FUNCTION BASED OBJECT CATEGORIZATION USING SIMULATION OF AN AGENT

EZER BAR-AVIV
FUNCTION BASED OBJECT
CATEGORIZATION USING SIMULATION
OF AN AGENT

RESEARCH THESIS

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE IN
COMPUTER SCIENCE

EZER BAR-AVIV

SUBMITTED TO THE SENATE OF THE TECHNION — ISRAEL INSTITUTE OF TECHNOLOGY
ADAR, 5769 HAIFA MARCH, 2009
THIS RESEARCH THESIS WAS DONE UNDER THE SUPERVISION OF PROF. EHUD RIVLIN IN THE DEPARTMENT OF COMPUTER SCIENCE

ACKNOWLEDGMENTS

I wish to express my thanks Prof. Ehud Rivlin for his guidance, support, help, and for urging me on when I needed it.

THE GENEROUS FINANCIAL HELP OF THE TECHNION IS GRATEFULLY ACKNOWLEDGED
Everything should be made as simple as possible, but not one bit simpler.

– Albert Einstein
Contents

Abstract 1

1 Introduction 3
  1.1 Proposed Solution ................................. 5
  1.2 Thesis Outline ............................... 6

2 Literature Review 7
  2.1 Geometric Models ............................... 7
  2.2 Parameterized Geometric Models .................. 8
  2.3 Structural Models ............................. 9
  2.4 Functional Approaches ......................... 12

3 Category Model 17
  3.1 Configuration Space ............................. 18
  3.2 Category Model Structure ....................... 18
  3.3 Limitations ..................................... 20

4 Object Categorization Framework 21
  4.1 Evidence Searching Algorithm .................. 23
  4.2 Using Collision Detection ....................... 25

5 Agents and Functionalities 27
  5.1 Virtual Human Agent ............................ 27
    5.1.1 The Category ‘Chairs’ ....................... 28
    5.1.2 The Category ‘Tables’ ....................... 29
    5.1.3 The Category ‘Beds’ ......................... 32
5.2 Virtual Palm Agent ........................................... 32

6 Results ......................................................... 37
   6.1 The Category 'Chairs' ................................. 37
   6.2 The Category 'Tables' ................................. 42
   6.3 The Category 'Beds' ................................. 46
   6.4 The Category 'Graspable' ......................... 47
   6.5 Sensitivity Analysis ................................. 49
      6.5.1 Functional Event ............................... 50

7 Conclusions and Future Work .............................. 52
   7.1 Conclusion ............................................. 52
   7.2 Major Contributions ................................. 53
   7.3 Future Work .......................................... 54

References .................................................. 57

Hebrew Abstract ............................................... i
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Part of 3D CAD DB for the category 'Chairs'</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>The SCERPO system of Lowe (Lowe, 1987)</td>
<td>9</td>
</tr>
<tr>
<td>2.2</td>
<td>An example for a parameterized geometric model</td>
<td>10</td>
</tr>
<tr>
<td>2.3</td>
<td>An example for structural model</td>
<td>11</td>
</tr>
<tr>
<td>2.4</td>
<td>Illustration of Biderman’s geons theory taken from Lowe (Biderman, 1987)</td>
<td>12</td>
</tr>
<tr>
<td>2.5</td>
<td>Part of the categories representation tree for the GRUFF system</td>
<td>13</td>
</tr>
<tr>
<td>2.6</td>
<td>The GRUFF system. Images taken from (Stark, Hoover, Goldgof, &amp; K., 1993).</td>
<td>15</td>
</tr>
<tr>
<td>5.1</td>
<td>Structure description of virtual human agent</td>
<td>28</td>
</tr>
<tr>
<td>5.2</td>
<td>Maximal configurations for the different categories recognized using a virtual human agent</td>
<td>28</td>
</tr>
<tr>
<td>5.3</td>
<td>Snapshots from the categorization process for the category 'Chairs'</td>
<td>30</td>
</tr>
<tr>
<td>5.4</td>
<td>Snapshots from the categorization process for the category 'Tables'</td>
<td>31</td>
</tr>
<tr>
<td>5.5</td>
<td>Snapshots from the categorization process for the category 'Beds'</td>
<td>33</td>
</tr>
<tr>
<td>5.6</td>
<td>Structure description of virtual palm agent</td>
<td>34</td>
</tr>
<tr>
<td>5.7</td>
<td>Major grasp categories</td>
<td>35</td>
</tr>
<tr>
<td>5.8</td>
<td>Snapshots from the categorization process for the functionality 'Light Wrap'</td>
<td>36</td>
</tr>
<tr>
<td>6.1</td>
<td>Chair categorization using simulation of embodied virtual human agent</td>
<td>38</td>
</tr>
<tr>
<td>6.2</td>
<td></td>
<td>39</td>
</tr>
<tr>
<td>6.3</td>
<td>Objects that weren’t classified as ‘Chairs’</td>
<td>39</td>
</tr>
</tbody>
</table>
6.4 Part of the real-chairs DB which was generated using single viewpoint scanning ........................................... 41
6.5 Part of 3D CAD DB for the category 'Tables' ................................. 43
6.6 Results for the category 'Tables' ............................................. 44
6.7 Some of the objects that weren't classified as 'Tables' ............... 45
6.8 Part of the real-tables DB, which was generated using single viewpoint scanning ........................................... 46
6.9 Results for the category 'Beds' ............................................. 47
6.10 Part of the DB of the categories 'Tools' and 'Glasses' ............... 48
6.11 Results for the action 'Light Wrap' as part of the classification of the categories 'Tools' and 'Glasses' ......................... 49
6.12 Two trajectories on top of a 3D chair model which were used to produce 2.5D models from different views .................... 50
6.13 Sensitivity analysis for 2.5D view-points ............................... 50
7.1 Sub-categorization of 'Tables' ............................................ 54
7.2 Sub-categorization for 'Table for one' ................................. 55
7.3 Sub-categorization for 'Table for pair' ............................... 55
7.4 Sub-categorization for 'Table for family' ............................ 56
List of Tables

5.1  Functionality definition for the category 'Chairs' . . . . . . . . . . 29
5.2  Functionality definition for the category 'Tables' . . . . . . . . . . 31
5.3  Functionality definition for the category 'Beds' . . . . . . . . . . . 32
5.4  Functionality definition for 'Heavy Wrap' . . . . . . . . . . . . . 34
5.5  Functionality definition for 'Heavy Wrap' . . . . . . . . . . . . . 34
Abstract

Encouraged by recent cognitive psychology studies, this research presents a novel functional approach for the categorization of 3D objects. The work presents a new approach for representing functionalities by means of agent and functions, in harmony with motor cognition psychology. By embodying the receiver of the functionality within a virtual environment and simulating its interaction with a candidate object, functional properties of the object can be revealed and validated. We present a generic search algorithm, which is used during the simulation process, designed to search the configuration space looking for evidence configurations that prove the fulfillment of the functionality. The algorithm involves two steps - searching for semi-functional configurations and relaxation. The first step is targeted to accelerate the search in the complete configuration space of both global-DOFs and inner-DOFs via a unique pruning. This pruning is inspired by cognitive motives and implemented by means of collision detection. The second step is the relaxation phase which simulates the action performed by the agent upon the candidate object. The action is represented using a sequential movements of the inner-DOFs of the agent until the goal configuration is reached.

The use of an agent in the categorization process enables us to overcome the shape-diversity of objects that belong to the same category and to handle noise and self occlusions that occur in range scans. To demonstrate the robustness of the approach we have implemented the categorization algorithm for several categories and functionalities. We have tested it on a large database of both 3D-CAD graphic models and real 3D scanned data generated from a single viewpoint. Sensitivity analysis shows that the concept is applicable and robust enough to be used in real applications.
Abbreviations and Notations

$C$ Category

$A$ Virtual 3D Agent

$f$ Functionality

$con$ Configuration Within the Configuration Space
Chapter 1

Introduction

Object categorization is considered one of the most important problems in computer vision. The difficulty originates from the lack of applicable definition for category models, one that can handle the huge diversity in shape of objects that belong to the same category (see Figure 1.1). The functional paradigm addresses this difficulty by concentrating on the way the object is being used rather than on its shape. We have taken this approach one step further: a category model is defined as a set of functionalities that can be exposed during a simulation in a virtual environment. The simulation imitate the interaction between a candidate object and a set of different agents defined within the category model.

The idea for this approach was driven from the domain of motor-cognition psychology, which investigates both the way actions are being performed (planned, prepared and executed) and the way they contribute to the representations of objects. One example is the widely studied action of grasping (Jeannerod, 1986) in which most of the fingers movements towards stable grasp occur way before contacting the object. The results imply that the representation of objects encodes relevant properties for potential interaction with the acting agent. Other studies suggest we can consciously simulate or imagine actions we do not execute. The authors of (I., Toni, Decety, Gregoire, & Jeannerod, 1997), for example, have compared the pattern of cortical activation during two different tasks: actions and perceptual analysis, in which no action occurs. The experiments revealed activation of homologous parts in both tasks, parts that pertain to the dorsal pathway,
Figure 1.1: Part of 3D CAD DB for the category 'Chairs'
which is believed to be connected to actions. The use of resources that are connected to actions during perceptual analysis means that we use simulation during object-oriented actions. Moreover, other experiments in motor-imagery (Sirigu, Duhamel, Cohen, Pillon, Dubois, & Agid, 1996; Parsons et al, 1995) imply that we perform unconscious simulation of actions before every task. In those experiments subjects are asked to answer questions regarding virtual tasks. As shown in (Parsons et al, 1995) for example, the response times for tasks relating to the arm have increased with the difficulty of the task and reflected biomechanically compatible trajectories. Meaning that the subjects have unconsciously simulated the movement before giving the response.

