May 1993

An Active Approach to Characterization and Recognition of Functionality and Functional Properties

Luca Bogoni
University of Pennsylvania

Ruzena Bajcsy
University of Pennsylvania

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-93-50.

An Active Approach to Characterization and Recognition of Functionality and Functional Properties

Abstract

Functionality in an object can be defined as its applicability toward the accomplishment of a task. We emphasize and develop an interactive and performatory approach to functionality recovery from sensor data in the context of robotic manipulatory tasks. By analyzing interaction of tool and target object and manipulation tasks as goal-oriented recognition processes, we propose to identify and characterize functionalities of objects. This interaction is not only a means of verification of the hypothesized presence of functionality in objects but also a way to actively and purposively recognize the object. The representation of functionality allows us to extend the recovery process to a hierarchy of functionalities allowing complex ones to be composed from simpler ones.

A formal model, based on Discrete Event Dynamic System Theory (DEDS), is introduced to define an interactive task for recovering and describing functionality. To observe and control the recovery process we introduce the notion of piecewise observability of a task by different sensors. This allows the description of a dynamic system in which not all events nor the time of their occurrence may be predicted in advance. An experimental system, with both vision and force sensors, for carrying out the interactive functional recognition is described.

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-93-50.

An Active Approach to Characterization and Recognition of Functionality and Functional Properties

MS-CIS-93-50
GRASP LAB 347

Luca Bogoni
Ruzena Bajcsy

University of Pennsylvania
School of Engineering and Applied Science
Computer and Information Science Department
Philadelphia, PA 19104-6389

May 1993

An Active Approach to Characterization and Recognition of Functionality and Functional Properties

Luca Bogoni

and

Ruzena Bajcsy

GRASP Laboratory
Department of Computer and Information Science
University of Pennsylvania, Philadephia, PA 19104

July, 14 1993

An Active Approach to Characterization and Recognition of Functionality and Functional Properties

Luca Bogoni

and

Ruzena Bajcsy

Abstract

Functionality in an object can be defined as its applicability toward the accomplishment of a task. We emphasize and develop an **interactive** and **performatory** approach to functionality recovery from sensor data in the context of robotic manipulatory tasks. By analyzing interaction of tool and target object and manipulation tasks as goal–oriented recognition processes we propose to identify and characterize functionalities of objects. This interaction is not only a means of verification of the hypothesized presence of functionality in objects but also a way to actively and purposively recognize the object. The representation of functionality allows us to extend the recovery process to a hierarchy of functionalities allowing complex ones to be composed from simpler ones.

A formal model, based on Discrete Event Dynamic System Theory (DEDS), is introduced to define an interactive task for recovering and describing functionality. To observe and control the recovery process we introduce the notion of **piecewise observability** of a task by different sensors. This allows the description of a dynamic system in which not all events nor the time of their occurrence may be predicted in advance. An experimental system, with both vision and force sensors, for carrying out the interactive functional recognition is described.

Contents

1 Introduction

Object recognition systems involving multisensory modalities are focusing more and more on being adaptive and capable of learning. Hence it is essential that a system supporting this flexibility be able to investigate its environment and determine not only the physical properties of an object but also its applicability in a task.

In this paper we investigate the recovery of functional properties of objects from sensory perception. We focus our attention to functionalities which can be investigated and modeled by manipulatory tasks. The recovery process presented addresses the observability of interactions as well as relations between objects. Hence the experimental set up developed is suitable for observing a manipulatory task being carried out, verifying the hypothesized properties of object, or for the recovering the material properties of an object.

Functionality is not a characteristic unique to a single object and a particular object may have more than a specific functionality. For example, a fork could be used for cutting. Many artifacts do, in fact, possess more than one functionality and do so in different degrees of performance. Furthermore, while a functionality can be described abstractly, the functional attribution of an object is context and application dependent. Thus, a knife can be identified as an tool suitable for cutting another object but it is the applicability of a particular object for cutting which allows us to identify it as a knife.

We develop an interactive, dynamic system in which functionalities in an object are investigated in several contexts. Such investigation requires that the interaction be controllable and observable by different sensor modalities and hence that the data obtained from different sensory sources be fused. Furthermore, it is necessary that the result of the interaction be qualitatively and quantitatively evaluated against the expectations. Thus, an object's functionality is hypothesized and then verified by carrying out a manipulatory task intended to verify the presence of the hypothesized functionality.

