow the demands of the task to influence how we abstract the situation. We choose to see it in a way that facilitates the task we want to do.

Definition: In task-driven representation the models chosen to represent a scene depend on the task to be achieved, and can change as the need to perform a different task or subtask arises. The robot has a mechanism for creating and searching through different possible representations, to find one appropriate for the task.

This differs from the typical approach to robot perception where scenes are processed bottom-up to recognise the objects, or affordances present in the scene, without any influence from the task we are trying to achieve. However in humans the task can exert a top-down influence on visual processing [19]; we often stretch the boundaries of a category when we make effort to see an opportunity to apply a particular skill. The idea can be traced back to William James:

[1] A creative solution to a problem can often seem straightforward after the fact; the creative solution often brings a new perspective from which the solution follows straightforwardly, and it can be hard to recall the more limited perspective that existed before. This is similar to sparse images where it is difficult to see the object, but once you find the right grouping it is almost impossible to recall how you perceived it before (see first figure in [18]).

> The same property which figures as the essence of a thing on one occasion becomes a very inessential feature upon another. Now that I am writing, it is essential that I conceive my paper as a surface for inscription. [...] But if I wished to light a fire, and no other materials were by, the essential way of conceiving the paper would be as a combustible material. [...] The essence of a thing is that one of its properties which is so important for my interests that in comparison with it I may neglect the rest. [...] The properties which are important vary from man to man and from hour to hour. [20]

The idea is already present in computational models of analogy, for example in 'High level perception' it is argued that "Our perception of any given situation is guided by constant top-down influence from the conceptual level. Without this conceptual influence, the representations that result from such perception will be rigid, inflexible, and unable to adapt to the problems provided by many different contexts." [21]

C. An Implemented Example of the Approach

We have implemented task-driven representation in a robot vision system for assessing how to use a previously unseen object as a tool for a particular task [22]. Given a particular task, and a previously unseen point cloud of an object, our system can decide on the best way to abstract that object to identify a part as the handle, and another as the action part, for performing the task. The system can also be used to inform a full manipulation system on how to orient the tool and what point to bring in contact with a target [23]. For example for the hammering task we can see how different tools can be abstracted into parts in Figure 2.

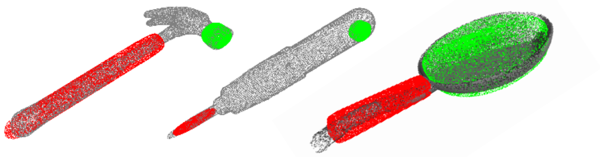


Fig. 2. Highlights of our "projection" technique being used to abstract objects for the task of hammering, to identify the best place to grasp, and the part that should hit a target. Red indicates the grasping part and green the action part. We see that when using a screwdriver for hammering (middle tool), the choice of grasping and action parts are very different to when the screwdriver is used for screwing.

Our system was trained to assess how good a tool is for five different tasks: hammering, lifting (e.g. a pancake), rolling dough, cutting, scooping. It has been tested on a test set of 3D scans of 50 household objects where ground truth was provided by humans. The system has outperformed a competing state-of-the-art system on this test set [22].

Our system works by fitting geometric shapes (superquadrics) into multiple randomly cut segments of a point cloud. Each pair of geometric shapes constitutes a candidate handle and action part; i.e. a possible way to abstract the tool. Each candidate is assessed by a task function to determine how good a tool with those parts would be for the task. The winning abstraction is the system's choice of the best way to use that object as a tool for the task.

The system can be seen in terms of bottom-up and top-down processes. Fitting geometric shapes is the bottom-up process in the perception, while the selection of a pair best for the task is the exertion of top-down pressure from the task. Our approach to the problem is inspired by Indurkhya's theory of 'projection' [24], meaning that the system is projecting what it wants to see in a particular tool when it assesses its effectiveness for a particular task. In Indurkhya's theory we would say that in Fig. 2 the concept of a hammer, and its organisation of parts (handle, hitting head), has been projected onto the other tools. In the example of the screwdriver it can be clearly seen how this top-down pressure causes a particular grouping of the low level elements of the data (i.e. the points of the point cloud). Certain points have been grouped together and 'seen' as a suitable hitting head (the green region) for no other reason than that they could serve the hammering task well. There are no local visual features in the region to single out those green points as special; it cannot be determined bottom-up.

