UQAC
UNIVERSITÉ DU QUÉBEC
À CHICOUTIMI

MÉMOIRE

PRÉSENTÉ À

L'UNIVERSITÉ DU QUÉBEC À CHICOUTIMI

COMME EXIGENCE PARTIELLE

DE LA MAÎTRISE EN INFORMATIQUE

PAR

KÉVIN BOUCHARD

B.SC.

QUALITATIVE SPATIAL REASONING FOR ACTIVITY RECOGNITION USING TOOLS OF AMBIENT INTELLIGENCE

MARS 2012

ABSTRACT

The aging population represents a growing concern of governments due to the extent that it will take in the coming decades and the speed of its evolution. This problem will result in increasing number of people affected by many diseases associated with aging such as the various types of dementia, including the sadly famous Alzheimer's disease. People with Alzheimer's must be assisted at all time during their everyday life. Technological assistance inside what is called a smart home could bring an affordable solution to solve this concern. One of the key issues to smart home assistance is to recognize the ongoing activities of everyday life made by the patient in order to be able to provide useful services at an appropriate moment. To do so, we must build a structured knowledge base of activities from which one or many intelligent agents (communicating with each other) would use information extracted from the various sensors to take a decision on what the inhabitant could be currently doing. The best way to build such an algorithm is to exploit constraints of different natures (logical, temporal, etc.) in order to circumscribe a library of activities. Many authors have emphasized the importance of the fundamental spatial aspect in activity recognition. However, only few works exist, and they are tested in a limited way that does not allow discerning the importance of dealing with space. Important spatial criterions, such as distance between objects, could help to reduce the number of hypotheses. Moreover, many errors can be detected only by using the spatial reasoning such as position problems (inappropriate objects are brought into the activity zone) or orientation of object issue (cup of coffee is upside down when pouring coffee).

This thesis provides potential solutions to the problem outlined, which deals with spatial recognition of activities of daily living of a person with Alzheimer's disease. It proposes to adapt a theory of spatial reasoning, developed by Egenhofer, to a new model for recognition of activities. This new model allows identifying the ongoing activity using only qualitative spatial criterions which we demonstrate through the text that some could not have been identified otherwise. It also allows detection of new abnormalities related to the behavior of an individual in loss of autonomy. Finally, the model has been implemented and validated in carrying out activities in a smart home on the cutting edge of technology. These activities were derived from a clinical study with normal and mild to moderate Alzheimer subjects. The results were analyzed and compared with existing approaches to measure the contribution of this thesis.

RÉSUMÉ

Le vieillissement de la population représente une préoccupation croissante des gouvernements en raison de l'ampleur qu'il prendra dans les prochaines décennies et la rapidité de son évolution. Ce problème se traduira par l'augmentation du nombre de personnes touchées par de nombreuses maladies liées au vieillissement telles que les différents types de démence, y compris la tristement célèbre maladie d'Alzheimer. Les personnes atteintes de la maladie d'Alzheimer doivent être assistées en tout temps dans leur vie quotidienne. L'assistance technologique à l'intérieur de ce qu'on appelle une maison intelligente pourrait apporter une solution abordable pour cette tâche. Une des questions clés inhérentes à ce type d'assistance est de reconnaître les activités courantes de la vie quotidienne faite par le patient afin d'être en mesure de fournir des services utiles au moment le plus opportun. Pour ce faire, nous devons construire une base de connaissances structurée à partir de laquelle un ou plusieurs agents intelligents utilisant l'information extraite des divers capteurs pour émettre une hypothèse ciblée concernant l'activité en cours de l'habitant. La meilleure façon de construire un tel algorithme est d'exploiter les contraintes de natures différentes (logique, temporelle, etc.) afin de circonscrire une bibliothèque d'activités. De nombreux auteurs ont souligné l'importance de l'aspect spatial fondamental dans la reconnaissance d'activité. Cependant, seuls quelques travaux existent, et ils sont testés de façon limitée qui ne permet pas de voir l'importance de considérer l'espace. Néanmoins, plusieurs critères spatiaux tels que la distance entre les objets pourraient aider à réduire le nombre d'hypothèses d'activités. Par ailleurs, de nombreuses erreurs peuvent être détectées uniquement en utilisant le raisonnement spatial, tel que les problèmes de type position ou d'orientation.

Cette thèse fournit des pistes de solutions aux problèmes décrits, qui traitent de la reconnaissance spatiale des activités de la vie quotidienne d'une personne avec la maladie d'Alzheimer. Elle propose d'adapter une théorie du raisonnement spatial, développé par Egenhofer, à un nouveau modèle pour la reconnaissance des activités. Ce nouveau modèle permet d'identifier les activités en cours en utilisant uniquement les critères spatiaux. Nous démontrons à travers le texte que certaines activités ne pourraient pas avoir été identifiées autrement. Le modèle permet également la détection de nouvelles anomalies liées au comportement d'un individu en perte d'autonomie. Enfin, le modèle a été implémenté et validé en réalisant des activités dans un habitat intelligent à la fine pointe de la technologie. Ces activités ont été tirées d'une étude clinique avec des sujets normaux et Alzheimer. Les résultats ont été analysés et comparés avec les approches existantes pour évaluer la contribution de ce modèle.

ACKNOWLEDGMENTS

The research project presented in this master thesis is part of the activities of the Laboratoire d'Intelligence Ambiante pour la Reconnaissance d'Activités (LIARA) at the Université du Québec à Chicoutimi (UQAC). I would like to use this opportunity to thank those who helped me throughout this project. First, I would like to express my most sincere gratitude toward my master director and co-director, Abdenour Bouzouane and Bruno Bouchard, for their continued support, for the encouragement to surpass myself and for their patience. Their valuable advices were a key element in the success of this project and for this, I am indebted.

Financial (in the form of scholarship) and in kind support received from the Natural Sciences and Engineering Research Council of Canada (NSERC), the Fonds Québécois de la Recherche sur la Nature et les Technologies (FQRNT), the Laboratoire d'Intelligence Ambiante pour la Reconnaissance d'Activités (LIARA), the Université du Québec à Chicoutimi (UQAC), the Département d'Informatique et Mathématique (DIM) and the Fond de Campagne de développement de l'UQAC.

Finally, I would like to express my warmest thanks to my life partner Andréanne Simard, whose indescribable patience, help with French writing and constant encouraging words were essential to the completion of this project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF ACRONYMS

AD: ALZHEIMER DISEASE

ADL: ACTIVITY OF DAILY LIVING

AI: ARTIFICIAL INTELLIGENCE

ALZ: ACTIVE LEZI

BADL: BASIC ACTIVITY OF DAILY LIVING

GIS: GEOGRAPHIC INFORMATION SYSTEM

GUI: GRAPHICAL USER INTERFACE

HMM: HIDDEN MARKOV MACHINE

IADL: INSTRUMENTAL ACTIVITY OF DAILY LIVING

JEPD: JOINTLY EXHAUSTIVE AND PAIRWISE DISJOINT

KTA: KITCHEN TASK ASSESSMENT

LED: LIGHT EMITTING DIODE

LIARA: LABORATOIRE D'INTELLIGENCE AMBIANTE POUR LA RECONNAISSANCE D'ACTIVITÉS

NAT: NATURALISTIC ACTION TEST

QSR: QUALITATIVE SPATIAL REASONING

QSRR: QUALITATIVE SPATIAL REASONING RECOGNITION

RCC: REGION CONNECTION CALCULUS

RFID: RADIO-FREQUENCY IDENTIFICATION

RIMER: RULE BASED INFERENCE METHODOLOGY USING EVIDENTIAL REASONING

SR: SPATIAL REASONING

CHAPTER 1

INTRODUCTION

1.1 RESEARCH CONTEXT

In the province of Quebec as throughout the Western world, the population is considerably aging. For now, we can only try to predict the multiple impacts this will have on our societies, but it is now undeniable that we will have to deal with new challenges in the decades to come [1]. One of them is the increasing number of people suffering from a type of dementia such as Alzheimer's disease [2]. These people suffer a gradual deterioration of their cognitive abilities over a period ranging from three to ten years, causing the loss of their autonomy and hence, their ability to take care for themselves. Therefore, at a certain stage in the evolution of the disease, these people must be assisted continuously for the rest of their lives [3]. The researches on Alzheimer are progressing, and several breakthroughs have been made in identifying the various possible causes of it. However, in the near future, the physicians have no options to cure or prevent Alzheimer's. We can accurately estimate that the disease will continue to ravage families for quite a number of years. Finally, the increase in the number of people requiring ongoing assistance combined with an aging population may cause a shortage of trained health workers, which will have the effect of causing enormous stress on our already fragile health system. Hence, it appears necessary to find technological solutions to address this complex problem.

The evolution of information technology and electronics now makes it possible to envisage different approaches to address this societal transformation. Technological assistance inside a home qualified as smart has positioned itself as a significant trend [4] giving a new hope in the effort to postpone the institutionalization of the elderly. A smart home can be seen as a technologically enhanced environment using sensors (e.g. electromagnetic contacts, motion detectors, touch pad, radio-frequency identification tags, etc.), processors integrated into the miniature objects daily living (fridge, coffee maker, clothing, heating, dishes, etc.) and intelligent software agents communicating with each other in a goal of cooperation in the sense of multi-agent systems [5]. These environments must take decisions while taking care to limit their intrusion in order to help the resident to perform its tasks without invading their privacy. For example, if a stove burner is open, the device, or rather the artificial agent associated with it, must have a good idea of the behavior of the occupant as the context in which it takes place (preparation of a meal) by communicating with other agents. Also, suppose the agent observes that water boils for over an hour due to an oversight by the resident related to his cognitive impairment. Then, it could ask the main system to assist him by sending a vocal or video prompt, or using a more discrete media (light, emoticon, beep, etc.) [6]. The message type must be chosen carefully in order to stimulate the brain reactivity of the individual so that he corrects himself his mistake. When continuous support is provided to a patient with Alzheimer, cognitive degeneration of the disease is slowed and the patient can remain independent longer [7]. The significant advances of Artificial Intelligence (AI) are driving us toward the application of techniques resulting from fifty years of research to this new stimulating problem that is the technological assistance of people with declining autonomy.

The first fundamental step is to be able to understand the ongoing Activity of Daily Living (ADL) of the inhabitant in order to identify potential problems that may interfere with its accomplishment. This difficulty is, in fact, a special form of a well-known problem in artificial intelligence, which is called plan recognition [8]. A plan corresponds to a sequence of elementary step representing a certain ADL. In our application context, the recognition of plans intends to interpret the behavior of a person to provide, timely, appropriate services without being rejected by the individual. It is why a growing community of scientists [9-11] like the UQAC team at the Laboratoire d'Intelligence Ambiante pour la Reconnaissance d'Activités (LIARA) [12, 13] are currently working on this specific problem of recognizing ADLs inside a smart home.

1.2 SPATIAL REASONING AND ACTIVITY RECOGNITION

Activity recognition consists to circumscribe a library of plans to extract a minimal set of assumptions. As shown on Figure 1.1. it is based: (i) on the observation perceived through the sensors as a result of interactions (actions) of the person with the environment, (ii) on a principle of selection of hypotheses (possible activities), and (iii) on a matching method between the observations and the plans from the library describing the activities that are potentially observable [12]. This library is an ontology that describes in a logical formalism, the concepts of actions and activities. Then, the term activity recognition refers to the fact that we presuppose the existence of an activity structure planned at the beginning by the observed entity (the patient). The result is that the assistant agent seeks to recognize from its knowledge base (library of plans). The objective of this agent is, after the recognition process, to be able to predict as accurately as possible future actions of the patient by minimizing the number of possible hypotheses [14].