This paper is organized as follows: section 2 presents the related work. Section 3 introduces a characterization of functionality. Section 4 addresses the importance of the interactive approach for recovering functionality. Section 5 introduces a formalism based on DEDS to describe a task model that can both express and interactively recover functionality. In section 6, we discuss how the individual functional tasks may extend to an algebra of functional tasks. In section 7, we introduce the experimental system being developed for determining the functionality in objects. Finally, in section 8, we conclude by pointing out further developments of the recovery process for functionality.

2 Related Work

Freeman and Newell [Freeman and Newell, 1971] were amongst the first who addressed functionality in objects as means of "devising artifacts for accomplishing goals". In ACRONYM [Brooks, 1980] one of the first attempts to bring functionality to object recognition is presented. In [Lowry, 1982], the author points out that functionality should be represented as a hierarchy of kinematic primitives, functional primitives and causal networks. In [Winston *et al.*, 1984], the authors use natural language descriptions to provide object physical identification and show how physical models can be learned using functional definitions.

In [Brady *et al.*, 1985] Brady et al. present a system, "Mechanic Mate", intended to assist a handyman in a generic construction and assembly operation. The paper addresses the interplay between planning, reasoning, and the functional significance of higher order structures. In [Connell and Brady, 1987] Connell and Brady describe a system, based on a modified version of Winston's Analogy program, which uses semantic nets to investigate the relation between form and function.

More recent investigations of functionality were carried out by Stark and Bowyer, [Stark and Bowyer, 1991]. They focused on the classification of CAD models of chairs. This work addresses the shape of the object and of its components as means of detecting functionality. Davis, [Davis, 1991], proposes two theories of "cutting". In the first one, he addresses the intrusion aspect of the knife; in the second one, he focuses on the chunks resulting from the cutting procedure.

In Psychology, Jordan, [Jordan, 1991], addresses the importance of physical properties in understanding object functionality. Smith and Medin, [Smith and Medin, 1981], point out how functional features should actually be considered part of the core features appearing in concepts description. In Robotics the work of [Cutkosky, 1989; Iberall *et al.*, 1988] addresses the importance of understanding functionality when manipulating and interacting with an object.

Finally, we find some additional sources on the importance of functionality in the studies carried out by anthropologist J.Goodall, [Goodall, 1986]. She investigated the functional usage of tools by Chimpanzees. The importance of functionality and the extraction of the functional properties of objects is also found in the study of the function of stone tools carried out by anthropologist R.Grace, [Grace, 1989].

What becomes apparent in the analysis of the related work is that the importance of functionality has been recognized for quite a while. However, the work in this area is rather sparse and only recently has it begun to receive widespread attention. The reasons for this apparent lack of interest are due to the complexities associated with the characterization, the representation, and the recovery of functionality. We will now investigate each one of these aspects.

3 Functionality Characterization

The **functionality** in an object identifies its applicability toward the accomplishment of a task. The function of support in a table, for instance, points out its applicability for bearing some object on it. The function of hammering, on the other hand, identifies in a tool its suitability to apply an impact force on some target object(s).

Properties that objects possess can be classified as:

- **Geometrical** properties identify quantifiable parameters defining shape in terms of length, width, height, volume, etc.
- **Material** properties are also quantifiable measures. Their attributes are defined in terms of units of weight, coefficient of friction on the surface, thermal properties, etc.
- **Kinematic** properties identify the mobility of parts in an object, such as in a pair of scissors.
- **Dynamic** properties describe how the object responds to forces applied to it, such as the behavior of a compressed spring.
- **Functional** properties in an object identify the set of physical, material and geometrical, kinematic and dynamic properties which characterize the functionality of an object.

Considering the properties listed, it becomes clear that different sensor modalities need to be employed to recover them. Global and local geometrical properties, such as volume may be recovered from visual observations using stereo, shape from X for monocular vision, laser-ranging sensors [Shirai, 1987]. Material properties may be recoverable visually by looking at reflectance and textured qualities of the surface. By using exploratory procedures (EP), [Klatzky *et al.*, 1987; Lederman and Klatzky, 1987], however, compliance and surface texture may be "felt" by using contact type sensors. Temperature probes may also be employed for actively determining constituents materials [Campos, 1992]. Kinematic [Campos, 1992] and dynamic properties [Sinha, 1992], however, require more complex EPs.

Functionality can be characterized as intended, imposed, intrinsic, and inherited.

- **Intended** functionality identifies functional properties defined in an artifact at the time of its design.
- **Imposed** functionality defines the ability of using an object for a function for which it is not necessarily intended.
- **Intrinsic** functionality denotes functional properties which either characterize an intended functionality, in the case of an artifact, or define a functionality in virtue of physical properties of the object.
- **Inherited** functionality denotes an object which is either a specialization of some object or a new object in which functional properties are combined from different objects to fulfill one or more functionalities.