To test in isolation the contribution of task-driven representation to our system's performance we ran an ablation study [22]. We ran one version of our system in a purely bottom-up fashion where the task did not influence the abstraction chosen to represent the tool, but rather the best fit superquadric was picked. The resulting system performed significantly worse in assessing the affordances of tools, relative to the ground truth (85% accuracy for the bottom-up system, vs. 91% for the task-driven system), and failed to outperform the baseline competitor system on two tasks.

The system has some limitations due to the particular way we have implemented projection: It will not suggest using a round stone as a hammer, because the system searches for a handle and a hitting end. The idea that a tool should have two parts has been hard coded into the system in violation of the verification principle [25], and becomes a weakness when moving beyond scenarios envisaged by the designers. Also, the system could be more efficient if it made more use of top-down information to guide the search, rather than randomly generating uniformly distributed candidates, and using the top-down pressure merely to select among these.

This implemented work is still a long way from the transferable library of support skills we have advocated in Sect. III, but it shows the outline of an approach to tackling it: robot perception needs to be able to take on board top-down input from the skill we want to apply, and use it to abstract a new situation in a way that facilitates application of the skill; i.e. abstract it such that there are components that can be mapped to the situation where the support skill has been used before.

V. Related Work and Neglected Issues

A. On Task-Driven Representation

The idea of using representations appropriate to the task is common in the literature. For example a work on folding clothing [26] assumes the cloth has infinite flexibility and zero thickness, so that it always falls flat and can be dealt with using 2D representations. On the other hand, when the task is flattening a garment, it is useful to represent the 3D surface in detail to determine the direction of major wrinkles and decide the direction to pull the cloth to flatten them [27]. Further examples of hand crafted representations appropriate to the task can be seen in reinforcement learning in robotics [28, Sec. 4.1]. The typical approach to designing a robotic solution for a task involves the human designer making a good choice of representation, which facilitates the task. We want the AI system itself to be able to select an appropriate representation at run-time. This is consistent with the verification principle, and in particular the principle of subjectivity [25]. The selection of an appropriate representation is present to a limited degree in work on manipulation planning which uses convex hull representations to simplify computation, but if these collide it can change representation to a more detailed mesh model [29]. However we would like the system to go further in creating representations that the designer might not have foreseen; we allow the system to borrow the representation used in a past scenario where a skill was successfully applied, and impose this representation on a new scenario.

One work which is very close to what we advocate involves warping point clouds from a source object to an unknown target object [30]. The warping approach can recognise key features of the target object, such as an aperture to pour into, or a rim point where liquid exits the pouring container, and the motor program can be adapted accordingly. This is close to our ideas because it shows how one can take an uninterpreted object (i.e. no prior abstraction has been given for the object, we just have sensor data, e.g. point cloud), and give it an interpretation that facilitates mapping an action to it.

Our approach has a similar idea to Tenorth and Beetz [31], where the robot can compute a symbolic view on sensor data as needed, but there is no single 'veridical' symbolic representation of the world that is maintained. Both our approaches share the idea of a particular representation (or symbolic view) of low level data being useful for a particular task, and being created and discarded as needed. We take the idea further in advocating that the robot itself would have a mechanism for creating novel representations of the data, under the influence of the demands of the task at hand.

B. *On Transfer and Models*

Our proposal to solve transfer is model-based in that the source scenario where a skill has been learned becomes a model to be imposed (top-down) on new target scenarios (or in our tool use example (Sec IV-C) the model is the knowledge of the geometric shapes and relations that make an effective tool for a particular task). Models allow for a symbolic type of reasoning which can explicitly consider model parts and their relationships, and understand what will happen when parts appear in different relationships, without needing to see that particular configuration in a training example. Lake et al. [7] argue that in order to build machines that learn and think like people, those machines need to use models to understand the world, plan actions, etc. They contrast model-building with the pattern-recognition approach of deep learning. Lake et al. argue that "human-level transfer learning – is enabled by having models with the right representational structure".