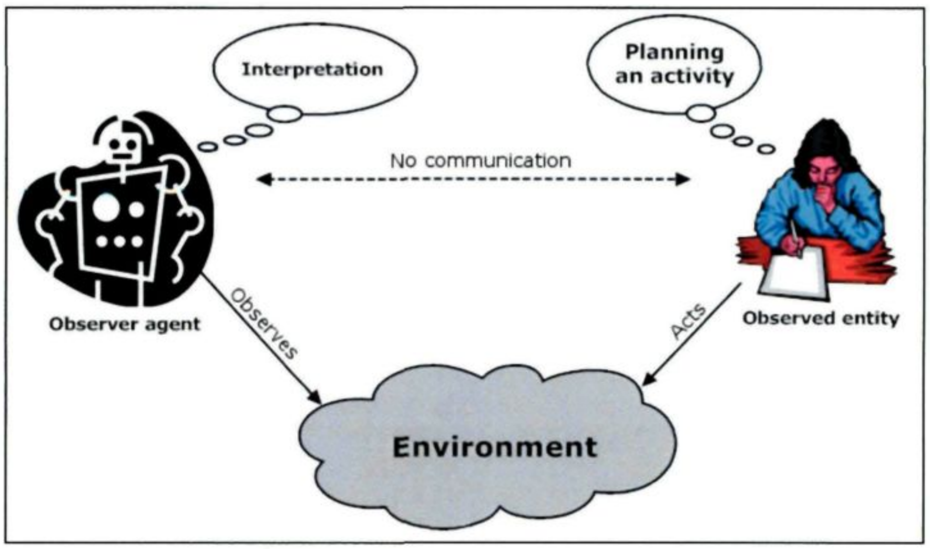


Figure 1.1: Simple schema illustrating the recognition process [15]

To achieve the recognition we have described, several discriminatory constraints of different natures (logical, probabilistic, temporal, etc.) must be used from which a shortlist can be made. For example, the activity *CookPasta* could be composed of the steps *BoilWater* and *PutPasta* having a sequential constraint specifying to do *BoilWater* before *PutPasta* (temporal constraint) [16]. A spatial constraint can be defined as the spatial state of one or many objects in relation to their environment meeting certain conditions. In other words, this corresponds to a condition of spatial nature on one or more objects to fulfill the conditions for carrying out an activity. Spatial reasoning is the natural way of thinking of human beings and should be considered in the analysis for the recognition of ADLs. It should also be understood that an activity can be performed in a valid sequence in time, but still be incorrect due to problems of spatial nature. For example, an activity may seem correct, but due to bad orientation of an object, execution is wrong. The same thing can happen if each step is correctly detected, but at some point, an object is moving away rather than closer to the activity's zone (the problem of distance). Therefore, it is of crucial importance to consider spatial aspects. However, the approaches in the field of activity recognition generally ignore it, even if they recognize its important role [17]. Moreover, most of the existing works are largely theoretical and not tested or only experimented in a non-realistic context that does not allow determining their actual effectiveness [18].

1.3 ILLUSTRATION OF THE IMPORTANCE OF THE SPATIAL ASPECT

To illustrate the importance of exploiting the spatial information embedded in the applicative context of activity recognition inside a smart home, we present in this section a small scenario highlighting the limits of a system ignoring this aspect for recognition and for assistance. Since a significant number of examples concern kitchen activities as washing hands [11], cooking meal [12] and preparing tea or coffee [19], it seems a good lead to choose a similar example. We also need our activity to be simple, but be composed of at least some steps. Finally, many authors like [20] have identified important criterions to describe spatial relations, and so we will use them for our example.

Let be Peter a fictive elder suffering from a mild to moderate degree (stage 3 or 4) of development of the Alzheimer disease. Our fictive subject lives in an intelligent house possessing cutting-edge technology and a multi-agents software. The agents analyze the actions of Peter in order to assist him in his tasks if needed. Peter takes out a kettle from a cabinet and fills it with water. The system receives a signal from Radio-Frequency IDentification (RFID) tags detecting the movement in a three dimensional coordinate in space. Then, it detects a change in the flow of tap water (increasing) using an ultrasonic flow meter. It associates this signals to a basic action *BoilWater* that is part of two different activities in the library: *MakeCoffee* and *MakeTea*. These activities are very similar, and both begin with the three same basic actions: *BoilWater*, *GetCup* and *FillCupWithWater*. They can only be distinguished at the step of putting the tea or coffee inside the cup. However, Peter does different actions that do not perfectly follow one of the two plans. He moves both the coffee and the tea. Considering only that information, it is impossible for the algorithm to differentiate between these activities. If we add the notion of **distance** between objects of the environment, the recognition system could perceive that the kettle, the coffee and the cup are near, forming a coherent group, and that the tea is farther, and hence probably not involved in the activity. Moreover, distance criterions would avoid the system from wrongly believe the ongoing activity is *MakeTea* if the subject had moved the tea before the coffee because it would have observed the distance between the activity area, and tea is increasing or too large anyway.

Using the previous example, Peter has now correctly carried out the entire required steps to the activity *MakeCoffee* in the right order. He is now going to pour the coffee into his cup. Everything seems fine for a recognition system that ignores the **orientation** of objects in space. However, the cup could have been upside down resulting into an unfortunate situation that could even lead to injury (burns) if no action is taken to correct the problem. Even thought dangerous problems related to orientation are not frequent, there are many other situations where considering it allows to discover a problem during ongoing activity. For instance, a carton of milk could be on the side, and the system could respond quickly enough to avoid a big mess.

Peter is now drinking his coffee and decides to store the coffee pot. In a glimpse of time, he forgot what he was going to do and went to the bathroom instead where he put the pot on a shelf and forgot it. This time, the system without spatial reasoning sees an action, and it could associate it with a plan or not, but it will not consider the possibility that it could be the result of a problem. On contrary, a system considering space would possess a set of **position** rules. Thus, when observing the movement of coffee pot in the bathroom, it understands that action could not be part of a coherent activity, and that it is an error because a pot of coffee should never be in the bathroom. Then, an assisting agent could prompt the person to help her find the object for later activity or to make her solve this erroneous situation right now.

1.4 Related works on spatial reasoning and recognition

The importance of the problem of activity recognition in the context of cognitive assistance inside a smart home has provoked a recrudescence of the works on this particular field during the last few years [11, 21]. Despite this, we note that most of the work focused on the field of smart homes offer rigid recognition models that do not take into account the spatial aspect, or that incorporates it in a very limited way [22], reducing their effectiveness in the matter of ADLs recognition or in the matter of identifying mistakes of spatial nature of a person with cognitive impairment. Very few other works have tried to include this aspect to improve recognition performance. Most of them only integrated little aspect from spatial reasoning such as the subject movements in the smart home [23, 24]. However, a lot of work has been done on the side of classical theories for reasoning with space, and they will be presented in detail.

From these work, it is essential to talk about Egenhofer & Franzosa [25] framework that was created with the intended goal of improving Geographic Information System (GIS). Their work is based on the mathematical principle of general topology that the object is to study and describe space of all dimensions. In their work, they demonstrate two very important things. First, the description of spatial relations in terms of set properties as topological invariant is simple. Second, they demonstrate that every topological spatial relations fall within that same framework. Finally, their model is easily extensible and can be used for various applications.

The second framework of appreciable importance is from Cohn [26], and it is named the Region Connection Calculus (RCC). That model describes spatial relations with a vocabulary containing a lot of similarities issued from Allen temporal framework [27]. On other hands, it uses a visual representation following the set-theories similar to the model of

Egenhofer & Franzosa. The RCC use *region* as base entity for his calculus and describe the space in a simple matter. It is derived in a couple of variants (RCC8, RCC23) where the names signify explicitly the number of possible relation types for two objects in relationship. The RCC is particularly interesting for dealing with relation between objects of various shapes notably those with concave parts. However, it does not describe very well the notion of power in a relation between two objects.

Excepted from the classical reasoning models, we also have found some approaches using the spatial aspect for activity recognition. For instance, Augusto [24] built a spatio-temporal logical algorithm based on a rule inference system. His system was not covering the same aspects that we did. In his experiments, he tried to recognize patterns of a person not moving. This is indeed useful to know if the person has fainted or if this behavior is normal. This works show that spatial constraints are undoubtedly important. However, that system was not built in the goal of recognizing all possible activities using spatio-temporal constraints but to demonstrate that smart homes could benefit from considering these aspects when designing them. Moreover, it is a very simple instance of what we present in this master's thesis with spatial consideration.

Another model comes from Riedel [23] following the same direction line than most works that deal with space in activity recognition. It incorporates the spatial aspect in a blind method. His model presents a learning method based on chemotactic bacteria that can recognize activity from spatial data extracted from a video imaging system. In this model, it is not specified what spatial criterions are taken into account for the recognition. So even though it gave good results, it only partially addresses the issue of recognizing ADLs from spatial point of view and completely ignore the problems related to space that can be caused by cognitive degeneration.

1.5 CONTRIBUTIONS OF THIS THESIS

The contribution of this master's thesis follows in the footsteps of spatial reasoning and activity recognition approaches that have been cited in the earlier section. This thesis tries to make a step forward by providing answers to the questions raised that are related to spatial aspects. The contribution is threefold: theoretical, practical and experimental.

At the theoretical level, we propose an extension of the fundamental framework of spatial reasoning of Egenhofer & Franzosa [25] by adapting it to a new activity recognition model. This robust framework has proven itself in the field of GIS and it is based on the solid foundation of general topology. The addition to the theoretical model is in the definition of a method to calculate the plausibility of ADLs and the definition of a library of plans based on spatial constraints between objects, activity zones and plans. In addition, we repatriate knowledge from years of research on spatial reasoning to address a classic problem of AI. That important information allows us to define the spatial context of activity recognition by exploiting existing model of spatial reasoning.

Second, from a practical standpoint, we have implemented this new model to evaluate how it would perform in a realistic context. This was done as multi-agents software on a computer communicating with a real intelligent habitat at the LIARA laboratory. This infrastructure was constituted of many different kinds of sensors on the leading edge of technology. It also contributed to the validation of RFID technology as a valid solution for the purpose of recognizing ongoing activity inside home.

Finally, this work contributes to the experimental knowledge with the presentation of rigorous experiments that were conducted using data extracted from clinical trials with both normal and Alzheimer's subjects. With this information, we defined real case scenarios of many ADLs incorporating errors or not. These scenarios could be a good lead for other teams that would like to conduct similar experiments in the future.

1.6 RESEARCH METHODOLOGY

The research project presented in this thesis was carried out by following a research methodology divided into four key steps.

The first phase of the project aimed to gain knowledge of the targeted area of research by conducting a review of the literature on the problem of activity recognition in general [8, 28, 29] and in particular, the approaches exploiting the spatial aspect [23, 30]. A study of the main formal frameworks in spatial reasoning, especially the theories of Cohn [31] and Egenhofer [25], was also performed in order to be able to understand the works explored. The first part has allowed having an overview of the field of activity recognition, particularly in an applicative context of technological assistance of people with reduced autonomy. It has helped to identify issues and specific needs of a recognition model designed for this purpose within a smart home. The second part of this phase aimed to achieve a state of the art focused on existing recognition models that exploit spatial information in their recognition process. This part has allowed arriving at potential solutions that have led to the proposed contribution of this thesis.