To clarify the distinction between intended and imposed functionality we note that a fork is constructed with the intended functionality of piercing and carrying, yet one may impose on it the functional property of cutting. Artifacts in general possess both intended and imposed functionalities. Natural objects, such as rocks, on the other hand, have imposed and intrinsic functionality.

Intended and intrinsic functionalities are characterized by necessary functional properties while imposed functionalities require the object to possess properties which are sufficient for it to be applicable in the context. The "rigidity" of a table's surface is a necessary material property of the object to afford the function of support. On the other hand, "thinness" in a penny is just a sufficient property for applying it as a screwdriver. In the case that the functionality imposed on an object coincides with the intended object's functionality, then the functional properties are both necessary and sufficient. This last case identifies the application of the proper tool for a specific task.

The characterization of functionality as inherited is useful for classifying an object in terms of its functionality with respect to others. This type of specification relates, for instance, the functionality of containment fulfilled by a tea–cup to that of a glass. The process of specialization of the functionality of an object or that of combining functionalities from different objects constitute a designation of one or more functionalities into an artifact. However, the designing of some functional properties in an artifact is what we had identified as intended functionality. Hence while this distinction is useful for classification, we will not dwell with it further at this point.

4 Functionality Representation

Functionality, unlike any physical properties identified in the previous section, adds an interactive component to the representation. Namely, it defines *how* the object is to be used. It addresses the interactivity of a specific object with some environment and introduces, implicitly or explicitly, the notion of expectation of the result of the interaction. Consider, for instance, the operation of cutting butter or a sponge. In both instances the expectation of the end–result would indicate a partitioning of the object into two parts. However, the intermediary states are important and contribute to qualify the performance and the different behaviors which occur in the two cases. Thus, while the results of the interaction are important, the intermediate states between the beginning and the completion of the interaction are also relevant. In particular, if one is to carry out the task describing a function, the intermediate states provide information about the properties of the objects involved in the interaction, the environment the interaction takes place in, and the quality of interaction. Thus, an abstract description of a functionality could be represented without details addressing how the transitions, which culminate in the different events, actually vary. However, when considering a function in a specific context (instantiated), these intermediate steps must be addressed. They deal with the controllability and observability of the operation been carried out and can effectively allow the attribution of qualifications to the interaction.

In order to characterize the functionality of an object in a task, we identify the following components:

- *intended user(s)* who should interact with the object.
- *intended recipient(s)* of the interaction.

- *interaction process* describing actions defining how the object is to be used.
- *expectation/observation process* using different sensor modalities to monitor and evaluate.

Specifically in "cutting" butter with a knife, the interactive process can be identified by: the approach, the contact, the progressive intrusion of the knife into the butter up to reaching the point of separation of the part from the whole. The visual observation can guide the knife in the desired position, monitor the progression and conclude that the operation is completed. The tactile sensing can guide us to modulate the pressure to be applied (the butter might still be frozen) and observe, for instance, the compliance of the material.

The representation and identification of a functionality is a difficult process, not only since it has to define a dynamic behavior, but also, as pointed out, because of the many–to–many mappings between functionality and object. These mappings could be handled adequately using classification techniques from pattern recognition [Fukunaga, 1986; Kittler, 1986] if prototypical properties for a given functionality could be easily characterized and recovered from the object investigated. Both of these operations, however, are non–trivial.

A high–level description of a functionality is used for constraining the parametric and operational space of the interaction. Posing these constraints allows the investigation for a given functionality to be systematic and focused. Furthermore, it allows the system to utilize previously existing or recently acquired knowledge about the object and the environment.

A domain theory for a particular functionality can be specified by describing how the given functionality may be characterizable and provable from a set of properties and relationships. In order to obtain the properties from a specific context, the recovery process must be endowed with some exploratory procedures (EP), [Klatzky *et al.*, 1987]. However, the set of performable EP's may be insufficient or perceptually limited. In such a situation, the interactive task, describing the functional behavior we are trying to establish, can be employed to verify the applicability of the object. Then, by observing the interaction, the hypothesis about the applicability of the object can be established. Thus, one could think of the actual task of carrying out the functionality as the ultimate exploratory procedure to determine the properties and qualify the performance of interaction in the context at hand.

The empirical approach proposed allows us, by carrying out the operation which the functionality entails, to verify its satisfiability in the specific context. Furthermore, by actively observing, new relations amongst properties may be recovered allowing the representation to be extended. This extensibility is a form of learning or knowledge extraction. Thus, on the one hand, the domain theory provides the abstract procedural definition of functionality and means of constraining its interactive recovery. The experimental component, on the other hand, can be employed to verify a functionality and to identify contexts in which it holds.