They give an example of learning in the computer game Frostbite: human players can transfer their general world knowledge and knowledge from previous video games to understand how to interact with various game animals, and ice. Transfer is mentioned many times in Lake et al. [7], and they describe a system to learn models for handwritten characters. However one thing not addressed is how to apply a model to a different domain; something which humans are clearly able to do, e.g. to see how knowledge from previous games might apply to the current one, or how knowledge about real world entities might be applied to the in-game entities. We believe that this cross-domain transfer requires a top-down process which can structure the lower level data from the new domain, to form appropriate abstractions which can map to the previous known models (Indurkhya describes this [24]).

Our proposal is in agreement with Lake et al., but we focus our attention more on this cross-domain transfer: how to apply a model to structure the abstraction of a new scenario in a way that allows transfer. The type of transfer we are concerned with has a component at the symbolic level, for example to decide which of several possible object parts (e.g. surfaces or apertures) should map to others from source to target. It also has a subsymbolic component when choices at the symbolic level must be instantiated sometimes by imposing an abstraction on vision data (what we called task-driven representation), or by adapting motion trajectories to move between object parts. This requires that we have compositional models of objects that can be explicitly reasoned with.

C. *Relationship to Affordances*

The 'skills' we have described could equally well be framed within the language of affordances [32], i.e. with an effect and behaviour, and in our case a search process to find the situations where they apply. They could also potentially be learned through exploration. Just like other work on affordances in robotics, we also aim to opportunistically exploit affordances in the environment as they arise. However, the typical approach to implementing affordances in robotics contrasts with part of our proposal: typically a vector of visual features determines whether or not an affordance is present, and this can be processed rapidly in a feedforward fashion. In contrast for our proposed task-driven representation it is computationally costly to determine where and how a particular skill can be applied, requiring a search involving interaction of top-down and bottom-up processes. We see no way to avoid this; there are many different ways in which a skill could be applied in a scenario; a reasoning step is required to consider the options. We do not think it feasible for an agent to enter a room and rapidly have all the affordances 'pop out', such that the agent is aware of them. E.g., in a workshop with a variety of tools and materials, consider all the ways in which objects can be put in a relationship, to exploit relational affordances, as well as the ways objects can be deformed: e.g., a nail can be bent to form a hook, a paper can be folded to form a container, operations include folding, tearing, gluing. It is more feasible that the robot approaches the scene with a task in mind, and selectively computes affordances likely to be relevant to that task. This is analogous to a recent model of human image interpretation [33] which has a bottom-up process making a rough guess of likely object classes, followed by a top-down process which attempts to validate the guesses. The top-down process uses a model including relations that are demanding to compute, and it is found that many relations are class-specific; hence it is efficient to only compute relations for the object class which is being tested.

A further contrast is that in typical affordance approaches the abstraction (from sensor data through to classification) is entangled with the assessment (how well it affords). This is effective for generalising to new examples in the neighbourhood of those in the training set, but not for significantly different creative applications of a skill. By separating abstraction (i.e. the models chosen to represent the scene) from assessment, we enable a different type of search through possible abstractions, which can consider more diverse transformations of the data, explicitly reasoning about alternative abstractions.

D. *Relationship to Planning*

Our proposal is similar in aim to earlier ideas of plan adaptation [34], [35]. However those works operate at a coarser level of granularity, where the sequence of planning steps is revised through additions, deletions or substitutions of steps. In contrast we focus on the adaptation of a single step (a step is the application of what we have called a 'skill'). Furthermore classical planning approaches abstract the world state with predicates, and treat this as an incontrovertible ground truth, whereas our approach emphasises the possibility of revisiting the world's data to find alternative descriptions.