The second phase consisted of the extension of a formalism of description of spatial relation to a new model of activity recognition by establishing new theoretical basis to solve the issues explained in the earlier sections. For that purpose, it has been decided to extend Egenhofer & Franzosa framework. The main reason is that it has proven itself as a well establish model, but also because it is based on general topology. That theory is known and well-respected in the domain of mathematic. The work of Egenhofer & Franzosa is also more general than Cohn's region connection calculus and thus easier to adapt to the problem of research. It allows abstracting a notion of "power" in a relationship between one or more objects.

The third phase was consisting into a software implementation of this new formal model of recognition in order to validate its performances and to establish a comparison basis for the other recognition approaches, especially those not integrating spatial constraints. To do so, we have chosen to develop it using Java programming language running on a standard personal computer. The application was directly communicating with a real smart home infrastructure full of sensors and effectors at the LIARA laboratory. It was decided to primarily use RFID technology to deal with the spatial aspect in our model. Further details will be provided in chapter 4.

The last phase of this project of research consisted in the validation of the new model created (and implemented). At the same occasion, it has the purpose of verifying the usefulness of spatial constraints for the process of recognizing ongoing activity. The first step of this phase was to gather important information in order to construct scenarios of activity for the tests. Since we wanted both normal scenarios and scenarios containing spatial errors, we built the scenarios from real data extracted from clinical trials conducted in parallel by the LIARA's team. These trials were done with normal and Alzheimer subjects executing ADLs from well-establish cognitive test. From that we conducted experiments on 78 execution tests of 26 different scenarios. The results obtained will be described in detail while highlighting differences and similitudes with other recognition approaches.

It might be noted that this master's project was the object of two scientific publications. The first one was a short paper (5 pages) [32], a first draft of the model, presented at the occasion of the 9th International Conference on Smart Homes and Health Telematics (ICOST'11) and published by the renowned Springer. The second one was an improvement, with new results, and a long paper of 10 pages [33] presented at the 2nd International Conference on Ambient Systems, Networks and Technologies (ANT'11) and published by Elsevier. That encouraging recognition from scientists in the field support the conclusion of this thesis and the importance of the works realized by our team, and the results obtained.

1.7 THESIS ORGANIZATION

This thesis is organized into five chapters that fall in direct chronology with the research methodology. The first chapter that is ending was intended as an introduction to the thesis, to our context of study and to the issues raised in this research. It has provided a summary of the problems related to space, and a comprehensive example to illustrate the importance of spatial reasoning.

The second chapter provides an introduction to formal tools of qualitative spatial reasoning in artificial intelligence. In particular, it describes what is the spatial reasoning and its qualitative field. It illustrates the importance in the context by providing significant examples of common situations in the lives of people with cognitive disabilities. This chapter is a state of the art focused on the main approaches to spatial reasoning identified in the first phase of our research. It begins by the description of some elements from the research in natural language area. Next, the chapter talks about Allen's temporal model because it constitutes a basis of several works in spatial reasoning. Also, his work can be improved to itself become a model of qualitative spatial reasoning. Then, both the model of Cohn and Egenhofer & Franzosa will be presented in detail. These models possess a lot of similarities and will be compared together. At the end of the chapter, we conclude with a comparison between advantages and disadvantages of the different models and justify our decision to extend the one of Egenhofer & Franzosa.

The third chapter is about the field of activity recognition and the current existing approaches. In a first time, it will be further described what is activity recognition and the history of the research on it. Then we will describe the different family of recognition approaches classified on a constraint type basis. For each family, we will review important works and their advantages and limitations. The first type of approach presented will be the works based on classic logical constraints. These are the oldest and the more mature of all, and they regroup a large variety of approaches. The second type presented will be the temporal based approaches, and the last one will be the spatial based approaches that are the less frequent. The chapter will conclude with an assessment of the different works to better situate our contribution.

The fourth chapter examines the theoretical, practical and experimental contribution resulting from this research. The first section describes the formal elements to extend the model of Egenhofer & Franzosa in order to incorporate it to an algorithm for ADLs recognition. The second section shows the implementation of this new model of recognition within an intelligent home based on RFID technology. The final part of the chapter discusses the process employed to validate the proposed model and presents a comparative analysis of results obtained with the main approaches in the field.

Finally, the fifth and final chapter concludes the thesis by presenting a detailed account of the research project highlighting the contribution of this work over previous works. This chapter will also address the limitations of the approach and future works arising from this research. The chapter concludes with a more personal assessment of this experience of initiation into the world of scientific research.

CHAPITRE 2

SPATIAL REASONING

2.1 INTRODUCTION TO SPATIAL REASONING

Spatial reasoning (SR) is a large field of study interested to the fundamental methodologies to describe space, to reason and to understand it. That field as been vastly studied at first in natural language[34, 35] with the spatial prepositions human use to communicate spatial information to each other (under, beside, etc.). It is not a specific field of computer science but rather a field from which it is based to solve everyday problems. A large part of it has been developed as early as in the beginning of the 20^{th} century by the advent of recent development of general topology [36] whose history dates as far as back in the 18^{th}. Spatial reasoning is a pervasive field that was largely exploited in the geographic information systems (GIS) [25]. It is also much used in robot-human interaction research [37] similar as what we found in natural language researches, but also for the navigation. Nowadays, it is also widely used in video game [38] that is one of the most applicative branch of SR. That field provides a good way to test our knowledge in an environment simpler than reality. We believe that, although it is a more complex situation, smart home could be a new opportunity for that field of research to apply the accumulated years of knowledge to a new exciting problem.

It would be very interesting but too long for the purpose of this thesis to talk about all these different works. Therefore, this chapter will mostly concentrate on the principal formal methods of SR. It is also very important to note that the approaches we will present are classified as Qualitative Spatial Reasoning (QSR) as opposite to the classical quantitative spatial reasoning. We will discuss at the end of the chapter, the differences and the implication of each type of SR. Though, it could be noted that QSR has begun with the advent of Allen's temporal logic [39] that is often said the one dimension equivalent of QSR.

2.1.1 WHY SPATIAL REASONING MATTER IN ACTIVITY RECOGNITION?

Before explaining in details all the approaches of spatial reasoning, it would be a good idea to first look at why SR is interesting in activity recognition. To answer this question, we must look back in a mirror to see how humans act in their environment. That is where it becomes interesting, human always deal with spatial aspects in their everyday life. It is almost as natural as breathing; we think, we reason, and we talk using spatial aspects. In our language, we use qualitative expressions such as "Take the keys on the table" to add a layer of information on how one can retrieve the keys. When taking decision, we will almost always take spatial criterions into consideration. Whereas choosing between plates containing a part of pie, we will choose in function of the size and in function of the natural proportion displayed by the element constituting the pie. Gestalt laws [40] teach us that humans will always tend to equilibrate elements in their environment. This applies in the realization of our ADLs. We will tend to organize required objects in logical structure. We will group tools to form a whole. So, when observing a human, there will be many spatial traces of what is his intended goal and we, human, would certainly use that information to infer the current ongoing activity as an observer. Thus, it is also crucial for computer AI to take this precious information into account if we want them to understand us.

Figure 2.1: Could you infer the correct object without spatial reasoning?

That is only the first part of the answer. We also need to take a look at the different kinds of situations where spatial criterions could help us identifying the ongoing activity or mistake made by the resident. In his paper, Cohn [20] listed different spatial criterions that should be considered: distance, position, orientation. From these criterions, we defined many situations (but not exhaustive) that could lead an activity recognition system (without

SR) to a false conclusion. In many of these examples, the resident would require immediate help to complete his activity correctly (and safely). The figure 2.2 shows these situations and the criterions they belong to.

Figure 2.2: Spatial criterions and the situations identified

2.1.1.1 Situations related to distance

Absence/Presence: Some objects can be missing or can be wrongly replaced by another type. A cup could be replaced by a glass or a spoon by a knife. These organizational errors could be detected by considering the distance between objects, and it would be possible to recognize the ongoing activity if enough objects are present.

Wrong increase: The distance from the activity zone of an object is wrongly increasing during an activity. For example, the subject could store the coffee before having done the step *PutCoffeeInTheCup*. It could not be possible to detect such incoherent behavior without consideration for the distance criterions.

Grouping: Many objects can be detected in the zone such as coffee and tea. Those could have both been moved by the person thus generating event for the recognition algorithm. The only way to distinguish between activities would be to compare distance with other objects and with the subject. If the tea is nearer to a cup and a spoon, it would probably mean that *MakeTea* is the ongoing plan.

2.1.1.2 Situations related to positioning

Object position: An object is moved in an area it should never be. Imagine the resident goes into the washroom and then brings back the shampoo bottle with him in the kitchen. This object is certainly not part of the current activity. If the position issue is not considered, a recognition algorithm could use this event and then miscalculate the plausibility of activities (increasing *WashingHair*, for instance). Moreover, a bad position could even be dangerous and thus should be avoided (something wrong in the oven).

Subject position: The subject position can help in determining the correct ongoing activity. This information can also be combined with temporal information to identify activity interruption. For example, the person could have gone to her room while cooking pasta, but if it takes too long that could result into a dangerous situation.

Interleaved activity: At a certain degree, spatial reasoning can help to identify interleaved activities. Imagine the subject is preparing pasta, and then he goes to his room, take a book and start to read. Considering the current position of the subject, the latest object movement and the relation in the subject zone, a spatial algorithm could conclude that the plan *CookPasta* has been interrupted to *ReadBook*.

2.1.1.3 Situations related to orientation

Wrong orientation: An object with a wrong orientation could result in a failure of an activity. It could even result in a dangerous situation for the resident. For example, if a cup of coffee is upside down before pouring the hot beverage, the person could burn itself. Such a matter seems simple for us but could be an issue for AD person.

2.2 EXAMPLE OF WORK IN SR FOR NATURAL LANGUAGE

The study of spatial reasoning in the field of natural language is generally used for development spatial query languages or for the improvement of computer-human interaction. Although at first glance this work does not seem to respond to our problem, it is interesting to look at what knowledge this particular branch can bring to activity recognition. Some criterions they define allow the understanding of certain aspects of humans that could lead to build better models of the spatial environment. Work of Retzschmidt [35] is an important example of that. It teaches us how human use spatial prepositions to describe the topological relationship between objects but also how it gives information about orientation and direction (at the left of, under, behind, etc.). These spatial prepositions can be classified in three distinct categories: deictic, intrinsic and extrinsic. To illustrate the difference, we take back the example from Retzschmidt that we illustrated on figure 2.3: "The ball is in front of the car". In case **A**, the ball is situated in function of the speaker; that is a deictic preposition. In case **B**, the object is situated in function of the orientation of the car itself; that is an intrinsic preposition. In case **C**, the ball is positioned in relation to the car direction; an extrinsic preposition.

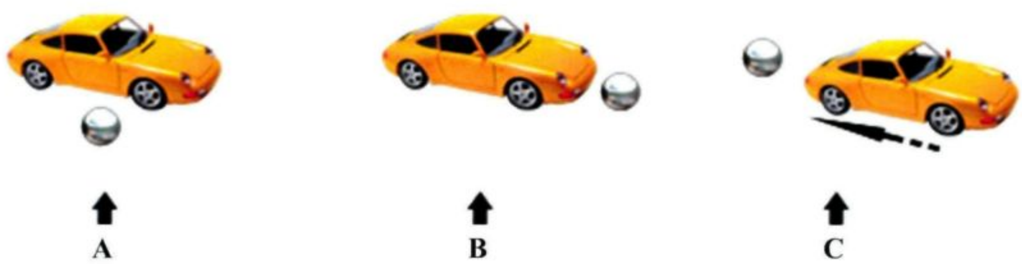


Figure 2.3: Illustration of deictic (A), intrinsic (B) and extrinsic (C) points of view

A good spatial reasoning model should be precise enough to prevent confusion about those three points of view. Human generally considerate (by default) that a spatial description is deictic if not otherwise precised. Knowing that could be useful for QSR real application.