A similar pattern of cortical activity in action-related areas is found even in the implicit cases of observing tools or even hearing action verbs (Perani, Cappa, Bettinardi, Bressi, Gorno-Tempini, Matarrese, & Fazio, 1995). Moreover, the simulation theory (Rizzolatti et al, 1996) suggests that we understand the action performed by others simply by simulating those actions. Studies that have investigated the pattern of activation during observation of actions (Gallese & Goldman, 1998) support the idea that actions performed by others can be understood to the extent that they can be simulated by the observer. This idea was encouraged by the discovery of the mirror-neurons in monkeys (Rizzolatti et al, 1996, 2004) and later on in humans (Rizzolatti et al, 1996; Decety et al, 1997). Mirror-neurons respond when a monkey executes certain kinds of actions or when it perceives the same actions being performed by another monkey. Most mirror-neurons selectively respond to only one type of action while others respond to two or even three different types of observed actions (Ferrari, Gallese, Rizzolatti, & Fogassi, 2003). The mirror-system discovered in humans is also believed to be the area that allows us to replicate action performed by others.

### 1.1 Proposed Solution

This paper integrates those ideas into a new functional approach. This approach emphasizes the connection between category instances and the acting agents benefiting from their functionality, in harmony with motor-cognition psychology.
According to the functional paradigm, a category is defined by certain functionalities. We have chosen to assimilate within the category model, one virtual agent for each functionality. These agents will be used to verify the existence of each functionality using simulation of actions.

As suggested by the simulation theory, the category model encodes properties that relate to the simulation of actions performed by the acting agents upon category instances. The encoded data includes the 3D graphical models of the agents and other input variables for the generic interaction algorithm. These variables control the interaction process, between an agents and candidate object within the virtual environment.

We hypothesize that in some scenarios, when humans are asked to classify an object, they perform unconscious simulation process of searching evidence configurations that demonstrate the functionality of that object. The interaction algorithm mentioned earlier searches the configuration space looking for such evidence configurations. In a sense, those evidence configurations represent the best way to interact with the object in order to benefit from its functionality. Each evidence configuration receives a score according to measure of functionality fulfilment in this configuration. The lower the score is, the more said configuration can be perceived as improvisation.

1.2 Thesis Outline

The rest of the work is structured as follows. In Chapter 2 we review related works. Chapter 3 present the structure of a category model and Chapter 4 presents the categorization framework. Chapter 5 investigates several agents and functionalities in the spirit of the new approach. Chapter 6 presents experimental results for the categorization of both CAD objects and real-objects of different categories. The real data objects were scanned from a single-view point and require the ability of handling occlusions and noise. In said chapter we also investigate the limits of our approach, and analyze its sensitivity. We conclude with Chapter 7.
Chapter 2

Literature Review

Visual recognition of objects is one of the most challenging problems in computer vision and artificial intelligence and works on this topic go back to the 1960’s. Accordingly, there is an extensive body of literature on the subject. Reflecting the historical development of the field, we first review the literature on single object recognition using geometric models, and then explore the different works on object categorization. These works include variety of methods such as the parameterized geometric models, the structural models and the function based models.

2.1 Geometric Models

Early works have concentrated on finding specific instances of objects. This problem is somewhat simplified by not having to consider intra-class variability. Variability in appearance is reduced to five sources: viewpoint change; differences in illumination; occlusion; background clutter and imaging noise. A representative approach is the geometric models representation. In this approach, the input is directly matched to a model of low-level geometrical features, such as interest points, surfaces and edge contours extracted from the interior and exterior of the object. The use of these features is convenient since it tackles two of the variability sources: it is insensitive to illumination changes and makes the determination of 2D or 3D pose relatively straightforward.
An important contribution was made by the papers of Lamdan and Wolfson (Lamdan, Schwartz, & Wolfson, 1988) and Rigoutsos and Hummel (Rigoutsos & Hummel, 1995), in which object models consist of sets of interest points. The sets are made invariant to affine transformations by using three points from within the set as a basis. Following, Rothwell (Rothwell, Forsyth, Zisserman, & Mundy, 1993; Rothwell, Zisserman, Forsyth, & Mundy, 1995), describe a recognition system for planar objects using projective invariants. A projectively invariant set of index functions is used to represent each object. Each index function is based on the geometrically invariant properties of a small group of points, lines or conics.

A slightly different approach is the one known as the alignment techniques. Typical papers include the work of Huttenlocher and Ullman (Huttenlocher & Ullman, 1987) which recognizes 3D objects using a model obtained from an image, the work of Ullman and Basri (Ullman & Basri, 1991) which represents a 3D model with a mixture of 2D models and matches it to the image using lines and points, and the SCERPO system of Lowe (Lowe, 1987) which extracts lines from a query image. Results for the recognition of razors using the SCERPO system are presented in Figure 2.1

2.2 Parameterized Geometric Models

A parameterized geometric model is created by replacing some of the constants in a geometric model with parameters that may be constrained to lie within defined range. Figure 2.2 illustrates the parameterized geometric model for the category 'Automobile', together with some variations in the category, as described in the work of Kanodo et al (Kadono, Asafa, & Shitai, 1991). More sophisticated parameterizations capture "object families" by allowing substantial shape variation between object instances. The work of Grimson (Grimson, 1987) is an example for such an approach.
Figure 2.1: The SCERPO system of Lowe (Lowe, 1987). (a) The original image of a bin of disposable razors; (b) Straight line segments derived from image; (c) The 3D wire-frame model of the razor shown from a single viewpoint; (d) Successful matches between sets of image segments and particular viewpoints of the model; (e) The model projected onto the image from the final calculated viewpoints.

2.3 Structural Models

The structural model specifies an explicit construction of the object as a set of parts. The model may be hierarchical, meaning that each part can be further reduced to primitive geometric descriptions. Figure 2.3 illustrates an example, taken from the work of Mulganokar et al (Mulgaonkar, Shapiro, & Haralick, 1984), in which a category model is described using three primitive part types: sticks, plates and blobs. The categorization involves matching of the different part, together with the connections between them.

An important work is Biderman’s recognition-by-components (RBC) theory (Biderman, 1987). Biderman suggested that objects can be decomposed
Figure 2.2: An example for a parameterized geometric model

into primitive 3D shape called "geons", much like the way any word can be decomposed into phonemes. The full family of geons has 36 members, and some of them are demonstrated in Figure 2.4. Categorization could then be achieved using a form of indexing based on structural composition of the object.

In a parameterized structural model, categories are modeled by parameterizing the location and number of some of the model components. One example is Brooks ACRONYM system (Brooks, 1984).

Recent works used probabilistic category models and employed learning algorithms to learn the model parameters. Different works vary widely on the way parts are detected and represented and on the way learning is employed. For
instance (Fergus, Perona, & Zisserman, 2003) models categories as flexible constellation of parts and uses a scale invariance feature-detector, while (Agarwal & Roth, 2002) represents images of the training set using a vocabulary that is automatically constructed from a set of sample images of objects that belong to the category of interest. Learning object category often requires huge training set. Recent works have tried to face this problem in various ways, for instance by extracting knowledge from recently learned categories (Li, Fergus, & Perona, 2006).

Figure 2.3: The structural model of the category 'Automobile', taken from (Mulgaonkar et al., 1984). The model is represented using three part types: sticks, plates and blobs.
2.4 Functional Approaches

According to function-based approaches, classification should refer to the functional description of the object rather than to the structural one. The beginning of the function-based categorization was the work of some AI and computer vision researchers who understood the potential of the approach. Winston (Winston, Binford, Katz, & Lowry, 1983) propose a theory explaining how physical models are learned and identified using functional definition. DiManzo (DiManzo, Trucco, Giunchiglia, & Ricci, 1989) present the FUnctional Reasoning project (FUR), which presents several primitive functions, and several functional expert concepts for identification of functional primitives. Connell and Brady (Connell & Brady, 1985) presented a system that builds semantic network description from image input and learns category concepts. Vaina and Jaulent (Vaina & M.C., 1991) described a functional-based approach which uses fuzzy-logic and computes the compatibility between an object shape and the functional requirements.
Figure 2.5: Part of the categories representation tree for the GRUFF system
One of the most influencing works in the function-based arena was the one of Bowyer and Stark (Stark et al., 1993; Stark & K., 1994, 1996) with the GRUFF and OMLET systems for 3D objects categorization under the domains of furniture, dishes and hand tools. Each category is defined using a set of functional properties, for instance seat support and back support. These properties were defined using knowledge primitives that imply about shape, for instance the dimensions, relative orientation, and proximity of surfaces. An efficient indexing of the knowledge base is proposed where during the recognition process an accumulated association measure reflects the systems confidence in support of the shape belonging to the hypothesized category, allowing different interpretations of a shape to be rank ordered.

Other than the GRUFF, works on categorization have evolved in different directions. Bajcsy and Solina (Solina & Bajcsy, 1987) propose a modeling system for generic objects, consisting of a prototype made up of parts based on the psychological notions of categorization suggested by Rosch (Rosch, 1977; Rosch & Lloyd, 1978). Rivlin (Rivlin, Dickinson, & Rosenfeld, 1995) presented a theory of function-based recognition that is a natural extension of the part-based shape recognition approach. The work have tried to reason about the functionality of object’s parts. A hummer, for instance, is recognized by a handle and a striking surface. Following, Froimovich et al (Froimovich, Rivlin, & Shimshoni, 2002) propose a system for function-based classification that assumes a priori knowledge of the objects and performed on range images of real 3D objects. Duric (Duric, Fayman, & Rivlin, 1996) have attempted to determine functionality from motion. Peursum (Peursum, Venkatesh, West, & Bui, 2004) concentrated in analyzing sequences of images rather than static 3D models, trying to understand the interactions of human with its environment. Amant (Amant & Wood, 2005) analyze the activity of agents as a user of tool, and Dickinson (Dickinson & Keselman, 2005) propose a generic model in which inferences are drawn from examples.