To extract such properties one needs to be able to:

- Identify the contexts in which the functionality may be satisfiable.
- Identify the properties relevant for the performability of the interaction.
- Determine the range of values of properties as weighted by their performance.

Each group of contexts identify ranges of performance and relevance of material properties. These can then be employed to define a notion of centrality and typicality of a property by considering the correlation between properties and their relevance in the performance of the functional task.

5 Formalism for a Manipulatory Task

The description of a task must provide for addressing its observability through different sensor modalities. It must also handle an environment in which not all interactions and exact time occurrence might be defined and hence predictable in advance. To describe an interactive process we adopt the formalism provided by Discrete Event Dynamic System theory, (DEDS) [Ramadge and Wonham, 1989]. This formalism allow us to model the behavior of a system in which uncertainty, external observability, and non–determinism can be addressed.

According to DEDS theory the behavior of a dynamic system can be modeled as a non-deterministic finite automaton (NDFA). In such NDFA arcs identify **events** and states identify fragments of operational behaviors or logical states of the system. Thus a state can be defined in terms of state variables. Transitions to other states may occur when these variables reach specified values. For example in the motion of a robotic arm the set of state variables might be those needed to specify the position of the end–effector. The transitions to a new state could be represented by the state variables having obtained a particular value identifying, for instance, contact. In this example we would have two states, the first one identifying the motion of the end–effector and the second one identifying the contact state.

Events which allow the transition from a state to another may be disabled or enabled as a means of guaranteeing controllability of the system. In [Sobh, 1991] transitions between states are also assigned probability functions. These functions determine the probability that a given event has been asserted.

Any task can be described as a simple action or as a sequence of actions or subtasks. Then we can identify some of these actions to represent states of the system. While events identify changes in the variables describing the system, we distinguish the following sets of events.

- A *change in the state variables*, Δ, in which the value of one or more variables describing the event has reached a specified value.
- The *assertion of logical expressions*, Λ, possibly denoting groups of events.
- The *reaching of a guarded value*, $\mathcal{G}$, for one or more state variables to which a particular meaning has been attributed, such as safety conditions. (where $\mathcal{G}$ is actually a subset of Δ)

This partition is done for convenience of expression. Logical expressions, in particular, are here intended as means of clustering events and attributing a meaningful interpretation.

5.1 Automata Model for the DEDS

The set of the labels of the events is given by $\Sigma = \Delta \cup \Lambda \cup \mathcal{G}$. A **string** $s = \sigma_1, \sigma_2, \ldots, \sigma_k$ from Σ^+ describes a sequence of events. The **admissible** subset of strings from Σ^+ defines physically possible sequences of events which constitute a task. A **recognizer**, M, can be described as a NDFA consisting of a set of states, Q, an initial state, q_0, a transition function $\delta : \Sigma \times Q \rightarrow Q$, and a set of final states, Q_m (marked states). The set $\Sigma(q_i)$ designates the collection of events which are associated with state q_i. The set $\Sigma(q_i)$ is defined as $\Sigma(q_i) = \Delta(q_i) \cup \Lambda(q_i) \cup \mathcal{G}(q_i)$. A recognizer M_{t_i} will accepts the strings from Σ^+ describing a sequence of events denoting a task, t_i. In particular, M_{t_i} characterizes the task's procedural description.

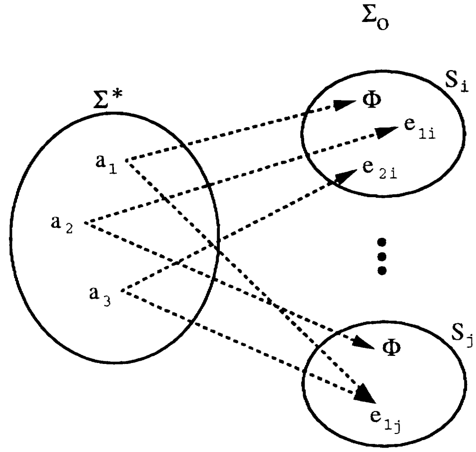

Figure 1: *Observable events, Σ_o, mapped to Sensors.*

5.2 Controllability

The set of events which we have identified above may include some which are *controllable* (that can be disabled) and some which might be *uncontrollable.* Thus, we can partition Σ into $\Sigma_c \cup \Sigma_u$. Enabling and disabling certain events can be described by the control pattern for the specific state. Let $\Gamma = \{0,1\}^{\Sigma_c}$ define the set of the binary patterns assignable to the elements from Σ_c. Then the function $\gamma : \Sigma_c \rightarrow \{0,1\}$ defines whether they are enabled or disabled.