2.3 ALLEN'S TEMPORAL RELATIONS FOR QSR

Before talking about the works using time to QSR, we should first introduce the basic notion surrounding temporal reasoning. Allen's theory of time is well-known in AI as the interval temporal logic [39]. For him, time was an infinite line constituted of an infinite number of points. The smallest unit of time to grasp in his logic is, however, what we call an interval. Such interval is a section somewhere on the infinite line of time. That concept is illustrated on figure 2.4. That logical unit can represent a certain amount of time for an event or an action such as *OpenDoor*. Many intervals can represent each action of a complete activity or be combined to represent that whole. For instance, we could have an interval for *MakeCoffee*, and we could subdivide it into smaller chunks corresponding to the actions *PrepareCup*, *BoilWater*, *AddWater* and *AddMilk*.

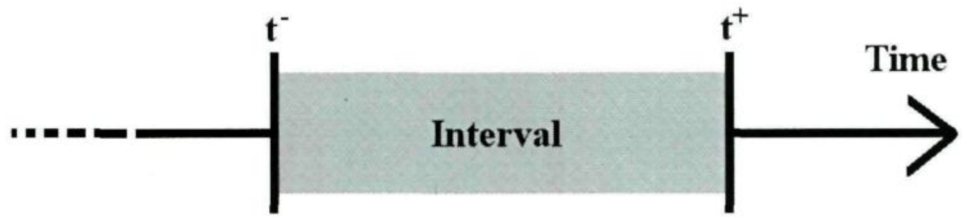


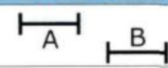
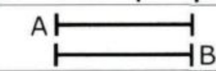
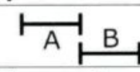
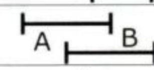
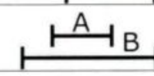
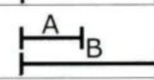
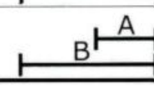
Figure 2.4: Representation of an interval of time

On a more formal approach, interval based systems are symbolic systems where an interval is defined as a sorted set of points T such that $(\exists t^-)(\forall t \in T)(t^- < t) \wedge (\exists t^+)(\forall t \in T)(t^+ > t)$. That means an interval possesses a pair of limit points (t^-, t^+) marking the beginning and end of the defined interval.

2.3.1 INTERVAL BASED RELATIONS

Based on the introduced notion of interval of time, Allen has defined thirteen Jointly Exhaustive and Pairwise Disjoint (JEPD) relations that can exists between two distinct time event (interval). These relations can be seen on the table 2.1 with their corresponding meaning and symbols. On that image, A and B are two arbitrary events that happen at some time. Each of these events is unique, corresponding to a different interval of time and thus possess is own pair of time points (t^-, t^+). The line located under the letter A characterize the relative duration of this event and in relation in function of the event B. The nonsymmetrical relations possess an inverse where an *i* is added to the symbol (except for < whose inverse is >).

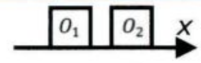
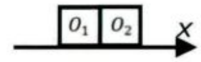
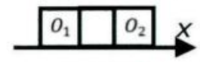
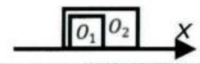
Table 2.1: The thirteen possible relations between two intervals

Relation	Symbol	Inverse	Illustration
A before B	$<$	$>$	
A equal B	$=$	$=$	
A meets B	m	m_i	
A overlaps B	o	o_i	
A during B	d	d_i	
A starts B	s	s_i	
A finishes B	f	f_i	

2.3.2 ALLEN'S RELATIONS FOR 1D SPATIAL REASONING

Many authors have used Allen's temporal relations for spatial reasoning [41, 42] and perhaps [43] was one of the first and the most interesting. An analogy can be made between Allen's temporal reasoning and spatial reasoning. In fact, one can view his logic as a way to do QSR on a one-dimensional basis. Instead of taking intervals on the infinite line of time, we will replace that by object placed on the infinite *x* axis. In his model, he used only eight of the thirteen relations defined by Allen. He also chose to use other symbols to represent them but that do not change the fact that it is the same relations. The table 2.2 shows the relations between two objects O_1 and O_2, the symbol to represent them and the equivalent from Allen's theory. So for example, the symbol $\prec$ meaning *left of* is the spatial equivalent of the symbol $<$ meaning before. Note that when dealing with space the meaning is relative to the axis we are using. Thus, that same relation would be *below of* on the *y* axis and *front of* on the *z* axis.

Table 2.2: Relations of Guesgen's model

Relation	Symbol, Inverse	Allen's equivalent	Illustration
O_1 left of O_2	$\prec, \succ$	$<$	
O_1 attached to O_2	$\preccurlyeq, \succcurlyeq$	m	
O_1 overlapping O_2	$\Leftarrow, \Rightarrow$	o	
O_1 inside O_2	$\sqsubset, \sqsupset$	d	

The formalization of these relations in spatial term is quite simple. Suppose we have O_1 and O_2 as the object and S_1 and S_2 as their corresponding space. Then, we can define the eight basic relations with the following four axioms:

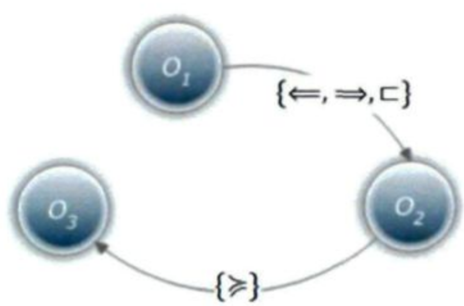
(1) $O_1 \prec O_2 \Leftrightarrow \forall x \in S_1, y \in S_2 : x < y$

(2) $O_1 \preccurlyeq O_2 \Leftrightarrow \forall x \in S_1, y \in S_2 : x \leq y \wedge \exists x \in S_1, y \in S_2 : x = y$

(3) $O_1 \Leftarrow O_2 \Leftrightarrow \exists x \in S_1 \, \forall y \in S_2 : x < y \wedge \exists y \in S_2, \forall x \in S_1 : x < y$

(4) $O_1 \sqsubset O_2 \Leftrightarrow \exists y \in S_2 \, \forall x \in S_1 : x < y \wedge \exists y \in S_2, \forall x \in S_1 : x > y$

The spatial relation between many objects can be represented as a network. A simple example with three objects in relation is shown on figure 2.5.



$$(O_1 \Leftarrow O_2) \vee (O_1 \Rightarrow O_2) \vee (O_1 \sqsubset O_2) \wedge (O_2 \preccurlyeq O_3)$$

Figure 2.5: Example of spatial network between three objects

In his model, Guesgen show with an example how we can reason using that kind of network by adding information on objects' relations or simply by inferring new information. Let imagine we now possess new information about the relation between O_1 and O_3. That relation is $(O_1 \Leftarrow O_3)$ meaning that O_1 overlaps O_3. The figure 2.6A shows the network with the new information. Knowing that new information we eliminate possible relations from the set of possibilities between O_1 and O_2. For O_1 to be inside O_2, the relation between O_2 and O_3 would require to be overlaps but it is *attach*. The relation O_1 overlaps O_2 is even worse so it needs to be eliminated too. The figure 2.5B show the network inferred from the new information.

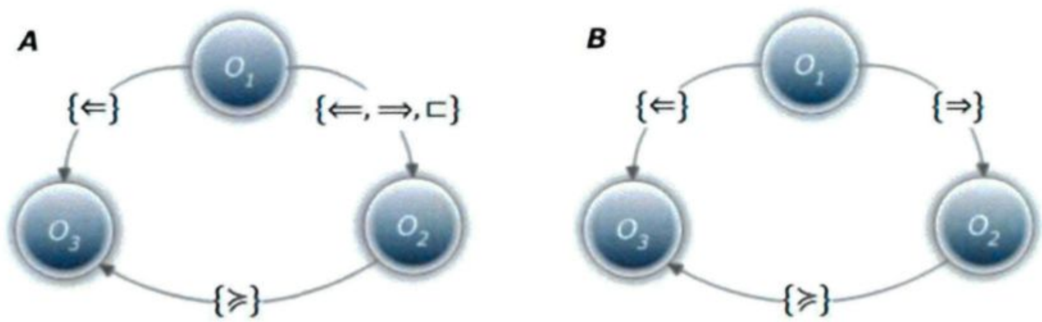


Figure 2.6: (A) New information (B) Inference from that information

2.3.3 Extension to multi-dimension reasoning

Guesgen's model [43] is very simple and straightforward, but it would not be interesting without the possibility to extend it to 2 or 3D. It is very easy to do; all that is required is to replace a spatial relation with a pair or a triple corresponding to each axis that situate the objects in space. Considers the 3D blocks on the figure 2.7. Using the traditional 3D Cartesian axis (x, y, z) we can define the relationship of objects in pair. On the *x-axis*

O_1 is obviously inside O_2. On the *y-axis*, O_2 is attached to O_1 and on the *z-axis* O_1 is again inside O_2. The result can be expressed as $O_1(\sqsubset, \succcurlyeq, \sqsubset)O_2$. We can do the same check up for O_2 and O_3 resulting into the relation $O_2(\prec, \sqsubset, \Longleftarrow)O_3$. The relation between O_1 and O_3 is however a little bit more complicated to define without precise coordinates of the blocks. For the *x-axis* it is easy to conclude that it is to the left, but for the *y* and *z* we cannot have a single conclusion from this image. On *y-axis*, it is possible that O_1 is inside or it can be overlapping. On the *z-axis* it is also probably inside or overlapping resulting in the following relation: $O_1(\prec, \{\Rightarrow, \sqsubset\}, \{\Leftarrow, \sqsubset\})O_3$.

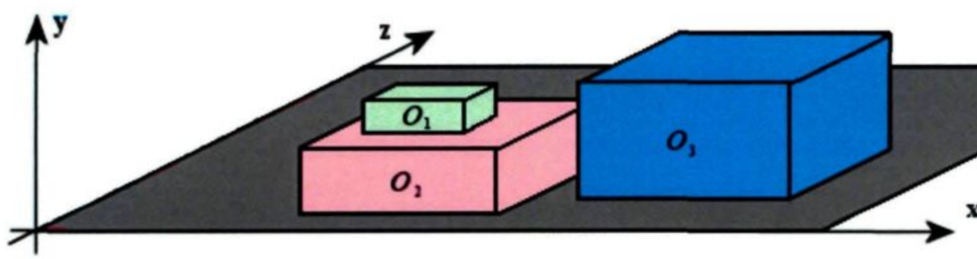


Figure 2.7: Reasoning in 3D [43]

2.4 EGENHOFER & FRANZOSA TOPOLOGICAL MODEL

The next framework we will present comes from Egenhofer & Franzosa [25]. Their first motivation was from a non-answered need for formal model to reason about space in the area of research on Geographic Information System (GIS). In the GIS area, users select data using query. That is the reason why many earlier works were on the spatial query languages [44, 45] explicitly incorporating spatial relations. These query languages follow the direction line of the earlier natural language work such as in [46]. The major drawback of these works is that they are often based on expressions, which are not exactly defined but only assumed as generally understood.

The framework of Egenhofer & Franzosa is a qualitative spatial reasoning model directly based on the mathematical theories of general topology that can also be called point-sets topology. They wanted to demonstrate two things during the elaboration of this framework. First, it has the goal of showing that the description of topological spatial relations in terms of topologically invariant properties of point-sets is rather simple. The consequence of this affirmation is that computational efforts should be little. Second, they wanted to demonstrate that there exists a framework within which any topological relation falls. The point-set approach is the most general form of expression of topological spatial relation. It generalizes models based on intervals [47] or simplicial complexes [48].