A key idea in the early works of functional object categorization was to decompose each functionality to a series of rather simple geometric measurements applied to the object’s parts. The GRUFF system mentioned earlier, for example, requires the existence of two surfaces acting as seat support and back support.
Figure 2.6: Demonstration of the GRUFF system during the categorization of armed chair. (a)-(d) Present the necessary primitives for the functionality “enable sitting” and “provide stable surface”, and the corresponding surface. Altogether, the object can now be categorized as conventional chair; (e)-(h) Present the necessary primitives for the functionalities “provide back support” and “provide arm support”; Images taken from (Stark et al., 1993).
in order to categorize an object as a chair. There are also limitation on the di-
mensions of those surfaces and the relations between them, as they should allow
sitting. In a sense, part of the shape diversity is translated into wide range of
allowed angles and dimensions. In addition, separating the object background
and segmenting it to its basic parts is not a trivial task.

Our work extends the familiar function-based paradigm. By embodying the
receivers of the functionality as an agent within a virtual environment, it is pos-
sible to simulate the actions performed by the agent upon the candidate object
and validate object’s functionality. The use of an agent encapsulate most of the
knowledge primitives mentioned earlier, since finding evidence configuration ful-
fishes all those low-level geometric tests, by definition. By imitating the way we
hypothesize humans perform categorization, we can directly categorize objects
without the need of pre-segmentation or complex shape model of categories.
Chapter 3

Category Model

One can look at a category as a set of functionalities and properties. While property usually represent physical condition that is easy to verify (e.g., round, stable, etc.), functionality represent the essence of the category. According to the proposed approach, functionality can be exposed, or validated, by involving its receiver in the categorization process. The receiver shall be referred to as the agent.

Considering a category with $n$ functionalities $\{f_1, f_2, \ldots, f_n\}$, the category model should contain $n$ corresponding 3D virtual agents $\{A_1, A_2, \ldots, A_n\}$ whose purpose is to verify these functionalities. It can be achieved by simulating the interaction between the agents and a candidate object, trying to find evidence-configurations in the agents’ configuration space that expose these functionalities. Instead of using unique searching algorithm for each agent (i.e., for each functionality), we propose a generic scheme that suits many different categories, with only a small number category-specific inputs and predicate functions to be used by the searching algorithm. The scheme avoids exhaustive search in the configuration space by including a cognitively-motivated algorithms during the simulation of the virtual-agents and candidate-object interaction. The simulation takes place within a designated virtual environment.

To conclude, a category should contain the specific agents, some initial conditions and unique inputs for the relaxation process and predicate (or indication) functions to be used by the searching/interaction algorithm, as will be explained shortly.
3.1 Configuration Space

Every agent has at least six degrees of freedom (DOFs). The first six DOFs referred to as the global-DOFs and pertain to the global rotation and translation of the agent in the virtual environment. In addition, an agent may have additional inner-DOFs which are derived from its inner structure. Generally, the configuration of an agent with \( k \) inner-DOF’s can be described as a point in the \((k+6)\)-D configuration space:

\[
\text{con} = (\text{tran}_x, \text{tran}_y, \text{tran}_z, \text{rot}_x, \text{rot}_y, \text{rot}_z, \text{in}_1, \text{in}_2, ..., \text{in}_k)
\]

(3.1)

3.2 Category Model Structure

A category model is defined as follows:

\[
C = < (A_1, I_1), ..., (A_n, I_n), P_1, ..., P_m >
\]

(3.2)

where \( \{A_i\}_{i=1}^n \) are the different agents meant to verify the fulfillment of the corresponding functionalities \( \{f_i\}_{i=1}^n \), \( \{I_i\}_{i=1}^n \) are inputs for the searching algorithm and \( \{P_i\}_{i=1}^m \) are the physical properties of the category. The assembly of agents and inputs to pairs is not accidental and relates to the idea that functionality can be expressed by means of agent and action, as will be explained shortly. We use a generic two-phase searching algorithm which is common to all categories. This algorithm, which is discussed in Section 4, requires some inputs which are category specific. Some of these inputs are used during the first phase of the algorithm, which is known as the \textit{pruning process}, while others are used during the second phase, which is known as the \textit{relaxation process}. During that process the inner-DOFs of the agent are iteratively changed towards an evidence configuration.

To conclude, each functionality is described via the following five parameters:

\textbf{Agent} - A 3D model of the receiver of the functionality. The virtual agent \( A_i \) is needed to verify the existence of the corresponding functionality \( f_i \). Embodying the agent’s model encapsulates the various knowledge primitives (Stark & K., 1996) and ensures that the object can fulfill the functionality.
Maximal configuration - An input for both the pruning and relaxation processes. It is actually the initial configuration of the agent. In a way, it expresses a cognitive-insight, by representing an initial inner configuration that can lead to almost any other inner configuration using only steepest-descent-like iterative process which is referred to as the relaxation process.

Anchor predicate - An input for the pruning process. The anchor predicate function indicates semi-functional configurations, from which the relaxation process can take place. The anchor predicate is actually a list of surfaces $P_S = \{surf_1, ..., surf_j\}$. Its target is to identify contact between the candidate object and said key-surfaces of the agent, which are related to the level of functionality fulfillment. The anchor predicate is mainly used to simplify the searching process and avoid exhaustive search for evidence-configurations. It allows us to perform functional-pruning of the configuration space, as will be explained later.

Relaxation list - An input for the relaxation process. As mentioned, the relaxation process takes place when the anchor predicate is satisfied and involves a steepest-descent-like process of several inner DOFs of the agent. When the relaxation process ends, the goal predicate function is called to test whether an evidence configuration that fulfills the functionality was reached. For an agent with $k$ inner-DOFs the relaxation list is actually a list of pairs $\{(DOF_7, p/n), \ldots, (DOF_{k+6}, p/n)\}$; inner-DOFs and direction (positive or negative) that determines the activation order and direction of the various inner-DOFs within the steepest descent process of the relaxation algorithm.

Goal predicate - An input for the relaxation process. The goal predicate indicates on fully-functional configuration in which the functionality is fulfilled by the corresponding agent. Though there are levels of fulfillment, it is actually the evidence configuration that we are looking for. Much like the anchor predicate, the goal predicate is is actually a list of surfaces $P_F = \{surf_1, ..., surf_h\}$, targeted to identify contact between the candidate object and those key-areas of the agent. The set of key-areas in the case of the goal predicate include those of the anchor predicate as well as other key-areas.

According to the above definitions, input $I_i$ will be defined as follows:

$$I_i = (con_i, P_{si}, L_i, P_{fi})$$  \hfill (3.3)
Where $con_i$ is the maximal configuration input, $P_{Si}$ is the predicate for semi functional configurations, $L_i$ is the relaxation list and $P_{Fi}$ is the predicate for fully functional configurations.

### 3.3 Limitations

While most categories can be inferred from a single static pose of the candidate object, some categories require dynamic pose (e.g., in the case of scissors). We refer to these categories as *static-category* and *dynamic-category* respectively. In this paper we will refer only to static categories, in which the functionality is obvious, for example man-made objects that were designed to fulfill certain functionalities. In addition, this paper focuses on verifying only the functionalities categories, while ignoring the physical properties, which should be addressed differently.
Chapter 4

Object Categorization Framework

According to the functional approach, a category is represented by a set of functionalities that express the essence of the category. Our approach relates to the unique way in which the existence of these functionalities is verified. Motivated by motor-cognition psychology, arguing that humans perform off-line simulation of action before every task, we came with the idea that actions can play critical role in the representation of functionalities. In order to do so, the actions must be performed by the potential receiver of the functionality, upon the candidate object, needed to be categorized. Using unique virtual environment, we have managed to simulate actions of different agents, hence, mimic the interaction between them and given candidate objects.

Our categorization framework requires a unique category-model for each category it should be familiar with. A category model includes, among others, pairs of agents and actions, where the different pairs are merely representation of different functionalities. The actions are unambiguously defined by the combination of the maximal configuration, relaxation list and goal predicate. These three define a series of inner-DOFs movements of the agent, hence simulating the performed actions. The categorization process requires one simulation for each functionality of the category.

All simulation processes are carried out by out two-phase searching algorithm that includes a pruning process and a relaxation process. The intention of this
algorithm is to find evidence configurations within the agent-object configuration space that fulfill a certain functionality to some extent. This information is crucial and helps in distinguishing between fully functional configurations and ones which are merely an improvisation.

The pruning process basically implements a cognitively-motivated pruning heuristic that enables efficient search within the 6D global-DOFs configuration space, looking for semi-functional configurations. The semi-functional configurations together with the maximal configuration and relaxation process, create situation in which any fully-functional configuration is reachable through a unidirectional iterative process, enabling the search to be independent in the number of inner-DOFs.

The second phase of the searching algorithm, the relaxation process, implements said unidirectional process that iteratively changes the inner-DOFs of the agent towards fully functional configurations. This phase actually simulate the action performed by the agent, upon the candidate object. Both the activation and termination of the action simulation are triggered by means of graphical collision detection techniques, as will be explained later.