The transition function δ above can now be defined as $\delta_c : \Gamma \times \Sigma \times Q \rightarrow Q$

$$\delta_c(\gamma,\sigma,q) = \begin{cases} \delta(\sigma,q) & \text{if } \delta(\sigma,q) \text{ defines and } \gamma(\sigma)=1 \\ \textit{undefined} & \text{otherwise} \end{cases}$$

Then the generator $\mathcal{G}_c = (Q, \Gamma \times \Sigma, \delta_c, q_0, Q_m)$ is called the **Controlled Discrete Event System**. Such a controller is called a *Supervisor*. Further details can be found in [Košecká, 1992].

5.3 Observability of the Interaction

A task t_i is **observable** if the sequences of events which defines it are observable. Figure 1 portrays an instance in which some of the events from a string from Σ^*, $\sigma = a_1\,a_2\,a_3$, are not observable and mapped to Φ. A projection function maps events from Σ^* to the individual sensors S_j's from $\mathcal{S}$ (set of sensors). Observability is contingent on the ability of monitoring the different events.

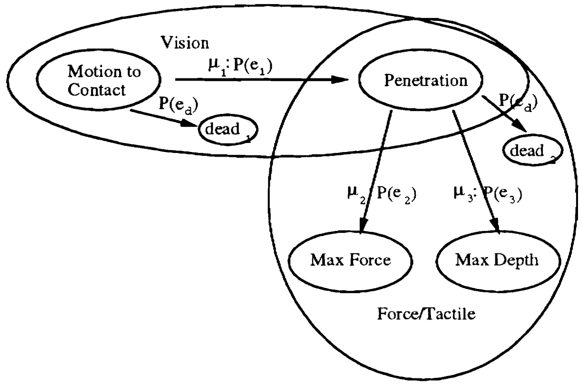

Figure 2: *Piercing Functional Task*

The observability of the individual events in the task must be guaranteed by the different sensor modalities if the overall task is to be observable.

A task t_i is **fully observable** if all of the events defining it are observable. Full observability is, in general, too strong a condition. If some of the events are not observable then some of the states may become indistinguishable (aliasing). To guarantee that the important events are not aliased, we identify states which must remain distinct, **distinguished states**. Not all the distinguishable states must be visited, but they must be unambiguously marked. If a task is partially observable by different sensors then it is **Piecewise** Observable. Redundancy in observing an event using more than one sensor can be employed to corroborate the evaluation of the observations. However, the application of more than one sensor modality goes beyond the issue of corroboration. In general, non–trivial tasks will require multimodal sensory observations.

Sensors are not always faithful and reliable informers and uncertainty has to be introduced in the system. In particular, it is important to be able to identify the uncertainty originating from sensor noise, from the environment, and from the detection of an event denoting a transition to a different state in the system. Thus, as described in [Sobh, 1991], we introduce a probability function associated with the occurrence of the events.

In Figure 2 we illustrate a high level description of a task for piercing a target object with a tool. The transition $\mu_i : P(e_i)$ indicates that a given transition is controlled by μ_i condition, and the occurrence of event e_i is characterized by some probability distribution P. Furthermore, we identify e_d as events which can not be controlled and which leads to a dead state. Such case could occur when a robotic arm is applying a force to push the tool into the target object causing