2.4.1 TOPOLOGY BASIS FOR THE MODEL

First of all, we need to take a look on what is general topology. It a subfield of topology that studies the properties of topological space and structures defined on them. Topology has itself emerged from the geometry and the set theory as the field that studies properties of objects preserved from continuous deformation. In this section, we will present the topological premises to understand Egenhofer's framework. Many proofs are purposely omitted, but an interested reader could find more details in any good academic book on general topology [36, 49].

2.4.1.1 *Topological space*

A topological space is a set $\mathcal{X}$ with a topology Σ on $\mathcal{X}$. So, to understand it properly, we must first define a topology. A topology Σ on $\mathcal{X}$ is a collection of subsets of $\mathcal{X}$ that satisfies the following three axioms:

I. The empty set $\emptyset$ and $\mathcal{X}$ are in Σ.

II. Any union of elements in Σ is an element of Σ.

III. Any intersections of a finite number of elements in Σ is an element of Σ.

The sets in a topology on $\mathcal{X}$ are called *open* sets and their complements in $\mathcal{X}$ are called *closed* sets. The collection of closed sets follows these three axioms:

I. It contains the empty set $\emptyset$ and $\mathcal{X}$.

II. It is closed under arbitrary intersections

III. It is closed under finite unions

The figure 2.8 shows two examples of topology on a three point set (**(i)** and **(ii)**). The example **(iii)** is not a topology because the union of the points {A} and {B} is missing ({A, B}). The example **(iv)** is not a topology because the intersection of {A, B} and {B, C} is missing ({B}).

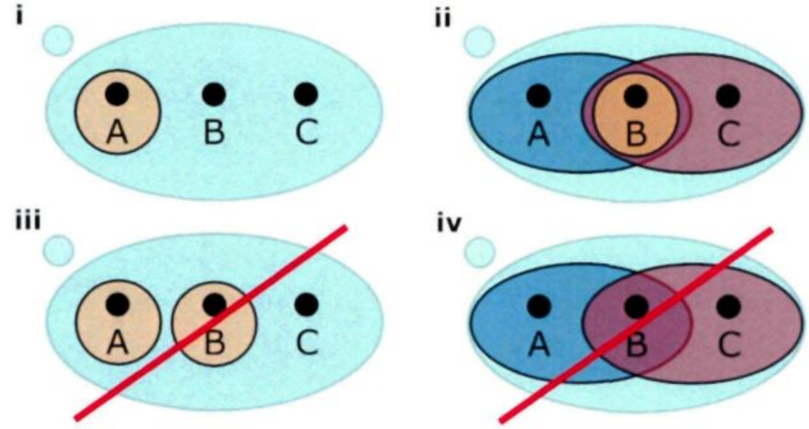


Figure 2.8: Example of topology (i-ii) and non-topology (iii-iv)

2.4.1.2 Subset of a topological space

Let take back $\mathcal{X}$ a set with a topology Σ following the axioms described in the previous section. If $\mathcal{S}$ is a subset of $\mathcal{X}$ then it inherits a topology from Σ that is called a *subspace topology*. That subspace topology is defined as $\Sigma_{\mathcal{S}} = \{\mathcal{S} \cap \mathcal{U} \mid \mathcal{U} \in \Sigma\}$. Equipped under this condition, $\mathcal{S}$ become a topological space in its own right that is called *subspace* of $\mathcal{X}$. Let's look at an example of subspace topology:

Figure 2.9: Subspace topology on $\mathbb{R}$

Suppose $\mathcal{X}$ is the line representing real number $\mathbb{R}$ with the usual topology (e.g. the set of all open sets on the line). An open set on that line is a kind of collection of open intervals such as it is shown in blue on the figure 2.9. Let $\mathcal{S}$ be the interval [a, b]. The inherited topology on [a, b] is the collection $\Sigma_{\mathcal{S}}$ of all intersections of [a, b] with the set of all open sets of $\mathbb{R}$. The open sets of $\Sigma_{\mathcal{S}}$ will consist of closed / open sets of type [a, b[and

]a, b] that are respectively left-closed and right-closed (or right-open and left-open). Finally, the topological space $\mathcal{S}$ with a topology $\Sigma_{\mathcal{S}}$ is a subspace of $\mathbb{R}$.

2.4.1.3 Definition of the framework primitives

Given $Y \subset \mathcal{X}$ we need to define some primitives that will be used in the framework for spatial reasoning.

Interior: The interior of Y is written Y°. It is defined as the union of all open sets that are contained in Y. So it means that Y° is the largest open set contained inside Y. An element $y \in Y^{\circ}$ only if there exists an open set U such that $y \in U \subset Y^{\circ}$. The interior of the empty set $\emptyset$ is the empty set. That is the interior of a set can be empty. The interior of $\mathcal{X}$ is $\mathcal{X}$ and if Y is open then $Y^{\circ} = Y$. If $U \subset Y$ then $U^{\circ} \subset Y^{\circ}$.

Closure: The closure of Y is defined to be the intersection of all closed sets that contains Y. It is written $\bar{Y}$. It means that the closure of a set is the smallest closed set that contains it. An element $y \in \bar{Y}$ only if the following intersection is respected: $U \cap Y \neq \emptyset$ for every open set U such as $y \in U$. The closure can be empty, but it only happen for the empty set $\overline{\emptyset} = \emptyset$. The closure of $\mathcal{X}$ is $\mathcal{X}$ and if a set Y is closed then $\bar{Y} = Y$. For a subset Z of Y, the closure of Z is also a subset of the closure of Y : $\bar{Z} \subset \bar{Y}$.

Boundary: The boundary of a subset Y of a topological space $\mathcal{X}$ is denoted ∂Y. It is the intersection of the closure of Y and the closure of the complement of Y. That is $\partial Y = \bar{Y} \cap \overline{X - Y}$. The boundary is a closed set and $y \in \partial Y$ only if for every open set U containing y $U \cap Y \neq \emptyset$ and $U \cap (X - Y) \neq \emptyset$. The boundary can be empty such as it is for the empty set $\emptyset$ and for $\mathcal{X}$.

The figure 2.10 illustrates these definitions. We can see, on A, that for the subset $\mathcal{S}$ the points $\{x, y\} \subset \mathcal{S}^{\circ}$ but that the element $\{z\} \not\subset \mathcal{S}^{\circ}$. However, $\{z\} \subset \partial\mathcal{S}$ and x, y are not in the boundary. All three elements are in the closure of : $\{x, y, z\} \subset \overline{\mathcal{S}}$. The figure 2.10B illustrates the set of real number $\mathbb{R}$ as a one dimensional infinite line. Both the point $\{x, y\} \subset \partial\mathbb{R}$ and the interior of $\mathbb{R}$ is empty: $\mathbb{R}^{\circ} = \emptyset$. Thus, the closure $\overline{\mathbb{R}} = \partial\mathbb{R}$.

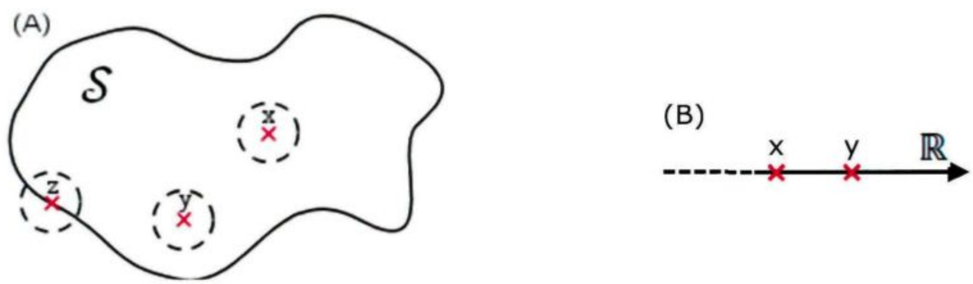


Figure 2.10: (A) Example of points in the topology $\mathcal{S}$ and on (B) $\mathbb{R}$ the set of real numbers

Before concluding, here are some important properties.

- $\mathcal{S}^{\circ} \cap \partial\mathcal{S} = \emptyset$
- If A, B separate $\mathcal{S}$ and $Y \subset \mathcal{S}$, then either $Y \subset A$ or $Y \subset B$ but not both.
- $\mathcal{S}^{\circ} \cup \partial\mathcal{S} = \overline{\mathcal{S}}$
- $\mathcal{S} \subset \mathcal{X}$. If $\mathcal{S}^{\circ} \neq \emptyset$ and $\overline{\mathcal{S}} \neq \mathcal{X}$ then $\partial\mathcal{S}$ separate $\mathcal{X}$

2.4.2 FRAMEWORK TO DESCRIBE RELATIONS

The framework to describe spatial relations between two sets A and B in a topological space $\mathcal{X}$ is a four-tuple (_, _, _, _) representing the intersections between interiors and boundaries of those same sets. These intersections are denoted by like this: $(\partial A \cap \partial B, A^{\circ} \cap B^{\circ}, \partial A \cap B^{\circ}, A^{\circ} \cap \partial B)$. The entries correspond to the values of topological invariants associated to the set-intersections. In other words, the framework proposes p^4 types of relation where p is the number of possible value of the associated property. The binary property of emptiness $(\emptyset, \neg\emptyset)$ has been chosen for their original framework because it is the simplest and most general invariant. As it is a binary property, p=2 and thus sixteen types of relation can be defined. However, when dealing with *spatial regions*, not all of them can be observed. A spatial region is an entity that follows these properties:

I. The interior of each region is **non-empty**.
II. A spatial region is closed and connected.
III. The boundary of a spatial region is **non-empty**.

Out of the 16 combinations, only nine exist between spatial regions. These valid combinations are represented in the table 2.3 with their associated semantic.

Table 2.3: The nine relations that exist between spatial regions

	$\partial A \cap \partial B$	$A^{\circ} \cap B^{\circ}$	$\partial A \cap B^{\circ}$	$A^{\circ} \cap \partial B$	
1	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	A and B are disjoint
2	$\neg\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	A and B touch
3	$\neg\emptyset$	$\neg\emptyset$	$\emptyset$	$\emptyset$	A equals B
4	$\emptyset$	$\neg\emptyset$	$\neg\emptyset$	$\emptyset$	A overlaps B
5	$\neg\emptyset$	$\neg\emptyset$	$\neg\emptyset$	$\emptyset$	A is covered by B
6	$\emptyset$	$\neg\emptyset$	$\emptyset$	$\neg\emptyset$	A is overlaped by B
7	$\neg\emptyset$	$\neg\emptyset$	$\emptyset$	$\neg\emptyset$	A covers B

8	$\neg\emptyset$	$\neg\emptyset$	$\neg\emptyset$	$\neg\emptyset$	A and B overlap with intersecting boundaries
9	$\emptyset$	$\neg\emptyset$	$\neg\emptyset$	$\neg\emptyset$	A and B overlap with disjoint boundaries

The ninth relation is somehow special from the others. To obtain such a relation, one of the spatial regions must be in the shape of a donut. The other one will cover the hole without ever touching the boundary of the donut with his own. The figure 2.11 represents that relation between A (donut shaped) and B.