Let us examine the example of the category ‘Glass’. Glass is characterized by the two functionalities - ‘graspable’ and ‘container’, and the property ‘stable’. Classifying an object as a glass involves simulating object interaction with two different agents, one for each functionality, followed by immediate calculation of object’s stability (e.g., by means of calculating the center of mass). The functionality ‘graspable’, for instance, can be represented by the agent ‘virtual-palm’ and an action that implements grasping. For that agent, the maximal configuration is defined as one in which the virtual-palm is wide open. In a sense, any grasping configuration is reachable from that configuration, using only simple steepest-descent like iterative process which we called the relaxation process. The anchor predicate, in this case, gives indication of contact between the candidate object and certain surfaces in the center of the virtual-palm. Any configuration which lacks to do so, can never lead to a final grasping configuration and therefore can be pruned in advance. From that point, relaxation process can take place, towards a final configuration. In our case the relaxation process is simply an iterative closing of the fingers (unidirectional movements of inner-DOFs) until a
dead-end configuration is reached. This configuration is later tested using the
goal predicate, verifying that a full contact between the surfaces of the fingers
and the center of the palm was reached as well.

4.1 Evidence Searching Algorithm

We now introduce the general algorithm for object categorization. Given an
object \( Obj \) and a category \( C \), Algorithm 1 determines whether the object is an
instance of the category and provides a set of evidence configurations for each
of the class functionalities. The algorithm tries to find evidence configurations
for each one of the functionalities, by placing the corresponding agent in its
maximal configuration within the virtual environment and then searching for
said evidence configuration. As mentioned, the search involves both pruning

\[
\text{Algorithm 1 : Is_Instance_Of}(Obj, C)
\]

\[
F \leftarrow \emptyset
\]

\[
\text{for all } (A_i, I_i) \in C \text{ do}
\]

\[
\text{Init}_\text{Virtual}_\text{Scene}(A_i, Obj)
\]

\[
\text{Set}_\text{Agent}_\text{Con}(A_i, con_i)
\]

\[
S_{\text{temp}} \leftarrow \text{Pruning}_\text{Process}(Obj, A_i, P_{S_i})
\]

\[
S_i \leftarrow \emptyset
\]

\[
\text{for all } con \in S_{\text{temp}} \text{ do}
\]

\[
\text{Set}_\text{Agent}_\text{Con}(A_i, con)
\]

\[
\text{final} \leftarrow \text{Relaxation}_\text{Process}(Obj, A_i, L_i)
\]

\[
\text{Set}_\text{Agent}_\text{Con}(A_i, \text{final})
\]

\[
\text{if } \text{Test}_\text{Predicate}(A_i, P_{F_i}; \text{final}) \text{ then}
\]

\[
S_i \leftarrow S_i \cup \{ \text{Grade}(\text{final}), \text{final} \}
\]

\[
\text{end if}
\]

\[
\text{end for}
\]

\[
\text{if } S_i = \emptyset \text{ then}
\]

\[
\text{return } (FALSE, \emptyset)
\]

\[
\text{end if}
\]

\[
\text{final}_i \leftarrow \text{Get}_\text{Highest}_\text{Grade}(S_i)
\]

\[
F \leftarrow F \cup \text{final}_i
\]

\[
\text{end for}
\]

\[
\text{return } (TRUE, F)
\]
process and relaxation process. The configurations found during the pruning process are the initial configurations for the relaxation process, which does not always end in a fully functional configuration. The final configurations are tested using the goal predicate and then graded. Since the grade is proportional to functionality level, we use a threshold to filter non-valid configurations with less than the minimal required functionality. The valid configurations are actually evidence configurations with functionality level that can vary from improvisation to full functionality. The highest grade configuration is then inserted into the set of evidence configurations $F = \{final_i\}_{i=1}^m$. Notice that this set represents the configurations that best demonstrate each functionality $f_i$, hence, enabling to determine the best position in order to fulfill certain functionality. For instance, it can recommend the best way to grab an object, or the the most suitable way to sit on a chair.

The pruning process is described in Algorithm 2. It first finds the bounding box of the object, quantizes it according to a certain grid spacing of global-DOFs, and defines the 6D configuration space by extracting all valid configurations. This way, we avoid exhaustive search of the inner-DOFs, and assure that the search is independent of the agent's inner structure. Next, each configuration is tested using our anchor predicate, looking for semi-functional configurations. The test involves two collision detection queries, trying to identify contact between the object and certain surfaces of the agent, as will be described during the next subsection. The only input for the test, beside the agent and the object, is the set of surfaces (namely, pointers for agent surfaces) given by $P_s$. For instance, the virtual-human model in Figure 5.1 contains several red areas that imply on back-support, seat-support etc. In order to distinguish between contact and collision we verify that the agent and the object do not collide while the agent and an extended version of the object (by $\varepsilon$) do collide. We call this situation $\varepsilon$-contact. The output of the relaxation process is a list of agent configurations, in which a contact of functional importance was made with the object.

The relaxation process is described in Algorithm 3. The main input for the algorithm is the relaxation list $L$ that defines the activation order for the agent's inner-DOFs. It also defines the directions in which each of the inner-DOFs are shifted. During the process, the inner-DOFs are shifted one by one, where each
such shifting is actually a series of movements in a single direction. When collision is identified, it means the current inner-DOF has reached local minima and the process continues to the next inner-DOF. Notice that this is a very quick process, linear in the number of inner-DOFs.

The combination of searching for semi functional configurations in a 6D configuration space and a quick linear process from each one of them, prevents exhaustive search in the much larger configuration space of both global-DOFs and inner-DOFs. The two processes are identical due to the fact that every fully functional configuration is reachable from a certain semi-functional configuration using the said relaxation process.

### 4.2 Using Collision Detection

Pruning process involves two collision detection queries for each configuration. The first one is meant to eliminate configurations which are not collision free. The second involves $\varepsilon$-contact test between the agent and the object, whose purpose is to test whether some parts of the agent, that imply of semi-functionality, are close enough to the object to be considered as contacting it. The $\varepsilon$-contact test is achieved by extending the 3D-object model by $\varepsilon$ and checking whether it collides
Algorithm 3: Relaxations Process$(Obj, A, L)$

for all $(DOF, dir) \in L$ do
    $flag = TRUE$
    $con_{new} \leftarrow \text{Get\_Agent\_Con}(A)$
    while $flag$ do
        $con_{old} = con_{new}$
        $con_{new} \leftarrow \text{Change\_Con}(con_{new}, DOF, dir)$
        Set\_Agent\_Con$(A, con_{new})$
        if not Collision\_Free$(Obj, A)$ then
            $flag = FALSE$
        end if
    end while
    Set\_Agent\_Con$(con_{old})$
end for
return $con_{old}$

with the agent. Notice that the extended object could be defined only once during
the categorization process.

Our collision detection implementation has relied on a bounding volume hier-
archy (BVH) model representation using axis-aligned bounding boxes (AABB’s).
Specifically, we have used the SOLID library (Van den Bergen, 1997) for collision
detection of three-dimensional objects undergoing rigid motion and deformation.
SOLID allows quick update of the BVH as the model is deformed and is espe-
cially suited for collision detection of objects described in VRML, such as the
ones which we use.
Chapter 5

Agents and Functionalities

Standard functional approaches perform many low-level functional tests to verify that a given object can supply certain functionality. For example, the surfaces of a chair should be of a certain area and with a certain angle between them, to allow sitting. The model of a virtual human agent conceals this low-level functional information, since only appropriate sitting areas and appropriate angle between them can provide decent sitting for the virtual human agent. Verifying that a given object is indeed a chair, comes down to finding a sitting configuration of a virtual-human agent on that object. Such configuration satisfies, by definition, all necessary low-level tests. In this chapter, we present two such agents: a virtual human agent and a virtual-palm agent. As mentioned, we express functionality as agents and action and therefore, with different actions, certain agent can reveal different functionalities, thus participate in the categorization process of different categories.

5.1 Virtual Human Agent

In general, a category is represented by several functionalities, each is represented by an agent and an action. We now present the virtual-human agent, which is part of the model of the categories 'Chairs', 'Tables' and 'Beds', and the specific actions for each of these categories. Since these categories are represented by only one functionality, the category model is unambiguously defined by that specific action (and the physical property 'stable').
Figure 5.1: Virtual human agent. The figure describes the agent’s joints (inner-DOF’s) and agent’s surfaces. The joints are labeled as $J_n$ and the surfaces are numbered. From left to right (a) front of the agent, (b) back of the agent.

Figure 5.2: Maximal configurations for the different categories recognized using a virtual human agent. From left to right: (a) For the category 'Chairs', (b) For the category 'Tables', (c) For the category 'Beds'.

5.1.1 The Category 'Chairs'

The category 'Chairs' is represented by the functionality 'sittable' (or 'provides sitting'). This functionality is exposed when the right action is triggered by the receiver of the functionality upon a category instance. For the category 'Chairs', this agent-action pair, that represents functionality, was chosen to be a virtual
human agent performing an action of sitting.

Figure 5.2a presents the maximal configuration for the category 'Chairs'. Given an object, the virtual human interacts with it, trying to find a sitting configuration. As mentioned, the interaction is of two stages. First, the agent is translated and rotated within a bounding box surrounding the object, looking for contact between the object and the sitting areas of the agent. That contact is our anchor predicate and the configurations which satisfy it are called semi-functional configurations. Second, the agent performs an iterative simple sequence of motions, described by the relaxation list in Table 5.1, from each such semi functional configuration. Notice that the iterative process together with the maximal configuration allow reaching every sitting configuration without the need for exhaustive search within the complete configuration space. Search is made only in the 6-DOFs configuration space of translation and rotation. Figure 5.3 illustrates the different stages of the sitting process as mentioned above.