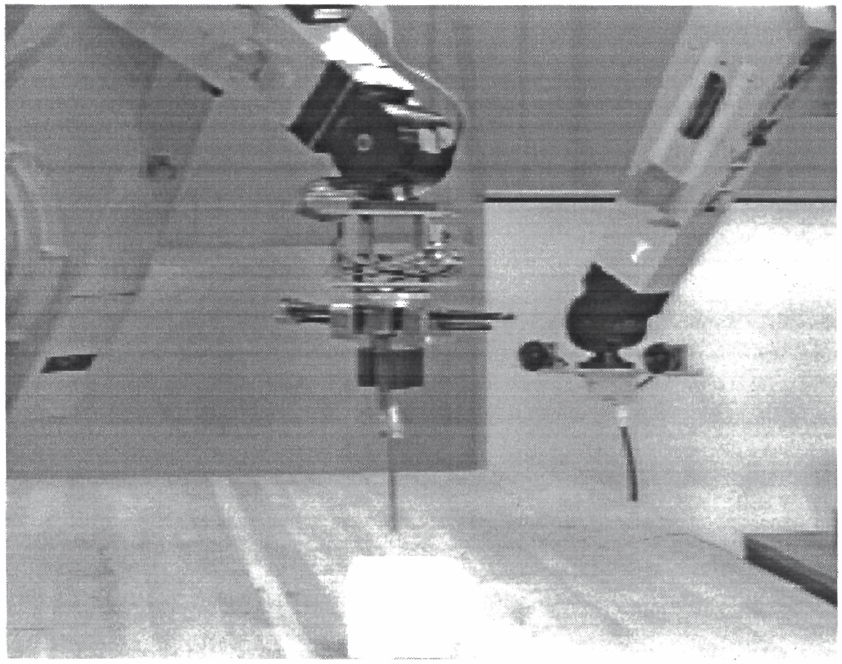

Figure 3: *Experimental Setup: (on the left) Puma with compliant wrist holding the tool; (on the right) Puma holding the pair of CCD cameras.*

it to shatter. From a dead state, the supervisor must provide means of returning the system to a controllable state in which the action may be repeated or a different course of action may be taken. As we can see in Figure 2, the overall task is piecewise observable by a vision sensor, and by a force/tactile sensor. It is, in fact, the combination of the two sensor modalities which make the whole task observable. We will return to discuss the behavior associated to the piercing functionality in section 6.

6 Primitive and Complex Functional Tasks

The interaction domain $\mathcal{A}$ can be constructed from an initial set of actions denoting functional tasks, $Ker(\mathcal{A})$, we define these to be the *Primitive* set. Complex tasks can be composed from a set of primitive tasks which have been fully explored. Furthermore, since the individual components are piecewise observable the resulting new action will be observable. We can define an algebra of tasks with the operations:

- *Composition* as the sequencing of a list of actions, we express that as $C(a_1, \ldots, a_k)$.
- *Repetition* as the composition of a given action $a_i \in \mathcal{A}$ with itself.

Sawing, for instance, could be easily seen as an operation in which composition and repetition occurs. In Figure 2 we exhibit a case in which two simpler functionalities have been composed into the task for piercing.

Extending $\mathcal{A}$ The newly generated actions are included in the action domain $\mathcal{A}$ by extending them to incorporate the new action. $\mathcal{A}_e = \mathcal{A} \cup \{ C(a_1, \ldots, a_k)\}$

Properties of Actions

- *(Closure)*: $\mathcal{A}$ is closed under composition and repetition.
- *(Action Decomposition)*: Any action $a_i \in \mathcal{A}$, $a_i \in Ker(\mathcal{A})$, can be expressed in term of compositions and repetitions of primitive actions.

The operations of composition and repetition are, however, not the only way in which $\mathcal{A}$ can be constructed and extended. Extensions can be created by modifying:

- The type of contact and the locus of contact between objects.
- The kinematics and the dynamic properties of the interaction.
- The type of operations carried out between the two objects.

Let's consider the difference between piercing and cutting in their simplest form of application. Both operations involve bringing a tool in contact with a target and both involve the application of force. However, the expectation of the result of the interaction, in its simplest form, differs. In the case of piercing we will have still one target object with a hole in it; in the case of cutting, instead, the target will have be partitioned into two objects. In the first case the type of contact is a point contact and in second case it is a line contact. If the locus of contact is varied cutting may become slicing. If a different amount of force is applied within the same period of time cutting becomes splitting. If the perpendicular application of contact in cutting is associated with a parallel translational component the result of the operation would now be carving or sawing.