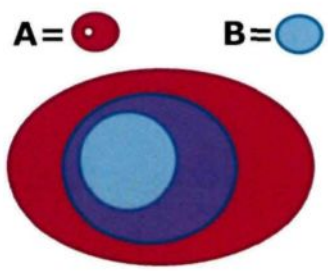


Figure 2.11: A and B overlap with disjoint boundaries

The other eight relations that are more common can be visualized on the figure 2.12. Before concluding on the work of Egenhofer & Franzosa, it is important to say that they demonstrated that the framework is general enough to be valid in $\mathbb{R}^n$. However there are relations that do not exist when n is equal to 1. An extension of this same framework has been developed to focus on spatial relations in 3D of spherical entity [50].

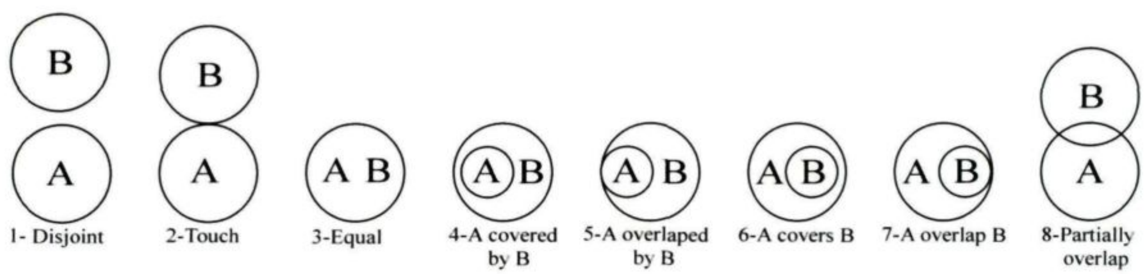


Figure 2.12: First eight relation types

2.5 Region Connection Calculus (RCC)

We will now present a qualitative spatial reasoning framework that has been developed through many years of research at the University of Leeds [31]. That formalism has been developed for similar purposes than the others presented in this section but differs from two points. First, the intended goal was to build a simple to understand framework that would be usable for many fields of application. Second, they wanted to deal with objects that possess uncertain boundaries because it was ill addressed until recently. The RCC is derived in more than one version. To our knowledge, there are at least the RCC5, RCC8, RCC15 and RCC23, where the numbers after RCC represent the quantity of possible types of relation between two spatial entities. We will only discuss the RCC8 because RCC15 and RCC23 are mostly an upgrade to deal with irregularly shaped spatial regions (that we do not need in our context, (see Chapter 4.3.2.4) such as donut or concave shape.

The name RCC reflects the first important difference from most models that reason about space. *Region* is the primitive they have chosen to use for their calculus instead of the point. We will discuss the advantages and disadvantages of that choice at the end of the chapter but for RCC, they made an exhaustive justification in the paper [51]. The RCC is closely based on Clarke's system that was published in many papers such as [52, 53]. That system was at first intended to be a spatio-temporal calculus. The basis of the system is a primitive relation $\mathsf{C}(x,y)$ read as "x connects with y". In the original point-based system, "connects" meant that at least one point is in x and in y. C is reflexive and symmetric and most of its power has been transcended in the RCC at the exception of the ability to distinguish between open and closed regions. That is a consequence of using regions instead of points as primitives. It is important to compare RCC with Allen's temporal system. Allen's logic is based on interval of time instead of the defined notion of **moment**. That is the equivalent in one dimension of using the *region* for spatial reasoning and thus closely related. As we demonstrated earlier in this chapter, it is possible to directly use Allen's temporal logic to deal with space, but we will discuss later some disadvantages over other methods.

2.5.1 INTRODUCTION TO THE CALCULUS

First of all, the primitive *region* should be exactly defined. In RCC, regions can be of arbitrary dimension. There are only two restrictions. First, all regions must be of the same dimension for the calculus. Second, a region must not be of mixed dimension. For example, a 2D plane with a spike surging on the z axis would not be a valid region. These regions are called *regular,* and as long as we deal with that type of region, RCC can be used in 3D as in 2D. The only other restriction to regions in RCC is that it cannot be null. Otherwise, a region can have holes, tunnels and be multi-piece without problems.

We briefly described the function $\mathsf{C}(x,y)$ from Clarke's system. But since we are dealing with regions instead of points in RCC, the definition must be adjusted. How to say that two regions are connected without using the point primitive? The informal interpretation of C is given by saying that two regions are connected if the distance between them is zero. That is, however, a weak definition of what is a connection. To my knowledge, it has not been demonstrated formally how work the connection with region primitive, but it is assumed that it is not important by many. The RCC is formalized with sorted first-order logic. So far, it deals with relationship between entities of type Region and NULL.

2.5.1.1 *Axioms of the function* **C**

We need to define the different axioms that are used to express the relationship of regions. First, we have said that C was reflexive and symmetric. Here are the formal descriptions of that affirmation:

(1) $$\forall x[\mathrm{C}(x,x)]$$

(2) $$\forall x \forall y[\mathrm{C}(x,y) \rightarrow \mathrm{C}(y,x)]$$

Using $\mathrm{C}(x,y)$, a basic set of relations can be defined. These relations and their semantic interpretation are given in the table 2.4 that almost directly come from the paper of Cohn & al. [31].

Table 2.4: Relations defined from C

	Relation	Definition	Interpretation
1	$\mathrm{DC}(x,y)$	$\neg\mathrm{C}(x,y)$	x is disconnected from y
2	$\mathrm{P}(x,y)$	$\forall z[\mathrm{C}(z,y) \rightarrow \mathrm{C}(z,y)]$	x is part of y
3	$\mathrm{PP}(x,y)$	$\mathrm{P}(x,y) \wedge \neg \mathrm{P}(y,x)$	x is proper part of y
4	$\mathrm{EQ}(x,y)$	$\mathrm{P}(x,y) \wedge \mathrm{P}(x,y)$	x is identical with y
5	$\mathrm{O}(x,y)$	$\exists z[\mathrm{P}(z,x) \wedge \mathrm{P}(z,y)]$	x overlaps y
6	$DR(x,y)$	$\neg\mathrm{O}(x,y)$	x is discrete from y
7	$\mathrm{PO}(x,y)$	$\mathrm{O}(x,y) \wedge \neg\mathrm{P}(x,y) \wedge \neg \mathrm{P}(y,x)$	x partially overlaps y
8	$\mathrm{EC}(x,y)$	$\mathrm{C}(x,y) \wedge \neg\mathrm{O}(x,y)$	x is externally connected to y

9	$\mathrm{TPP}(x,y)$	$\mathrm{PP}(x,y) \wedge \exists z[\mathrm{EC}(z,x) \wedge \mathrm{EC}(z,y)]$	*x* is a tangential proper part of *y*
10	$\mathrm{NTPP}(x,y)$	$\mathrm{PP}(x,y) \wedge \neg\exists z[\mathrm{EC}(z,x) \wedge \mathrm{EC}(z,y)]$	*x* is a nontangential proper part of *y*

Unless otherwise specified, all arguments are *Regions* in the sense we defined them. Some of the relations defined are non-symmetrical (P, PP, TPP, NTPP) and thus must support inverse. The notation used is Φi where the set contains {P, PP, TPP, NTPP} ∈ Φ. In RCC5, an earlier version, only {DC, PO, EQ, PP, PPi} existed (and the definition required for them). The figure 2.13 shows these five types of relation between two regions. It is provable that the relations {DC, EC, PO, EQ, TPP, NTPP, TPPi, NTPPi} are jointly exhaustive and pairwise disjoint. That is, for any relations between two regions only one hold and there is always exactly one of those relations that will hold. That set is the RCC8 and not surprisingly they are represented graphically as the eight relations from Egenhofer's system (figure 2.12). In fact, they are the same relation types. Only the definitions and the primitives change.

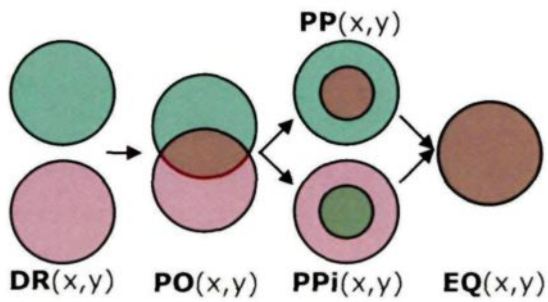


Figure 2.13: The five types of relation in RCC5

There is a thing that we omitted to say about these eight types of relation during our explanation of Egenhofer's system. We can place these relations into a neighborhood graph. This is true for RCC8 too. This is a feature that allows extending easily those models. It can be used to calculate similitude when comparing two relationships. We used that property in the realization of our model. The figure 2.14 shows the neighborhood of RCC8.

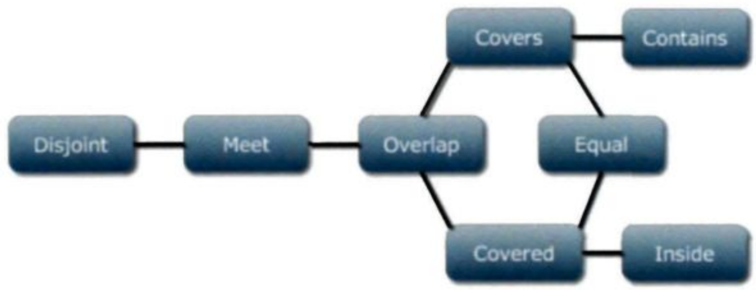


Figure 2.14: Neighborhood of the eight relations in RCC8

2.5.2 FUNCTIONS AND THEOREMS OF RCC

Following Clarke's original system, the Leeds University team has defined a series of functions on regions. The table 2.5 shows those functions with a brief explanation of their purpose.

Table 2.5: Basics function for region calculus

Function	Explanation
sum(x, y)	The sum of the region x and y
compl(x)	The complement of the region x
prod(x, y)	The product (intersection) of x and y
diff(x, y)	The difference between x and y (part of x that does not overlap y)

It has been already said that a region can be formed of multi-pieces. From the introduction of the *sum* function, it is now important to define a predicate that will test if it is the case:

(3) $$CON(x) \equiv_{def} \forall yz[\boldsymbol{sum}(y,z) = x \rightarrow C(y,z)]$$

They also define an axiom from Clarke's system. In fact, Clarke stipulates that every region has a nontangential part, and that it is essential to P(*x*,*y*) definition. However, that is true in his theory because it includes topological interpretation. Even so, RCC includes the following axiom:

(4) $$\forall x \exists y[NTPP(y,x)]$$

The consequence of having that axiom is that since every region possesses a nontangential part, there is an infinite number of regions in every model. However, when implementing RCC in a system, it is not required to have infinite data structure. That infinity can be abstracted from concrete systems. The axiom (5) defines the notion of two identical regions. It means that if and only if all regions connected to a region x are also connected to a region y and vice-versa, x and y are identical.

(5) $$\forall xy[x = y \leftrightarrow \forall z[C(z,x) \leftrightarrow C(z,y)]]$$

The theorem 6 is about the identity. It means that any region z which overlaps a closed region x will also overlap its open interior and vice-versa. That means that closed and open regions are identical in RCC.

(6) $$\forall xy[x = y \leftrightarrow \forall z[O(z,x) \leftrightarrow O(z,y)]]$$

We will not describe all theorems of RCC since it would be too long, but a last one is very important. The theorem 7 signifies that regions are connected with their complement. It is different from topological approaches and from Clarke's original theory.

(7) $$\forall x[EC(x, compl(x))]$$

The next section will discuss the implications of the fundamental difference between RCC and the other presented approaches.