<table>
<thead>
<tr>
<th>Agent</th>
<th>virtual human</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximal configuration</td>
<td>Figure 5.1a</td>
</tr>
<tr>
<td>Anchor predicate</td>
<td>$P_S = {8, 9, 10, 11}$</td>
</tr>
<tr>
<td>Relaxation list</td>
<td>$L = {J_5, J_1, J_2, J_3, J_4, J_6, J_7}$</td>
</tr>
<tr>
<td>Goal predicate</td>
<td>$P_F = {8, 9, 10, 11}$</td>
</tr>
</tbody>
</table>

5.1.2 The Category 'Tables'

In the case of the category 'Tables', the anchor predicate which implies on semi functional configurations is different. Basically, there are many kinds of tables, with different shapes, and different sizes but nevertheless, the variety of their functionality is quit limited. We distinguish between two kinds of tables: working/eating tables and living room tables. The main different is that the first kind provides working platform that can be used for various actions such as eating, writing etc., while on the second kind we usually place other objects. The two kinds have completely different functional meaning. In this work we will identify objects in the category 'Working Tables'. As mentioned, this category
is represented by the functionality 'provides working platform', which is in turn represented by the agent-action pair of a virtual-human agent trying to benefit from the working platform. For that purpose, we need to search for configurations that can expose the fact that a certain object possesses a working platform. It is achieved by defining the anchor predicate for 'Tables' as the one in which contact between the object and the stomach areas of the agent is reached. It won’t have any meaning without the corresponding maximal configuration, shown in Figure 5.2b, which is used during the search for those semi-functional configurations which satisfy the anchor predicate.

Intuitively, semi-functional configuration in this context means that a human can sit close enough to the working platform. The second stage, described in Table 5.2, in which the agent tries to lay its arms on that apparently platform, is

![Figure 5.3: Snapshots from the categorization process for the category 'Chairs'. From left to right: (a) initial state, (b) semi-functional configuration, (c) after relaxation of inner-DOFs that relate to the back (d) after relaxation of inner-DOFs that relate to the arms (e) fully functional configuration after relaxation of inner-DOFs that relate to the legs](image-url)
Table 5.2: Functionality definition for the category 'Tables'

<table>
<thead>
<tr>
<th>Agent</th>
<th>virtual human</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximal config</td>
<td>Figure 5.1b</td>
</tr>
<tr>
<td>Anchor predicate</td>
<td>$P_S = {16, 17}$</td>
</tr>
<tr>
<td>Relaxation list</td>
<td>$L = {J_1, J_2, J_3, J_4}$</td>
</tr>
<tr>
<td>Goal predicate</td>
<td>$P_F = {16, 17, 6, 7}$</td>
</tr>
</tbody>
</table>

Figure 5.4: Snapshots from the categorization process for the category 'Tables'. From left to right: (a) initial state, (b) semi-functional configuration, (c) after relaxation of the first inner-DOF that relate to the arms (d) fully functional configuration after relaxation of the second inner-DOF that relate to the arms.

meant to gain evidence that the agent can benefit from the platform. It is done by verifying a contact between the agents’ arms and the object in goal configurations. The categorization process for the category 'Tables' is demonstrated in Figure 5.4.
5.1.3 The Category 'Beds'

The category 'Beds' is defined by the functionality 'provides lying'. In means of agent-action pair, we represent this category using a virtual-human agent performing an action of lying, by trying to lie down on a candidate object.

Table 5.3 describes the category model. The search for evidence configurations starts with the maximal configuration presented in Figure 5.2c, searching for semi functional configurations using the anchor predicate. The anchor predicate indicates on contact between the object and certain surfaces of the agent that imply on partial lying. From this point, the relaxation process begins, towards the final configuration, which is verified as fully-functional using the goal predicate. Figure 5.5 illustrates the different stages of the above mentioned process.

<table>
<thead>
<tr>
<th>Agent</th>
<th>virtual human</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximal configuration</td>
<td>Figure 5.1c</td>
</tr>
<tr>
<td>Anchor predicate</td>
<td>$P_S = { 8, 9, 10, 11, 12, 13 }$</td>
</tr>
<tr>
<td>Relaxation list</td>
<td>$L = { J_5, J_1, J_2, J_3, J_4 }$</td>
</tr>
<tr>
<td>Goal predicate</td>
<td>$P_F = { 1, 2, 3, 4, 5, 8, 9, 10, 11, 12, 13 }$</td>
</tr>
</tbody>
</table>

5.2 Virtual Palm Agent

Many categories involve grasping during the use of their instances. Although the way an object should be grasped is usually insufficient for full categorization, it implies on functional properties of the category. The human hand is capable of a variety of grasps which were extensively studied both in the medical literature and in the field of robotics. Basically, grasps are divided to power grasp and precision grasp. Power grasp involve large areas of contact between the surfaces of the palm and the grasped object. They are used whenever stability and security are necessary. Precision grasp, on the other hand, are chosen when sensitivity and dexterity predominate. Those grasps involve the tips of the fingers and the thumb.
Figure 5.5: Snapshots from the categorization process for the category ‘Beds’. From left to right: (a) initial state, (b) semi-functional configuration, (c) after relaxation of inner-DOFs that relate to the legs (d) fully functional configuration after relaxation of inner-DOFs that relate to the arms.

Figure 5.7 shows the major grasp categories. In this work, we will refer to the cylindrical (or wrap) grasp. As shown in Figure 5.7, this type of grasp is divided to heavy wrap, medium wrap and light wrap (either with or without adducted thumb). The cylindrical grasp is eligible for variety of tasks, such as holding a glass (heavy wrap) or using a hammer (light wrap). By using a virtual human palm, we can simulate cylindrical grasp of a given object and therefore decide whether its functional properties (i.e., the way it could be grasped) correspond to those of a given category. This is basically a low-level functional test, which together with other functional properties can determine whether a candidate object is an instance of a given category.

Figure 5.6 shows the a virtual-palm model that we have assimilated in the system. We have explored two types of grasping: 'heavy wrap' and 'light wrap'. These two will be used in the categorization process of 'Glasses' and 'Tools'.

33
Figure 5.6: Virtual palm agent. From left to right: (a) description of the virtual palm’s joints (inner-DOFs) (b) description of the virtual palm’s surfaces needed for functionality analysis

<table>
<thead>
<tr>
<th>Table 5.4: Functionality definition for 'Heavy Wrap'</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Agent</strong></td>
</tr>
<tr>
<td><strong>Maximal configuration</strong></td>
</tr>
<tr>
<td><strong>Anchor predicate</strong></td>
</tr>
<tr>
<td><strong>Relaxation list</strong></td>
</tr>
<tr>
<td><strong>Goal predicate</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 5.5: Functionality definition for 'Light Wrap'</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Agent</strong></td>
</tr>
<tr>
<td><strong>Maximal configuration</strong></td>
</tr>
<tr>
<td><strong>Anchor predicate</strong></td>
</tr>
<tr>
<td><strong>Relaxation list</strong></td>
</tr>
<tr>
<td><strong>Goal predicate</strong></td>
</tr>
</tbody>
</table>

There are two main differences between the two grasping types: the radius of the object being grasped and the position of the thumb. In the case of heavy wrap, the thumb is first bent horizontally, and then both the fingers and the thumb iteratively secure the object. In the case of light wrap, the finger iteratively...
close on the object against the palm surface, and only then the thumb bent to
gain closer grip. It is called light wrap with adducted thumb since the thumb is
adducted for more clamping action on the object.

Table 5.5 and Table 5.4 present functionality model 'heavy wrap' and 'light
wrap' respectively. The maximal configuration is common for both of them and
represent the case in which the palm is wide open. In a sense, this configuration
can lead to any grasping configuration using a simple steepest-descent-like pro-
cess. The anchor predicate is similar for the two types as well. It is triggered when
contact between the object and the palm surfaces occurs. The list of surfaces for
which the contact should be verified is represented using $P_S$ where the surfaces
numbering is presented in Figure 5.6(a). The difference between the two models
lies in the relaxation list and goal predicate. We have used the notation $F_n - J_m$

![Grasp Categories Diagram](image_url)

**Figure 5.7:** Major grasp categories according to (M. & P.K., 1986)
to describe the m’th joint of the n’th finger, as can be seen in Figure 5.6(b). The relaxation list models the action taken by the agent when the anchor predicate is triggered during the interacting with a candidate. Said process is illustrated in Figure 5.8, demonstrating the categorization process for the functionality 'light wrap' when used in the categorization of 'Tools'.
Chapter 6

Results

This section presents experimental results for categorization of various man-made objects in which the functionality of the category is clear. In the experiments, we use two different agents, where an agent can be used by several categories. As mentioned, the action performed by the agent defines the unique way in which the agent interact with a candidate object. Therefore, the use of the same agent together with different actions, results in exposing different functionalities.

We present classification results for four different categories. These results include both the classification of 3D CAD-models and the classification of real-data objects scanned from a single view-point (2.5D). Our scanner produces a 3D point cloud, from which a 3D triangle mesh is generated. Since the object is scanned from a single view point, much of it is obscured due to self occlusions. Yet, our categorization scheme manages to overcome these hurdles.

The section is organized as follows. The first three parts deal with the categories 'Chairs', 'Tables' and 'Beds'. Those categories all use a virtual human agent. The fourth part examines the different types of grasping and presents categorization results using a virtual human palm.

6.1 The Category 'Chairs'

The functional definition for the category 'Chairs' is 'something one can sit on'. The category is represented functionality 'sittable' (or 'enable sitting') and by the property 'stable'. As mentioned in Subsection 3.3, the framework of this
Figure 6.1: Chair categorization using simulation of embodied virtual human agent
paper does not include verifying properties, which merely represent the physical condition of the object.