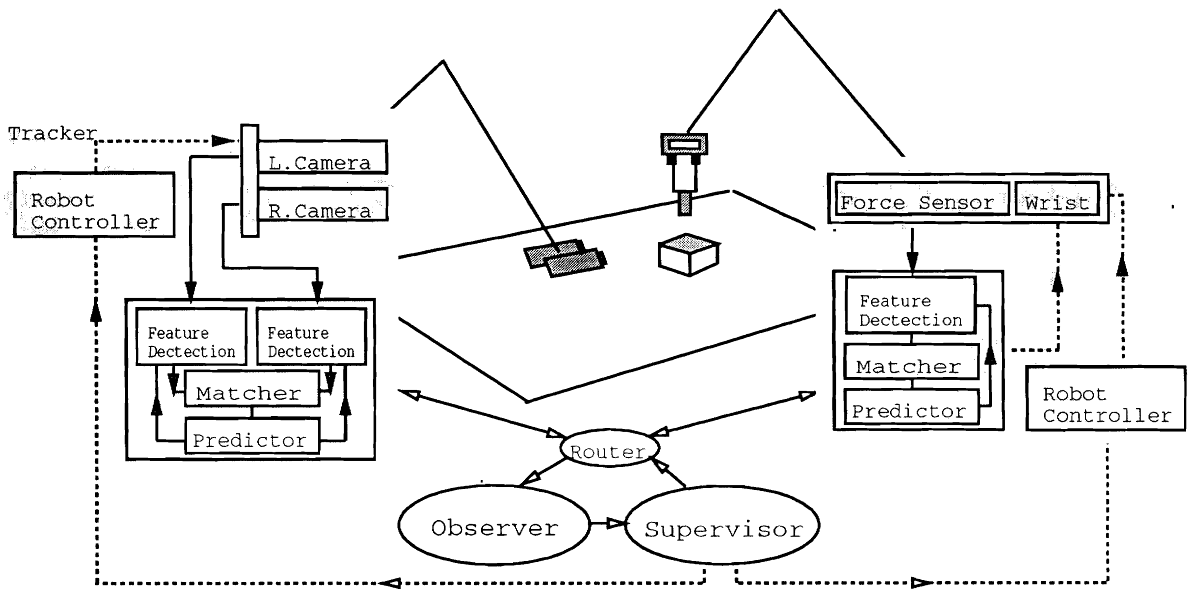

Figure 4: *Vision and Force/Tactile components and their connections to both External Observer and the Supervisor.*

The construction of the domain of functionalities is only outlined here. We plan to investigate it further subsequently.

7 An Overview of the Experimental System

We are currently developing a system for testing manipulatory functionalities which can be described and observed in terms of visual tracking and contact forces.

The vision component involves the tracking of an object by a pair of CCD cameras mounted on a Puma 560 arm. The two cameras also provide us with the ability to estimating the distance of the object from stereo. The contact sensor, a compliant wrist [Xu, 1989; Lindsay, 1992] with 6 degrees–of–freedom, is mounted on the end–effector of another Puma 560 arm holding the tool, Figure 3. The diagram the system is schematically described in Figure 4. The supervisor orchestrates the task through the feedback provided by the sensors and guarantees that the dynamic system will always be in a well defined state and that the correct sequence of operations will be performed. The observations from the individual sensors are also provided to an external observer whose purpose is that of evaluating the performance of the interaction.

In the case of piercing, the supervisor guarantees that contact be obtained and that incremental application of force without over–stepping critical thresholds. The observer qualifies the interaction by noticing features such as: the rate of penetration, the response of the target material (rigid, plastic or elastic deformation), the amount of force required to pierce the target object.

By varying the type of materials, the geometric shape of the tools employed, the amount of force applied, etc., we can learn about the functionality of a tool in a set of contexts and extract relevant functional properties identifying a higher performance in a given tool.

As we have stated in section 3, for each property, we can learn about its applicability in classes of contexts. This can be accomplished by systematically varying some parameters in a specific context and ranging the variations in a class of contexts. Having determined the ranges for the properties we can then define a measure over the properties reflecting their contribution to functional performance of the tool.

8 Conclusion

We have investigated what is meant by functionality in an object and presented an abstract task description and an interactive process and the necessity for recovering it. We have identified the representation of functionality as an interactive process in which four components can be defined. We have noticed the importance of recovering material properties, of observing and verifying the interaction and of evaluating its performance from an initial set of hypotheses. We have focused on the performatory component of the task and in particular examined the observability aspect. Finally we have developed a testbed for investigating functionality in objects.

We will need to expand on means for incorporating the result of the recovered evaluation of the interaction in the object description. In particular we would like to extend the system so that it can extract functional features, their importance, and degree from the specific interactions.

Acknowledgments This work was supported by Navy Grant N00014-92-J-1647, AFOSR Grants 88-0244, AFOSR 88-0296; Army/DAAL 03-89-C-0031PRI; NSF Grants CISE/CDA 88-22719, IRI 89-06770, and ASC 91 0813.

References

[Brady *et al.*, 1985] M. Brady, P.E. Agre, D.J. Braunegg, and J.H. Connell. The mechanic's mate. In *Advances in Artificial Intelligence: European Conference of Aritificial Intellingece*, pages 79–94, 1985.

[Brooks, 1980] R.A. Brooks. Symbolic reasoning among 3-d models and 2-d images. *Artificial Intelligence*, 17:285–348, 1980.

[Campos, 1992] M. Campos. *Robotic Exploration of Material and Kinematic Properties of Objects.* PhD thesis, University of Pennsylvania, 1992.

[Connell and Brady, 1987] J.H. Connell and M. Brady. Generating and generalizing model of visual objects. *Artificial Intelligence*, 31:159–183, 1987.