Our 'skill' above is analogous to a planning operator. The precondition is flexible in that it can apply in a diversity of situations depending on how the situation is abstracted. The postcondition is more straightforward to describe as the effect achieved, e.g. the exposure of surfaces, or placement of an object flat on a surface. To decide if a skill is applicable in a state the system needs to look at the raw data of a scene, not a version abstracted at a high level. Therefore forward planning requires a simulator rendering expected future states (e.g. as point clouds), for task-driven analysis, to determine what skills could be applied. Backward chaining is more problematic because there is not a unique precondition of a skill. Approximate abstractions of the skills can be used at the abstract level for simpler planning, but this may miss opportunities to see new perspectives at the sensorimotor level (and hence apply skills). The solution is to conduct a planning process at both a low level (e.g. point cloud level) as well as an abstract level, with a tight integration and interaction between the processes (as done in Dornhege et al. [29]; see also Beetz et al. [36] where plans can unfold opportunistically, guided by perception of what the situation affords).

To fully exploit the ideas of task-driven representation in a planning system the ideas of top-down and bottom-up interaction should be extended to the planning system; we envisage an architecture similar to Copycat [37], but adapted to planning: Top-down processes would decompose a goal into a rough sequence of high level steps to achieve it. Bottom-up processes would analyse scenes in a task-driven way (the sub-goals being the tasks) and propose skills that could be applied. These skills are proposed steps in a workspace where plans are assembled. Partial fragments of plans are developed in the workspace to achieve subgoals, and assembled to produce increasingly complete candidate plans. The workspace also records the future simulated states resulting from plan fragments, for further bottom-up analysis. Candidate plans compete for selection and are ranked according to probability of completing the goal. If one promising partial candidate eventually (when developed further) leads to a low probability of success, the system backtracks to try the next best candidate.

E. *Physics*

Our proposed transferable skills implicitly embody physics knowledge because application of a skill is associated with an expected effect. A collection of these skills can make predictions about the effects of a variety of actions and could be seen as a model for the naive physics that an infant builds during exploration. I.e. each skill corresponds to a 'schema' and is refined and spins off new skills when unexpected effects happen (as described in schema-based account of development [13]). Further processes of representational redescription [13], as yet poorly understood, would need to generalise across skills to abstract increasingly sophisticated fragments of physical concepts such as force, momentum, friction, etc. In this way a physics model to support planning could be learned through exploration. However this would be a daunting challenge. A more pragmatic solution available at present is to simply code in physics knowledge with an accurate physics engine in the planner to predict the consequences of actions.

VI. Summary, Recommendation and Conclusion

Let us recap the three connected ideas from the Abstract: Firstly a manipulation robot should have a library of highly transferable support skills. These skills allow the robot to cope with the unforeseen, because their strong ability to generalise makes it highly likely that there is a skill that can apply to a situation to achieve a substep bringing the situation closer to the goal. Secondly the robot should have the ability to model a scene in several different ways. A robot that is built for one purpose might do well enough by modelling situations in only one way, as foreseen by the designer. For general purpose manipulation it will be necessary to model the same scene in different ways at different times, depending on the task at hand. Thirdly the robot should use the task to provide top-down guidance on how a scene should be modeled. This helps to direct the search through the many possible ways to represent the scene, and finds a representation that facilitates the application of a known skill, that achieves the task.

The approach requires that each skill in the robot's repertoire has an associated representation of the scenario where it works, and this representation can be 'projected' (or imposed) on new scenarios, producing novel abstractions of situations, that facilitate the application of manipulation skills in ways the designer might not have conceived. In this way each of the robot's skills can be seen as 'productive', in producing new ways to represent scenes. This helps the robot to opportunistically exploit affordances as they arise, because of its ability to shift perspective and see the scene in a different way. The approach also requires that there is some criterion of 'goodness'; i.e. how good a particular choice of representation is for a task. In our tool use example this 'task function' was learned from examples in simulation [22].

In conclusion task-driven representation offers a way to achieve human-like transfer, to cope with rare cases, and so to enable robots to work in less constrained, open environments.

References

[1] Stanford University, "Stanford personal robotics program." 2008, http://personalrobotics.stanford.edu/ , Accessed Dec. 2018, See especially "Robot Cleans a Room" https://youtu.be/oyHWkQcin7I.