The functionality 'sittable' can be verified using a virtual human-agent, by finding configurations of the agent in which it sits on the candidate object. Verifying that a certain configuration indeed involves sitting concerns verifying that a set of pre-defined surfaces of the agent are in contact with the object. This verification process is performed by the goal predicate function, whose inputs are the set of specific surfaces.

Results for 3D-CAD models

The results for the different 3D-CAD chair-models in the DB are presented in Figure 6.1. The figure presents the best-score configurations that the agent has found. The score is based on the functionality-level of the final configuration. We can determine a threshold from which configurations are considered to be legal. Lowering that threshold will allow the human-agent to improvise configurations.

![Figure 6.2: Some variations that got lower scores but show the variability of the solution](image)

![Figure 6.3: Objects that weren’t classified as 'Chairs'](image)
that are semi-functional. For example, Figure 6.2 presents configurations with lower scores that shows the variability of the solutions. Some of these configurations are definitely legitimate improvisations of sitting positions.

Figure 6.3 demonstrate some of the 3D-CAD models that weren’t categorized as chairs, and emphasizes the strength of the functional paradigm in general and ours in particular. Even though some of the object are very similar to chairs, as long that we haven’t found an evidence configuration of sitting, they won’t be categorized as chairs. The lack of ability to find evidence configuration can be related indirectly to some of the object’s properties, for example: no sitting platform (Figure 6.3a), too small sitting platform (Figure 6.3b) or even unaccessible sitting platform (Figure 6.3c). In any case, using the agent saves us from verifying these situations separately. We simply look for the existence of at least one evidence configuration.

Results for 2.5D real-data models

Figure 6.4a shows part of the real-chairs DB that we use. Figure 6.4b shows the same real objects after they were scanned and inserted to the virtual simulation environment. As mentioned, the scanning was performed from a single view-point, generating a 2.5D object with only partial information due to self occlusions. The use of simulations helps facing the difficulties of self occlusions and noise (i.e, the noise in the mesh, holes within the surfaces etc.). As before, the agent is searching for fully-functional configurations. In case that there exists a noticeable sitting configuration from the scanning view-point, that configuration will be found during the simulation. As can be seen from Figure 6.4c, view points from which there is sufficient information on object’s structure allow correct classification. A more detailed analysis will be presented in Section 6.5
Figure 6.4: Part of the real-chairs DB, the corresponding real-data models generated by scanning the real chairs from a single view-point and the configurations that got the highest grade.
6.2 The Category ’Tables’

As mentioned, we distinguish between ’Working Tables’ and ’Living Room Tables’. The functional definition for a working table is ’something that a human sit next to, and benefit its working platform’. The use of virtual human agent together with the specific action defined in the model, encapsulate this definition into a simple test of finding an evidence configuration. Using the anchor predicate, the searching algorithm first finds configurations in which the stomach surfaces of the agent are in contact with the candidate object, verifying that it agent can be positioned in a sitting configuration which is indeed close enough to the object. Only then an action, whose purpose is to verify the existence of a working platform, is triggered. The action involves trying to lay down agent’s arms on the candidate object and by doing so, verifying several properties of the candidate object, and the working platform it should posses. It is performed by the relaxation algorithm together with the goal predicate that checks for contact between the arms of the agent and the object, hence indirectly estimating the existence of the working platform, its area, its positions etc. All of these properties are function of the number of contact areas between the agent and the candidate object.

Results for 3D-CAD models

To test our method, we have used a large DB of 3D-CAD tables. A part of it is presented in Figure 6.5. The result are presented in Figure 6.6. These results demonstrate the best-score evidence configuration found by the human agent. In the case of tables, the agent may find many evidence configurations for each category instance. In Chapter 7 we talk about how the analysis of the number of non-overlapping solution can hint on the sub category of the object.

Figure 6.7 present some examples that weren’t categorized as ’Working Tables’. Figures 6.6 (a), 6.6 (b), 6.6 (c) and 6.6 (f) for example are ’Living Room Tables’ that supply different functionality than ’Working Tables’ and should be categorized using different agent-action pair. Figure 6.6 (d) present a knowledge limitation: for us it is obvious that the table is covered by a tablecloth, but the agent doesn’t know the properties of the tablecloth and the fact the it is flexible.
Figure 6.5: Part of 3D CAD DB for the category 'Tables'
Figure 6.6: Results for the category ‘Tables’
In fact, the tablecloth prevents finding evidence configuration since the agent assumes it is rigid. Figure 6.6 (e) present another example for which the agent didn’t manage to find evidence configuration. The reason is that the middle table leg prevent the agent from being close to the table and benefit from its platform.

**Results for 2.5D real-data models**

Figure 6.8a shows part of the real-tables DB. Figure 6.8b shows the same real objects after they were scanned from a single view-point and inserted to the virtual environment. Since the viewpoint exposes the working platform and the area next to it, the agent have managed to find evidence configurations, shown in Figure 6.8c, which prove that the objects presented are indeed instances of the category ‘Working Tables’.
6.3 The Category 'Beds'

Figure 6.9 presents results for the categorization of beds 3D-CAD models. The results show evidence configuration in which the agent lies down on the bed, and expose its functionality. As in the case of tables, there were several solutions, demonstrating fully-functional configurations, in which the agent is lying down on the bed in different orientations. For a human, the direction in which one should lie down on a bad is quite obvious, and depends on recognizing the principal...
axis of the bad. This kind of test can easily be integrated into our framework by providing higher grade for solutions which are aligned with the principal axis of the bad.

6.4 The Category 'Graspable'

This section demonstrates the categorization results for the functionality 'Graspable'. This functionality is part of the functional definition of many categories, among them 'Tools' and 'Glasses'. As mentioned in Chapter 5, the functionality 'Graspable' can be represented by the agent-action pair of human-palm agent performing a cylindrical grasp. We have also explained that the cylindrical grasp
Figure 6.10: Part of the DB of the categories 'Tools' and 'Glasses'

includes several sub-categories, among them are the 'Heavy Wrap' and 'Light Wrap' grasps. The difference between the two was also examined.

Figure 6.10 present some of the 3D-CAD objects that belong to the categories 'Tools' and 'Glasses'. We have tested these objects under the action 'Light Wrap Grasp'. The results are presented in Figure 6.11. We can see that all instances of the category 'Tools' posses the functionality 'provides light wrap grasping' (Figures 6.11 (a), 6.11 (b), 6.11 (c), 6.11 (d) and 6.11 (e)). Some instances of the category glass (namely the wine glasses) also enable light wrap grasping. As was explained, the light wrap characterizes grasping small radius object. Since a wine glass can be grasped in two different ways, it provides both light and heavy wrap grasping.
6.5 Sensitivity Analysis

The following section investigates the sensitivity of our approach in the categorization of 2.5D objects (3D objects viewed from a single viewpoint). The analysis takes into account (a) Categorization sensitivity to viewing angle and; (b) Categorization sensitivity to the number of contact areas (of the agent). We have used a 3D model of a chair and chosen two viewing trajectories on the chair’s viewing sphere (see Figure 6.12). For each trajectory, we have generated 360 2.5D snapshots and categorized each one of them using our system. The results are presented in Figure 6.13a and 6.13b.
Figure 6.12: Two trajectories on top of a 3D chair model which were used to produce 2.5D models from different views.

Figure 6.13: Sensitivity analysis for 2.5D view-points. A 3D chair model was used to produce 2.5D models from different views on two trajectories. Each trajectory was tested with two robot-actors with different number of support areas: (a) The two viewing trajectories (b) Grading vs. viewing angle for trajectory A (c) Grading vs. viewing angle for trajectory B.

6.5.1 Functional Event

In general, one can see that Figure 6.13a and 6.13b qualitatively present almost identical graphs for the two different trajectories, suggesting some robustness to trajectory elevation. Along the trajectories, we path through degenerate views and visual event in the form-plane, as defined in (Kender & Freudenstein, 1987). These visual events occur since the projected shape of the object (the form-plane)
undergoes dramatic changes along the trajectories. Since form and function are mapped in some way, as mentioned in (Rivlin et al., 1995), it would be interesting to see what would happen in the function-plane along the trajectories, and that is the purpose of the graphs in Figure 6.13a and 6.13b. We can see that the graphs are intermittent constant. The leaps between the constant segments imply for events of some kind, and since our categorization takes place in the function-plane, those leaps are actually *functional events* in a sense of fulfilment of functional requirements.

Since the mapping between form and function is not injective, a visual event in the form-plane (such as the appearance of the face of a cube) does not necessarily generates functional event. Therefore, from a categorization point of view, functional events are much more informative than visual event since they expose the existence of certain object functionalities, and therefore helps in determining its categorization. For example, we can see from Figure 6.13 that at viewing angle of approximately 30 degrees, there is enough visual information for the virtual human agent to partially sit on the 2.5D object and to expose its ‘sittable’ functionality. At this angle, the visual information crosses a certain threshold causing a functional event.

Categorization’s sensitivity to the number of contact areas was performed by using virtual human agent with additional contact areas. As mentioned, the support areas are at the base of finding semi functional configuration as well as grading the final configuration. They are used during the collision detection queries to verify contact between the agent and the object at strategic areas of the agent. The more support areas, the better the accuracy in grading functional configurations. From the graphs, we can see that additional support areas do increase resolution but the major functional events remain approximately at the same angles.
Chapter 7

Conclusions and Future Work

7.1 Conclusion

We have presented a new, cognitively-motivated approach, for functional categorization of 3D object using the embodiment of virtual agent. Our approach models a category using a set of functionalities and properties. The new insight relates to the way in which functionalities are represented. In harmony with motor-cognition psychology, we represent functionality by a pair of agent and action. By performing a certain action upon a category instance, the corresponding functionality can be exposed or validated.