2.6 Discussion on spatial reasoning

We presented three important approaches to SR in this chapter. The reader may have noticed that the approaches conclude to similar relation types between objects. It is true, and that is because qualitative spatial reasoning is a well-established field where a consensus has taken shape. Although, the three approaches are interesting, we will briefly compare them together. First, Allen's based theories are very simple to understand and are based on a theory that no longer needs to prove itself. However, they are often more complicated to apply on real situations than the models created for SR purpose [31]. Moreover, this as the fundamental limitation that it only works correctly for rectangular objects aligned to axes. Consider the figure 2.15. With Allen's based QSR, we will detect the object B as inside the object A for both axes when looking individually to each axe. But it is not so in 2D.

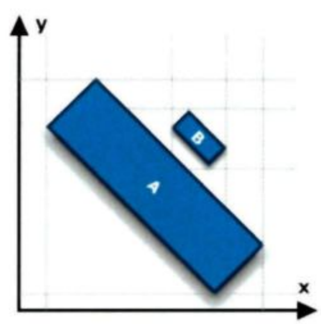


Figure 2.15: Illustration of the problem of reasoning in 2D with Allen's model [31]

Egenhofer's model and Cohn's model are both very similar. Egenhofer's model is more general approach since it is based on topology and also more powerful (more expressive). However, with RCC, it is easier to reason about irregular shapes such as concave shapes or shapes with holes. That could be very important for some problems, but it is not required for this thesis. Both models can be easily used to reason in 3D space, and overall computational complexity is very similar.

2.6.1 Quantitative or qualitative spatial reasoning?

We did not present purely quantitative approaches to spatial reasoning in this chapter. That is because we believe that QSR is a lot more adapted for our research problem. However, it is important to justify our choice by discussing the implications of it. First of all, we must argue that purely formal quantitative models are intuitively built in mathematics and for mathematics. Another important point is that nothing prevents us to use QSR models with quantitative information. But there is a more interesting point in using QSR that is discussed in Hernandez work [54]. In fact, QSR is perfectly fitted to abstract complex reality into simpler models. It also involves a lesser computational complexity than his quantitative opposite. Moreover, forcing a variable to take a numeric value can waste qualitative information such as equality between two variables.

2.6.2 Which primitive should we use?

One of the fundamental differences between Egenhofer's model and RCC is the primitive they use in their model. While Egenhofer uses the mathematical point, RCC uses *region* as primitive. First, we must say that *region* is a more natural way of thinking than point. It is rare to refer to point. Any physical object is a *region* and even when talking about points we generally refer to a very small *region* rather than the mathematical point. Points are better to express an infinite granularity. But it is not required in most case. That is why we will use mostly *region* in our model like RCC does. In fact, it will not be our primitive but physical object will be described only with them. However, we will use points for some calculation (to position objects in space, etc.).

2.7 CHAPTER CONCLUSION

The ending chapter was a small introduction to spatial reasoning and more precisely to QSR framework that we can use. In this thesis, we did choose to extend Egenhofer's model because it is more general and mathematically proven as valid. However, the presented models cannot directly be used for activity recognition, because they do not provide calculation methods to evaluate activities. Also, they are not made to deal with the various kinds of sensors and the imprecise information they provide. That is why we need to adapt it to our situation. Keep in mind that an entire book could have been dedicated to SR, and that we did only talk about the part that interested us in this thesis.

CHAPTER 3

RELATED WORKS ON ACTIVITY RECOGNITION

3.1 INTRODUCTION

Activity recognition is an instance of an old and well-known problem of computer science named the plan recognition paradigm. It has been a very active topic during the past few decades [8]. Still, it is only since recently with the arrival of ubiquitous computing and the advances of smart environment that it has become a master piece of ambient intelligence [14]. In this specific and applicative context, spatial information embedded in activities (position, orientation, distance, etc.) is crucial to optimize library circumscription, minimize plan hypotheses and identify certain types of errors. From that point of view, we can divide activity recognition approaches into three categories according to the kind of constraints they try to exploit in order to circumscribe the plans' library and narrow down hypothesis. This chapter proposes to review the principles of activity recognition and to present a state of the art of the different families of algorithms classified by their constraint types (logical, temporal and spatial). The first part of the chapter will portray the activity recognition problem and more specifically its application to the context of smart home assistance. The second part will describe the different approaches of activity recognition and will focus on spatial constraints based model. The chapter will conclude with a comparison between the presented models that will help positioning our works in the field.

3.1.1 ACTIVITY OF DAILY LIVING DEFINITION

Since the beginning of this thesis, we often referred to the concept of *activity of daily living* (ADL) but one could legitimately ask what meaning this concept is referring to. The notion of ADL has been first described by the Dr. Katz [55] as the set of activities that an individual performs in his routine to take care of himself. That includes activities such as preparing meals, getting dressed, toileting himself, etc. Healthcare professionals often evaluate the level of autonomy (functional status) of an impaired person with the capacity or incapacity to perform a certain activity of daily living. This metric is useful in assessing the degree of cognitive degeneration of a patient and to successfully discern the type of support he will need [56]. That is why many cognitive tests are based on ADLs performance such as the *Kitchen Task Assessment* [57] and the *Naturalistic Action Test* [58]. To summarize, the activities of daily living is a set of common activities that a normal person is supposed to be able to realize to be qualified as autonomous. Today, a consensus from researchers distinguishes three different types of ADLs:

Basic ADLs (BADLs): The basic activities of daily living (BADLs) are the set of activities that are fundamental and mandatory to answer primary needs of a person. Moving around without assistive device (ambulation), going to the bathroom, self-feeding, functional transfers (getting onto or off the bed), etc. These activities are composed of only a few steps and do not require real planning.

Instrumental ADLs (IADLs): This kind of activity needs basic planning to be performed and implies objects manipulations. These activities are needed to live alone and to live in society. For a person, being able to realize all instrumental ADLs means being relatively autonomous. That category includes activities such as: preparing a meal, managing money, shopping, using a phone to call someone, etc. IADLs are more complex, are composed of a higher number of steps and require better planning than basic ADLs.

In the scientific literature on assisting technology inside smart homes [59, 60], researchers mostly use ADLs without distinguishing the specific type. However, most of the time assisting systems in smart home focuses on recognizing and assisting instrumental ADLs. The main reason is that a person that cannot accomplish successfully basic ADLs will need more comprehensive care that smart home assistance is inappropriate to provide.

3.1.2 Defining activity recognition

Human intelligence is amazing in many facets. A good example comes from the fact that we use perceived information from the observation of a pair to deduce his action plan and his intended goal. That formidable ability allows us to anticipate the needs of others and therefore, promotes collaboration and assistance. From that fact, artificial intelligence has worked long on this problem that was firstly renowned as the *plan recognition* problem

[8]. The first definition that we can find in the literature comes from Schmidt [61]. In his work, he defines plan recognition as "…to take as input a sequence of actions performed by an actor and to infer the goal pursued by the actor and also organize the action sequence in terms of a plan structure". In that definition, we can deduce that to perform the *plan recognition* we suppose the existence of a plan structure (sequence of action organized in time and space) planned by the observed entity (in our case an Alzheimer resident). That structure constitutes the result that the observer tries to recognize (in our case, the smart home's sensors are the senses of the observer agent, and the algorithm is its brain).

That vision of the *plan recognition* problem is inherited from the first expert systems that were created to solve planning problem [62]. The problem of planning an activity is also a well-known challenge of the AI scientific community[63]. We can considerate it as the opposite of the activity recognition problem. The difficulty resides in the identification of an actions sequence (a plan of activity) by an agent that will allow it to attain a certain objective at the end of its execution [64]. By opposition, *activity recognition* implies an observed agent that does not know the initial goal of the other agent (the observed entity) and that intends to deduce the objective by inferring from observed actions the possible structure of the ongoing plan.

3.1.3 ACTIVITY RECOGNITION INSIDE SMART ENVIRONMENT

Since the original definitions from AI problem, the definition of activity recognition has evolved, getting precised by many notable authors such as [12, 65, 66]. Each has tried to adapt it to the very specific context of activity recognition inside a smart home. The trend has been to refine the notion of ambient environment to formally link it with the challenge of the activity recognition problem. For example, Goldman [67] describes it as the process of inferring an agent's plan from the observation of his action. The main distinction from previous definitions is the differentiation between the action of the observed entity, and the observation perceived by the observer. That distinction reflects the fact that actions are not directly observable in smart home context. Patterson [68] has recently proposed to upgrade the definition by specifying how the observations are made: "...observation made from data from low-level sensors". This new definition adheres to the paradigm of pervasive computing [69] and is much closer to the reality of the problem. It encourages the creation of enhanced environment where common objects will embed multi-modal sensors to remain less intrusive as possible. This thesis applicative context corresponds to this definition were multi-agents will communicate together observations made from various sensors in order to take decisions to help a resident with cognitive impairment. It also distances the problem of activity recognition from the legacy algorithms that considered we had access to the basic action executed by the observed entity. It is, in fact, not realistic in our context. The definition of Patterson, in contrast, assumes that only the indices triggered by actions are observable (change in the position of objects, change in the state of a sensor, etc.).

3.2 ACTIVITY RECOGNITION: CONSTRAINTS BASED CLASSIFICATION

During the development of activity recognition algorithms, numerous approaches were developed. Initially, researchers have classified the models into three main categories: logical, probabilistic and learning method based. The first family of algorithms derives mainly from Kautz [28] and uses pure logical constraints to recognize activity. The second family of algorithms use traditional probabilistic methods such as Bayesian network or Hidden Markov Machine (HMM) and the last one uses mainly the various learning methods [70] that exist in computer science. However, we can see that the terms probabilistic and learning methods are rather qualifiers and therefore, all families of activity recognition algorithms can possess algorithms with these attributes. This was the first indication of the need for a new method of classification. Secondly, the process leading to recognition of activities is "...to use of useful information extracted from observations to circumscribe a plans' library". This can be done by comparing observed properties with defined properties in the plans' library. These *discriminating properties* can be used to establish constraints of various kinds in the definition of activities in the library.

In the context of this master's thesis, we decided to classify models of recognition from the nature of the constraints used to circumscribe the library of plans and for detection of anomalies in their execution for the previously stated reasons. We regrouped algorithms into three families of constraint types: logical, temporal and spatial. The reason is that legacy models could all fall to the logical categories, and that we needed to distinguish earlier work on temporal and spatial recognition algorithms. Of course, mostly all algorithms incorporate at least a little bit of temporal and spatial aspects, but the distinction is made from the method for circumscription and errors detection. Finally, all algorithms, whether they are based on logical, spatial or temporal constraints, can be probabilistic or incorporate machine learning techniques. For this reason, we will discuss advantages and disadvantages of integrating such techniques at the end of this chapter.

3.3 LOGICAL BASED APPROACHES

The most important model that exists as the ancestor of many logical based approaches is the formal theory of Kautz [71]. We classify it into logical work even though it can be extended to integrate limited temporal and spatial constraints. It is because the model was not built in that purpose and do not require it to work properly. Moreover, constraints are very simple (temporal ordering of step) as we will explain later. Py [72], Nerzic [73] and Wobcke [74] have created enhanced versions each correcting one of the drawbacks of Kautz's theory. However, as we will discuss at the end of the section, none of them answer all the needs for cognitive assistance in smart home.

3.3.1 KAUTZ'S FORMAL THEORY FOR PLAN RECOGNITION

Kautz's theory for plan recognition is based on the exploitation of first-order logic to formalize the process of inference (deduction) of the ongoing plan. The theory presumes that there exists a plans' library made of plan schemas (activities' description) accessible to the observer agent. In this model, plans and actions are indifferently considered as events.