[2] E. Davis and G. Marcus, "Commonsense reasoning and commonsense knowledge in artificial intelligence," *Communications of the ACM*, vol. 58, no. 9, pp. 92–103, September 2015.

[3] M. Ersen, E. Oztop, and S. Sariel, "Cognition-enabled robot manipulation in human environments: Requirements, recent work, and open problems," *IEEE Robotics & Autom. Magazine*, vol. 24, pp. 2–16, 2017.

[4] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," in *Robotics: Science and Systems (RSS)*, 2017.

[5] M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, J. Schneider, S. Sidor, J. Tobin, P. Welinder, L. Weng, and W. Zaremba, "Learning dexterous in-hand manipulation," *CoRR*, vol. abs/1808.00177, 2018.

[6] P. Agrawal, A. Nair, P. Abbeel, J. Malik, and S. Levine, "Learning to poke by poking: Experiential learning of intuitive physics," *CoRR*, vol. abs/1606.07419, 2016.

[7] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman, "Building machines that learn and think like people," *CoRR*, vol. abs/1604.00289, 2016.

[8] D. E. Wilkins and M. desJardins, "A call for knowledge-based planning," *AI Magazine*, vol. 22, no. 1, pp. 99–115, 2001.

[9] T. Fitzgerald, A. L. Thomaz, and A. K. Goel, "Human-robot co-creativity: Task transfer on a spectrum of similarity," in *International Conf. on Computational Creativity (ICCC). Atlanta, Georgia*, 2017.

[10] D. Hofstadter and E. Sander, *Surfaces and Essences: Analogy as the Fuel and Fire of Thinking*. Basic Books, 2013.

[11] G. Lakoff and M. Johnson, *Metaphors We Live By*, ser. Phoenix books. University of Chicago Press, 1980.

[12] F. Guerin, P. A. Ferreira, and B. Indurkhya, "Using analogy to transfer manipulation skills," in *2014 AAAI Fall Symposium*. AAAI, 2014.

[13] F. Guerin, N. Kruger, and D. Kraft, "A survey of the ontogeny of tool use: From sensorimotor experience to planning," *Autonomous Mental Development, IEEE Transactions on*, vol. 5, no. 1, pp. 18–45, 2013.

[14] L. Rat-Fischer, J. O'Regan, and J. Fagard, "The emergence of tool use during the second year of life," *Exp Child Psychol.*, vol. 113, no. 3, pp. 440–446, 2012.

[15] L. Rat-Fischer, L. Jeancolas, J. O'Regan, and J. Fagard, "Ability of infants to generalize from spoon use to tool use," in *International Congress of Infant Studies, New Orleans (USA), May 24-29th*, 2016.

[16] J. Fagard, R. A. Grupen, F. Guerin, and N. Krüger, "Mechanisms of Ongoing Development in Cognitive Robotics (Dagstuhl Seminar 13072)," *Dagstuhl Reports*, vol. 3, no. 2, pp. 55–91, 2013.

[17] N. Correll, K. E. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Okada, A. Rodriguez, J. M. Romano, and P. R. Wurman, "Lessons from the amazon picking challenge," *CoRR*, vol. abs/1601.05484, 2016.

[18] S. Dickinson, A. Levinshtein, P. Sala, and C. Sminchisescu, "The role of mid-level shape priors in perceptual grouping and image abstraction," in *Shape Perception in Human and Computer Vision*, S. Dickinson and Z. Pizlo, Eds. Springer, London, 2013, pp. 1–19.

[19] A. Harel, D. J. Kravitz, and C. I. Baker, "Task context impacts visual object processing differentially across the cortex," *Proceedings of the National Academy of Sciences*, vol. 111, no. 10, pp. E962–E971, 2014.

[20] W. James, *The Principles of Psychology*, ser. American science series. H. Holt, 1890, no. v. 1.

[21] D. J. Chalmers, R. M. French, and D. R. Hofstadter, "High-level perception, representation, and analogy: A critique of artificial intelligence methodology," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 4, no. 3, pp. 185–211, 1992.