The implementation of the concept involves using a virtual environment in which a simulation of the interaction process between the agents (one for each functionality) and a candidate object takes place. We have presented a generic search algorithm that search the configuration space looking for evidence configurations that prove the fulfilment of the functionality. The algorithm involve two steps - searching for semi functional configurations and relaxation.

The first step is meant to spear the need of exhaustive search within the full configuration space of both global-DOFs and inner-DOFs. Instead, we look for the 6D configuration space of global-DOFs, looking for semi functional configurations from which a unidirectional iterative process can begin. The semi functional configurations are identified using the maximal configuration and anchor predicate. This step is actually functional pruning of the configuration space motivated by cognitive heuristic.
The second step is the relaxation phase which simulate the action performed by the agent upon the candidate object. The action is represented using a sequential movements of the inner-DOFs of the agent. When this process encounters a dead end, the goal predicate is called to check for fully fulfilment of the functionality. both the anchor and goal predicate are implemented by means of two quick collision detection queries supported by our deformed BVH-tree.

We have tested the algorithm on five different categories: 'Chairs', 'Tables', 'Beds', 'Tools', and 'Glasses'. For this purpose we have presented two virtual agents - virtual-human agent and virtual-palm agent. By simulating different action performed by these agents we have managed to represent the five categories mentioned. We have tested our method on both 3D-CAD models and real-data objects that were scanned from a single view-point. The categorization algorithm has shown very good results, and has managed categorize objects correctly despite noise and self occlusions.

7.2 Major Contributions

Here we summarize our main results:

1. **Number 1:** We have integrated insights from the world of motor-cognition psychology and simulation theory into the world of computer vision, bringing new approach for the problem of object categorization.

2. **Number 2:** We have shown how functionality can be represented by means of agent and function. Since the agent is the receiver of the functionality, when it perform the right action upon a category instance, the functionality is exposed and therefore can be easily validated.

3. **Number 3:** We have demonstrated how categorization can be translated into a simulation process in which the agent interact with the candidate object within a virtual environment

4. **Number 4:** We have presented a search algorithm for finding evidence configurations, which is independent on the number of inner-DOFs of the agent, and is based on functional pruning of the configuration space.
5. **Number 5:** We have defined the concept of functional events, and investigated the connection between these events that take place in the function plane and visual events in the form plane.

### 7.3 Future Work

There are two directions in which this concept can be extended. The first one is trying to generalize the model to other categories, other than man made objects. And the second one is to perform sub categorization of the objects.

Regarding the sub categorization, we have mentioned the idea of analyzing the evidence configurations found by the agents (in most cases there is more than one solution). Let’s analyze, for example, the solutions of the tables presented in Figure 7.1. Unlike chairs, for which the number of solutions is quite limited, the agent many evidence configurations for each table. By analyzing the number of non-overlapping solutions, we can get a hint on the sub-category to which the object belongs to. For example, if we find only one solution, it might be a student table, or an office table that enable sitting for only one person as presented in Figure 7.2. Other cases, such as Figure 7.3 and Figure 7.4 represent chairs of different functionality. The first one is a table for two while the other one, for which there are four or more solutions, represent family table, or meeting room table.
Figure 7.2: Sub-categorization for 'Table for one'

Figure 7.3: Sub-categorization for 'Table for pair'
Figure 7.4: Sub-categorization for 'Table for family'
References


씨ורד עצמים פונקציונאלים מבוסס
синמלציה על-ידי סוכם

ער בר-אבל
סיווג עצמים פונקציונאלים מבוסס

سميוליציה של-ידי סוכן

חריר על מתקר

לש מוסר חקל של הדרישות לקבלי ההוזה

מנוסר להודעה במנועי המחשב

ערד ב.-אביב

ה泯ה לסכנת טכנולוגיה – מכון טכנולוגיה לישראל

חריש מ. ’ט

מצרפ 2009
דביקה צ炆ים לשינוי בצוותה הופשה בוחר, ולא פשע מכ

— אלברט איינשטיין
# תקציר

<table>
<thead>
<tr>
<th>מספר</th>
<th>ת목</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>תקציר</td>
</tr>
<tr>
<td>1.1</td>
<td>הפרטים המתוクロ</td>
</tr>
<tr>
<td>1.2</td>
<td>מבנה התוכן</td>
</tr>
<tr>
<td>2</td>
<td>סקירת ספרות</td>
</tr>
<tr>
<td>2.1</td>
<td>יוזמות תעס סמסフェפי</td>
</tr>
<tr>
<td>2.2</td>
<td>שיטות פואטרים</td>
</tr>
<tr>
<td>2.3</td>
<td>שיטות פואטרים</td>
</tr>
<tr>
<td>2.4</td>
<td>שיטות פואטרים</td>
</tr>
<tr>
<td>3</td>
<td>מדד ק用量ך</td>
</tr>
<tr>
<td>3.1</td>
<td>מדד ק用量ך</td>
</tr>
<tr>
<td>3.2</td>
<td>מבנה מדד ק用量ך</td>
</tr>
<tr>
<td>3.3</td>
<td>הנבדלות</td>
</tr>
<tr>
<td>4</td>
<td>מסכתת רבעון עצם</td>
</tr>
<tr>
<td>4.1</td>
<td>אלגוריה חפות שליטה</td>
</tr>
<tr>
<td>4.2</td>
<td>שיעורי בעלות והתנהלות</td>
</tr>
<tr>
<td>5</td>
<td>מרךעם רפרסנטליות</td>
</tr>
<tr>
<td>5.1</td>
<td>סוכן וירטואלי דמוי אדם</td>
</tr>
<tr>
<td>5.1.1</td>
<td>קטעוריה 'מיסא'</td>
</tr>
<tr>
<td>5.1.2</td>
<td>קטעוריה 'שולח'</td>
</tr>
<tr>
<td>5.1.3</td>
<td>קטעוריה 'מייט'</td>
</tr>
</tbody>
</table>
רשימת אורות

חלק המחברת על מדליים וארופים תחת מימד מיший קטגוריה

1.1 כיסא

2.1 מוערכת

2.2 דגמה ת Helvetica

2.3 דגמה במ党的领导

2.4 מספר דגמאות בשכיהZA, עניין התחייו של 유지בוסט

2.5 קול מתין זוג קטגוריה

2.6 לדמה של מערבת

5.1 תאור המבנה של סוכם ויתראולים בדם צדד, האזור מתאר את

5.2 המפרקים והativo של מעשה במגע ממעטנא לים: קנט מבלינס, מבט מראור

5.3 עוגר הקטגוריה שלוחן, עוגר הקטגוריה 'מיטה'

6.1 עכל בטנייה של והתלך משגוי עוגר הקטגוריה 'כיסא'. מושמאל

6.2 ל��ן: מצב התהליך, מציאת עם פנקסואליות הקליא. אחרי ל-

6.3 הקצינו של דורון חפש מيمن הקצירה של, אחראי לクラス

6.4 הקצירה של דורון חפש מيمن הקצירה של, אחראי לクラス

7.1 שולחני המבנה של התלך הסיווג עוגר הקטגוריה שלוחן' מושמאל

7.2 ל��ן: מצב התהליך, מציאת עם פנקסואליות הקליא. אחראי ל-

7.3 הקצינו של דורון חפש מيمن הקצירה של, אחראי לクラス

7.4 פינות עם פנקסואליות שלחה אחראי לクラス שולחני הקצירה של
שבלים הבכירים של החלק ושל מקום הקוטגוריית ומלאת כנפיו

ל大大提高 את התפקידים של הקוטגוריית וה commercים reklami, התפקידים וה מוצרים של הקוטגוריית

המקורות של הממונה

תאורה במנה של סוכן וירטואלי בדמויות כ-ד. משחזריLim: תיאור

מגדה התוכן, תיאור onFailure של机油.

канטר危險 תוניית מרכז

שלבים הבכירים של החלק והסימונים עומר והפקודות לאחיה זהה.

משמשלLim: מסב החלק, תיאור של מקום הקוטגוריית והלקחים,

הצוגה סימונים של מקום הקוטגוריית ומלאת אחריה לקבוצת של דרジョン

אומנות מפורשים

ซอยי יכולות два סיכונים של סוכן וירטואלי בדמויות אדום.

מספר וירטואלי עם וכפל יוצר המיומנות של מומחיית הפרט

האפרסיין

עניבים של הטוונה קסופה

תקנון מנגורים המדליות והאמורית של סיכונים שיפורים בแนะנעות כ-יר.

 PureComponent חצי

תקנון מנגורים של מדריפס התכלת ממידים של לקטורייה

שלחט ο'קשת

ה שכלת סוחר הקוטגוריית מצלית

וה וכלת עזר הקוטגוריית מצלית.

תקנון מנגורים של המדליל הגרופי של התכלת ממידים של בקתגוריית

אוליס 'ο'קשת

فعاليات על הגשם של מספין תכלת ממידי בהב פ.mousePositionי בוליש

מדליות חלקיות וסוכנים של התוכננתים של יחידי ייחד

בייחוד ורשות עזר פסימס של חלוקת ממקים יחידי

6.1
6.2
6.3
6.4
6.5
6.6
6.7
6.8
6.9
6.10
6.11
6.12
6.13

רשימת צלצלות

29 .......................... 5.1
31 .......................... 5.2
32 .......................... 5.3
34 .......................... 5.4
34 .......................... 5.5
תקציר

סיווג acompaña لماולקטוק חמש כאותות הביעויות החשיבות בחוזה הרצאות הממד.
והשבח. הקחshi טעם בעבדה שיש שונית פנימי בין ענפים השיניים שלאות הממלכה.
ולכן קשה להגדיר Màoao מקוללת בציור פורמלית. היווה את כלגרון הביעה נקראה "סיווג
ענפים מביסס מדל"יל פניה גיתח מאולק העם או דלי א conexión פורמלית ספגתי.
ככ שולח
ענפים במאולקה כיון חותם מתחא על דלי אכנה פורמטים ספגטי. באישת גישת כיון.
ייתוי התוכן שמח. מעריצים. ה الكرט הענף ענפים שניה חכימה לאולקהת הממלכה הזה
ככל זה מארץ (וישבב משל על הממלכה "מסאות" על השוחרי הפרטים בין ענפים במע.
המקהל.