[Cutkosky, 1989] M. R. Cutkosky. On grasp choice, grasp models, and the design of hands for manifacturing tasks. *IEEE Transactions on Robotics and Automation*, 5(3):269–279, 1989.

[Davis, 1991] E. Davis. *The Kinematics of Cutting Solid Objects.* Technical Report 541, New York University, Dept. of Computer and Information Science, Courant Institute of Mathematical Sciences, 251 Mercer Street NY, 1991.

[Freeman and Newell, 1971] P. Freeman and A. Newell. A model for functional reasoning in design. In *Proceedings of IJCAI-71*, pages 621–632, 1971.

[Fukunaga, 1986] K. Fukunaga. *Handbook of Pattern Recognition and Image Processing*, chapter Statistical Pattern Classification, pages 3–32. Academic Press Inc., London, England, 1986.

[Goodall, 1986] J. Goodall. *The Chimpanzees of Gombe, Patterns of Behaviour*, chapter Object Manipulation, pages 535–564. Belknap Press of Harvard University Press, 1986.

[Grace, 1989] R. Grace. *Interpreting the Function of Stone Tools*, pages 9–16, 106–115. B.A.R. International Series, Oxford, England, 1989.

[Iberall *et al.*, 1988] T. Iberall, J. Jackson, L. Labbe, and R. Zampano. Knowledge-based prehension: capturing human dexterity. In *International Conference on Robotic Research*, pages 82–87, 1988.

[Jordan, 1991] D.S. Jordan. *The Role of Physical Properties in Understanding the Functionality of Objects.* PhD thesis, Stanford University, 1991.

[Kittler, 1986] J. Kittler. *Handbook of Pattern Recognition and Image Processing*, chapter 3 Feature Selection and Extraction, pages 60–81. Academic Press Inc., London, England, 1986.

[Klatzky *et al.*, 1987] R. Klatzky, S. Lederman, and C. Reed. *There's more to touch than meets the eye: The salience of Object's Attributes for haptics with and without vision.* Technical Report, University of California, Cognitive Science, Santa Barbara, CA, 1987.

[Košecká, 1992] J. Košecká. *Control of Discrete Event Systems.* Technical Report MS-CIS-92-35, University of Pennsylvania, Dept. of Computer and Information Science, Philadelphia, PA., 1992.

[Lederman and Klatzky, 1987] S. Lederman and R. Klatzky. Haptic classification of common objects: knowledge-driven exploration. *Cognitive Psycology*, 22():421–459, 1987.

[Lindsay, 1992] T.S. Lindsay. *Teleprogramming: Remote Site Robot Task Execution.* PhD thesis, University of Pennsylvania, 1992. Technical Report MS-CIS-92-64.

[Lowry, 1982] M.R. Lowry. Reasoning between structure and function. In *Proceedings fo the Image Understanding Workshop*, pages 260–264, Science Applications In., Mclean VA, 1982.

[Ramadge and Wonham, 1989] P.J.G. Ramadge and W. M. Wonham. The control of discrete event systems. In *Proceedings of the IEEE*, pages 81–97, 1989.

[Shirai, 1987] Y. Shirai. *Three Dimensional Computer Vision.* Springer-Verlag, 1987.

[Sinha, 1992] P. Sinha. *Robotic Exploration Of Surfaces and Its Application to Legged Locomotion.* PhD thesis, University of Pennsylvania, 1992. Technical Report MS-CIS-92-12, GRASP Lab 301.

[Smith and Medin, 1981] E. E. Smith and D. L. Medin. *Categories and Concepts.* Harvard Unversity Press, Cambridge, MA, 1981.

[Sobh, 1991] T. Sobh. *Active Observer: A Discrete Event Dynamic System Model for Controlling an Observer under Uncertainty.* PhD thesis, University of Pennsylvania, 1991. Technical Report MS-CIS-91-99.

[Stark and Bowyer, 1991] L. Stark and K. Bowyer. Generic recognition through qualitative reasoning about 3-d shape and object function. In *Proceedings of CVPR*, pages 251–256, 1991.

[Winston *et al.*, 1984] P.H. Winston, T.O Binford, B. Katz, and M. Lowry. *Learning Physical Descriptions from Functional Definitions, Examples, and Precedents*, chapter Connecting Perception to Action, pages 117–135. MIT Press, Cambridge, MA., 1984.

[Xu, 1989] Yangsheng Xu. *Compliant wrist design and hybrid position/force control of robot manipulators.* PhD thesis, University of Pennsylvania, 1989.