[22] P. Abelha and F. Guerin, "Transfer of tool affordance and manipulation cues with 3d vision data," *CoRR*, vol. abs/1710.04970, 2017.

[23] P. Gajewski, P. Ferreira, G. Bartels, C. Wang, F. Guerin, B. Indurkhya, M. Beetz, and B. Sniezynski, "Adapting everyday manipulation skills to varied scenarios," in *IEEE International Conference on Robotics and Automation, Montreal*, 2019.

[24] B. Indurkhya, *Metaphor and Cognition*. Dordrecht, The Netherlands: Kluwer Academic Publishers., 1992.

[25] A. Stoytchev, "Some basic principles of developmental robotics," *IEEE Trans. Auton. Mental Development*, vol. 1, no. 2, pp. 122–130, 2009.

[26] S. Miller, J. van den Berg, M. Fritz, T. Darrell, K. Y. Goldberg, and P. Abbeel, "A geometric approach to robotic laundry folding," *I. J. Robotic Res.*, vol. 31, no. 2, pp. 249–267, 2012.

[27] L. Sun, G. Aragon-Camarasa, S. Rogers, and J. P. Siebert, "Accurate garment surface analysis using an active stereo robot head with application to dual-arm flattening," pp. 185–192, May 2015.

[28] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Rob. Res.*, vol. 32, no. 11, pp. 1238–1274, Sep. 2013.

[29] C. Dornhege, M. Gissler, M. Teschner, and B. Nebel, "Integrating symbolic and geometric planning for mobile manipulation," in *Safety, Security & Rescue Robotics (SSRR), 2009 IEEE International Workshop on*. IEEE, 2009, pp. 1–6.

[30] S. Brandi, O. Kroemer, and J. Peters, "Generalizing pouring actions between objects using warped parameters," in *2014 IEEE-RAS International Conference on Humanoid Robots*, Nov 2014, pp. 616–621.

[31] M. Tenorth and M. Beetz, "Representations for robot knowledge in the knowrob framework," *Artif. Intell.*, vol. 247, pp. 151–169, 2017.

[32] E. Şahin, M. Çakmak, M. R. Doğar, E. Uğur, and G. Üçoluk, "To afford or not to afford: A new formalization of affordances towards affordance based robot control," *Adaptive Behav.*, vol. 15, no. 4, pp. 447–472, 2007.

[33] G. Ben-Yosef, L. Assif, and S. Ullman, "Full interpretation of minimal images," *Cognition*, vol. 171, pp. 65 – 84, 2018.

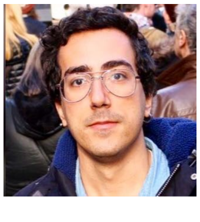
[34] S. Hanks and D. S. Weld, "A domain-independent algorithm for plan adaptation," *J. Artif. Int. Res.*, vol. 2, no. 1, pp. 319–360, Mar. 1995.

[35] H. Muoz-Avila and M. T. Cox, "Case-based plan adaptation: An analysis and review," *IEEE Intell. Systems*, vol. 23, no. 4, pp. 75–81, July 2008.

[36] M. Beetz, D. Jain, L. Mösenlechner, M. Tenorth, L. Kunze, N. Blodow, and D. Pangercic, "Cognition-enabled autonomous robot control for the realization of home chore task intelligence," *Proceedings of the IEEE*, vol. 100, no. 8, pp. 2454–2471, 2012.

[37] D. Hofstadter and FARG, *Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought*. New York: Basic Books., 1995.

Frank Guerin obtained Ph.D. degree from Imperial College, London, in 2002. Since 2003 he has been a Lecturer in Computing Science at the University of Aberdeen. He is interested in bringing ideas from Psychology to Artificial Intelligence, in order to understand how to implement human-like concepts in artificial systems.



Paulo Ferreira obtained Ph.D. degree from University of Aberdeen in 2018. During 2018 he has worked as a researcher at the University of Birmingham, and in 2019 moved to DELL EMC, Brazil. He is interested in robot vision, computer vision, and models of cognition.