ההגנה הפיקודים מאולקת מעמדות על הביעות ביצר את. במדכeten ל hồים
לגרד הצגה. גישה גיוס חידי ענפים על פי הפיקודים של מאולקת שיח.
ככ בגרדוז. קסא ודאר "מראה שיווה לשבח עליל". ארמן המדה. המ樂ה.
קציונאות להזורות מהמלך ענפים על די סל של הפיקודים תפוקדות בא сталиים
שיהוא ממלאת. לארה מנ. הבינרי ענפים מסתיים. יש לזרוק שירות ערא מסוגל התלמה
ככ א zd מאיתפירים הפיקודים תפוקדות של.

את הגישה המקוריאת בחוזה ממטרה כל הפיקוד תפוקדות ביסיに乗ור.
סט של המלך פפרט, של התזה ממטרה על די הפיקודים אוטומטיים ביסי.
ככ למשול סס מזרן על די הפיקוד תפוקדות ביסי - "מספק שיווה". הפיקודוז.
וכל מתח משלי סס מזרן על די הפיקוד תפוקדות "מספק השיווה של". במ. כל את
מהור על די התזה "מספק השיווה של". "מספק השיווה של". ומ. כל את
 digitalWrite(13,1); // turn on light
 digitalWrite(13,0); // turn off light

 // turn on light
 digitalWrite(13,1);
 digitalWrite(12,1);
 digitalWrite(11,1);
 digitalWrite(10,1);
 digitalWrite(9,1);
 digitalWrite(8,1);
 digitalWrite(7,1);
 digitalWrite(6,1);
 digitalWrite(5,1);
 digitalWrite(4,1);
 digitalWrite(3,1);
 digitalWrite(2,1);
 digitalWrite(1,1);
 digitalWrite(0,1);

 // turn off light
 digitalWrite(13,0);
 digitalWrite(12,0);
 digitalWrite(11,0);
 digitalWrite(10,0);
 digitalWrite(9,0);
 digitalWrite(8,0);
 digitalWrite(7,0);
 digitalWrite(6,0);
 digitalWrite(5,0);
 digitalWrite(4,0);
 digitalWrite(3,0);
 digitalWrite(2,0);
 digitalWrite(1,0);
 digitalWrite(0,0);
الفוקפסיאולים של העצם. בטלה הרקמה של המיקוד לתוך פונקפוסיאולים על ידי א-טלה קבוצת פונקפוסיאולים שאמוריזים לתוך קלבת (למשל ת_APB). היקה (ר'), ההבלד והא-טלה שבין הקבוצת פונקפוסיאולים המופצלת לאחר תיפוי את הצAPS פונקפוסיאולים Украינה לשתי קבוצות פונקפוסיאולים. כל קבוצת פונקפוסיאולים יוצרת יצרן במובח מובח למובח למובח במובח למובח במובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח למובח لم

האנטרוקפסיאולים בינוון בין המים במובח

בנוסף, מוחל פונקפוסיאולים לכל מבנים מחנה במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח

מעשים אלו קובעים את הדריך בסיסון יוצאת בצבע המים במקום יוצאת בצבע המים בישראל בחינה ובין מנהלי הסומטולוגיה של המים.

묻אים שלוש ימי- переход דרך צדyses בין המים slaughtered במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח במובח

ולאנטרוזמס הפונקפוסיאולים בינוון בין המים. מרחב הזווית המופלג דרור החופש בצורת המרובע. המרובע המזרחי והחופש הגלובלית (שש דרור החופש של ההתקפה של הקבוצה)

דרור החופש פגמים של הסוסון (וקמוסות בחימר מפיבאוס). עד שאליה מוחש ומגמה של מרחב התטרודון. פיגור הגלובלים הופך ד-שלב

.detects על ת有任何 קוגניטיביים, מבני הפרפרים של הדור בין המים.

ה焗 הטרודון והגלובלית בבל. בלע המים, האלגרטים מופלצ上の פונקפוסיאולים. הקבוצת הפונקפוסיאולים בלילה ממוקזת דרור האניית טיפוס על ת有任何 המרובע הגלובלית. אטר

הפונקפוסיאולים של הקבוצת ממורב בת阿拉伯 ומע멎ית מגן בין אריה עיוני מבויתים של הסוסון בינוון. בלע זה בצע מבצע של פונקפוסיאולים במובח החופש. מורג שלושה מעג כה, מחחי שלוב השתייך שחקא שלב הרוקסזיאים. בז reimbuated
סדידות פעולות ח"-כוניות של שנויי כ-אחת מדגמי הѥבעים הפרמייסים ועל הסדר
סדר פעולות ח"-כוניות של שנויי כ-אחת מדגמי הѥבעים הפרמייסים ועל הסדר
הפרמייסים. שולחן העצםált קובע מוגק ב-אינוויזיון הפרמייסים השל הסדר-picture.
כאמור את נהג מעיד על קושי מומל של הפרמייסים-alt. כן למשולס בוא הסדר מתא.
תורור השופט בלשוןدم המבואר והב עוצרו האיטי של של. אין יצורתי ימיםSTER
שאיפה מספק את הפרמייסים ה阿富汗י מסכין מכאע בין לש שבעה ארבע פרמייסים
LabelText
אות השימטשון. גובא את הת化合 השבינה, שופע הגטרת מרשל אש-וירטואלי
היפות התופעה שיבחה שול עונה על כל הידועות ההלל ברורה אתונהינית.

ופסקאות המסרה הנותנת צוילłąת הפרורה. על-פיに入 הופסקאות הפרמייסים
המודרנויות ב-כאמור עדיה ונהג מעיד על הפרמייסים הפרמייסים ה阿富汗י. הפרמייסים
מעיד ע-אילנות. על-ディ בחרת הת imagePathו ו-האיון הנו בוני ח *&ךחרת ערור אפיש
בינセンター הולרגס מתי חירר הוטה בחירו פכיר פלועל על הווה כ-די לפוצות את
הפרמייסים הפרמייסים ה阿富汗י של, לשלו מתי חירר הוטה בחירו לאות בכי מיסים.

ולשבור על אמע מסיסים כ-די כבכל חומרי מיקרית.

על מונח לבוות את השישה, בגבי מדרידים של-ש יס בוכס boosted מים. סרסי
א那份 הסור דמי כ-די. בישולב ע-פעולות שרות. סרסי אל מסוליס של-תונקוד
הפרמייסים הפרמייסים ה阿富汗י. אנא השתרצתי בוססכים אל כ-די להזח תמחלק
עמעום שנותו: 'כסalion', 'שלוםת', 'מיוג', 'כלי-קרב'. 'מטווז' -ר-פרווס. המופרעת מביקה
ון עבר סרס Çok מדרידים תד-מדיסים, והז עומר פסמיים שמגזרי שומניה
(typewriter) (מור שניה עטור בפריים וחפיים ומיד בזרזזר רבם ולא כביקה).
ה perso小男孩 מאיירו על-됄ות. שואור אנ פ פריים למצלר גבר. בנסס סלש-חקי
יס בכסירה כי, מכיוון שסוכן הסירה וה-גחיד. ישס השחרות שמסימי המ𝐚턱וד
על גיאון העניב הפרמייסים הפרמייסים ה阿富汗י. למדורו, והשתתף הצלחת כעילה מזרז
הפרמייסים הפרמייסים על עומר המדרידים תד-מדיסים והז עומר צסיום ה탈ד:ן-

ם.
בנוסף, בחינת א יותר מערכות מערכות לוויתן הרחבנות. מיצוג תוצאות פיתחניה רמת התפקוד:

דיוט הפונקציות לארוך מוסיפים מעגל סוסים סבינה הצעה, שמעל בישוע קפיציות

ב揭露ת הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפריט

תחוש הנקרא "ג_basename הפונקציות", הפונקציה למישור הالفaktivיטות הביצוע אוזוריון הלמידה של מישור

כפיונטים מהרתם כפעילות является חเต็י מודים והנסוגת שוחזר רמת התפקודיות פונק.

כפיונטים הדישה, לפנים נשגיעה绔 רמות, ולפי ניוג לווה את מש.

ות השיבוץ, כלק קושי קידום שלט כיסא. בקדחה פסימית, משטת השיבון חטף

מספיקים פ שיוואה אופסף שישבה. הביאו גם עיצון קיפוח לא רציה白马 התפקוד:

דיוט הפונקציות המ潦ות מוסיפים מספיקים (מאותה תקודה מבנה). הביא שיעור בועה הסיום

ולמلاقات איראונות שauga מהגים הרבע יוצרreeze ויויאל.

ليسים, הוגיה расположен שישה צייגות המבוססות על הקרונות פיסICollection com.

 وهنا הרעיון של הפרוגנום הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפונקציות הפריט

מסווגים שיאה והשכינה למחלקות שותף, התראות מחלדות כי מודוב ברשות עטולה

ממדנרט על השתייהدورו בק אפוסים שיעם של ראזיה ועל הרב מצוקה ומק על יער השחרור

עטימ <!>