An event can be specialized to form the *abstraction* (ABS) of one or more events. An event can also be decomposed into many steps with the *decomposition* axiom (DEC):

(ABS) $\forall x. E_1(x) \supset E_2(x)$

(DEC) $\forall x. E_0 \supset E_1(f_1(x)) \wedge E_2(f_2(x)) \wedge \ldots \wedge E_n(f_n(x)) \wedge \kappa$

The symbol κ describes a conjunction of constraints on E_0. In fact, Kautz used sequential constraints derived from Allen's temporal theory [39] to order the steps of an event. However, the constraints were not used for other means than that; no temporal error detection was implemented, and the goal was to organize steps of a plan more than to exploit temporal aspect.

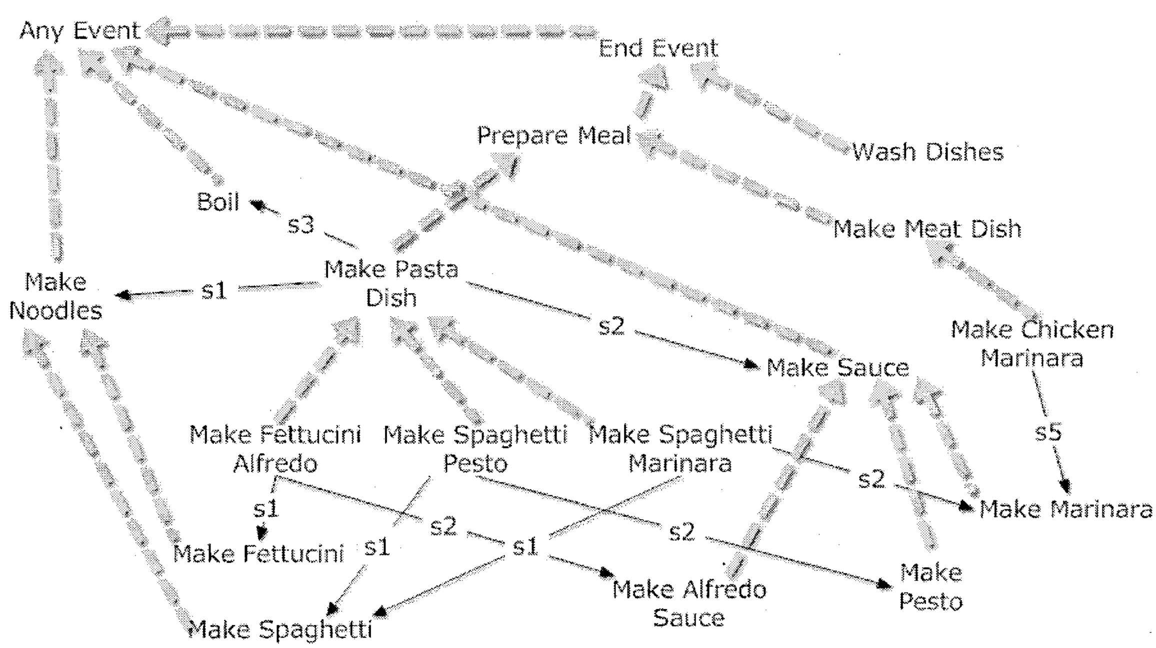


Figure 3.1: Cooking plans' library from [28]

The figure 3.1 is an example of library that represents the shared knowledge of a cook and an observer. The grey arrows denote the abstraction relations while the thin black arrows denote the decomposition of an event. The *End Event* is the root of self-motivated events. Such types of events represent intentions of the observed entity. The *Any Event* is the abstraction of all events of the hierarchy. In Kautz's theory, the hierarchy is always presumed correct. That means the observer agent will assume that it does not include any error in the relations of abstraction & decomposition. The second assumption is that the hierarchy is complete, meaning that it includes all the possible events that can be observed. Here are some examples of first-order axioms that are used to encode the library on figure 3.1:

(1) $\forall x\,.\,PrepareMeal(x) \supset EndEvent(x)$

(2) $\forall x\,.\,WashDishes(x) \supset EndEvent(x)$

(3) $\forall x\,.\,MakePastaDish(x) \supset PrepareMeal(x)$

(4) $\forall x\,.\,MakePastaDish(x)$

$$\supset MakeNoodle(s1(x)) \wedge MakeSauce(s2(x)) \wedge Boil(s3(x)) \wedge \kappa$$

The symbols *s1-3* map a plan to its steps. Note that they do not reflect any ordering and that their name is a subjective label. Each event can possess an abstraction. For instance, *MakePastaDish* is the abstraction of these three events: *MakeFettuciniAlfredo*, *MakeSpaghettiPesto*, and *MakeSpaghettiMarinara*. In our example, we did omit purposely to put constraints κ on the axioms. There are various types of constraints that could be defined. Kautz gave four types in his original work [28] that can be seen with an example on *MakePastaDish* in table 3.1.

Table 3.1: Examples of constraints on *MakePastaDish*

Constraint type	$\forall x . \boldsymbol{MakePastaDish}(x) \supset \boldsymbol{MakeNoodle}(s1(x)) \wedge \boldsymbol{MakeSauce}(s2(x)) \wedge \boldsymbol{Boil}(s3(x)) \wedge$
Equality	$agent(s1(x)) = agent(x) \wedge$ $result(s1(x)) = input(s3(x)) \wedge$
Temporal	$During\left(time(s1(x)), time(x)\right) \wedge$ $BeforeMeets\left(time(s1(x)), time(s3(x))\right) \wedge$
Preconditions	$InKitchen(agent(x), time(x)) \wedge$
Effects	$PastaDish(result(x))$

The constraints of equality (as seen on table 3.1) can be used to assert that the agent doing each step is the same that is doing the overall activity. It also can be used to make sure the noodles the agent made (*result* of the step *s1*; *MakeNoodle*) is the thing getting boiled (or the *input* of step *s3*; *Boil*). The temporal constraints, from Allen's theory, are used to state relations between a step and the plan or to order the execution of steps of a plan in time. So, it can specify that *MakeNoodles* must take place *during* the time of *MakePastaDish* and that *Boil* must follow the step *MakeNoodles*. As the name explicitly says so, *Preconditions* are used to assert that a condition is respected for an activity. For example, to *MakePastaDish*, the resident must be in the kitchen. Finally, an *effect* of the event is a consequence, such as there is a *PastaDish* that is the *result* of the event.

3.3.1.1 Kautz's assumptions

Considering a certain library valid as the definition provided in the previous section, Kautz describes a recognition process based on four inferences rules. These rules allow extracting a minimal interpretation model from that library (a subset) from the introduction of observations as logic assertions. The result of this inference process is a disjunction of hypotheses (a disjoint set of possible activities) that corresponds to the activities that are included in the minimal covering tree. This process is directly inspired from McCarty's circumscription theory [75]. It uses the fourth following assumptions:

(EXA) $$\forall x. E_0(x) \supset (E_1(x) \vee E_2(x) \vee \dots \vee E_n(x))$$

(DJA) $$\forall x. \neg E_1(x) \vee \neg E_2(x)$$

(CUA) $$\forall x. E(x) \supset End(x) \vee \left(\exists y. E_{1,0}(y) \wedge f_{1i}(y) = x\right) \vee \dots \vee \left(\exists y. E_{m,0}(y) \wedge f_{mi}(y) = x\right)$$

(MCA_n) $$\forall x_1 \dots \forall x_n. (End(x_1) \dots End(x_n) \supset \bigvee_{i,j} (x_i = x_j)), \quad i,j \in [1,n] \wedge i \neq j$$

The exhaustiveness assumption (EXA) expresses that if an observation x is associated to an event of type E_0, it is also the instance of at least one of its specialization. For example, if we observed *MakeSauce(x)*, we know that either *MakeMarinara(x)*, *MakePesto(x)* or *MakeAlfredo(x)* is true. Note that it would not work without the presumption that our library is complete. The disjointness assumption (DJA) complements the precedent one. It specifies that an observation of type E_n can be the instance of more than one type of event from the same abstraction event. For example, if *MakePesto(x)* is true, then neither *MakeAlfredo(x)* nor *MakeMarinara(x)* can be true since the three of them are specialization of *MakeSauce(x)*. The third assumption, the component/use (CUA), define that an event observed x must emanates from an *End* event (a self-motivated plan) or be part of a plan that itself emanates from *End*. Let's say that we observed *Boil*. This assumption expresses that *Boil* not being an *End* event is necessarily a step part of a greater objective (*MakePastaDish, etc.*). Finally, the minimum cardinality assumption (MCA) specifies that if two actions can be part of a certain plan, they will be assumed as so. Thus, if we had observed *MakeSpaghetti* and *MakeMarinara*, we will assume that both are part of the plan *MakeSpaghettiMarinara* instead of associating *MakeMarinara* to the plan *MakeChickenMarinara*.

3.3.1.2 Recognition process

From the formal theory we presented, Kautz's recognition process works as follow: (1) after each observation, applies the (CUA); (2) uses the (ABS) recursively to obtain an *End* type action; (3) tries to reduce the plans with (DJA)(EXA); (4) fuses multiple observations to fewer plans as possible with (MCA). In order to illustrate the algorithm, let's look at an example from the *Cooking World*:

[1]	$MakeNoodles(o_1)$	Observation
[2]	$MakePastaDish(p_1) \wedge step_1(p_1) = o_1$	(CUA)
[3]	$PrepareMeal(p_1)$	(ABS)
[4]	$End(p_1)$	(ABS)

The first thing observed is the action *MakeNoodles*. From the (CUA) assumption we infer that it is the first step of the plan *MakePastaDish* (p_1). Next, we use recursively the (ABS) axiom that allow us to find the root of self-motivated event (*End* node). Now, let us suppose that due to a lack of a certain ingredient, we know that we cannot *MakeAlfredoSauce*:

[5]	$\forall x. \neg MakeAlfredoSauce(x)$	Knowledge
[6]	$MakeSpaghettiMarinara(p_1) \vee MakeSpaghettiPesto(p_1) \vee MakeFettuciniAlfredo(p_1)$	(EXA)
[7]	$MakeFettuciniAlfredo(p_1) \supset MakeAlfredoSauce(step_2(p_2))$	(DEC)
[8]	$\neg MakeFettuciniAlfredo(p_1)$	Modus Tollens
[9]	$MakeSpaghettiMarinara(p_1) \vee MakeSpaghettiPesto(p_1)$	Elimination
[10]	$MakeSpaghettiMarinara(p_1) \supset MakeSpaghetti(step_1(p_1))$	(DEC)
[11]	$MakeSpaghettiPesto(p_1) \supset MakeSpaghetti(step_1(p_1))$	(DEC)
[12]	$MakeSpaghetti(step_1(p_1))$	From 9,10,11

From that knowledge, we were able to reason with the various assumption and we concluded that the type of pasta the person was going to cook was spaghetti. Even though the plan is not precisely recognized, this allows predicting some information in order to assist a resident.

3.3.2 Others logical constraints based approaches

The approach of Kautz suffers many limitations for the application to cognitive assistance. First of all, it does not allow defining erroneous plans of activity. It is required to be able to recognize erroneous plans for cognitive assistance and that limitation alone make the use of this model impossible. Py [72] has extended the framework of Kautz in order to include a new type to correct that problem. To do so, it defined a new self-motivated event named *error*. That new type allows introducing mistaken plans' descriptions in the library. However, as Kautz himself pointed out, it is very limited because it requires to define every possible erroneous instance of a plan; a potentially enormous number of variations. In the same line of idea, Nerzic [73] has improved the work of Kautz by exploiting the properties of second order logic. It allowed him to adapt the inferential process in order to identify possible erroneous plans. His proposition can be resumed by this explanation. When two observations fail the fusion process, and they can be part of the same plan, the second observation is considered as the erroneous continuation of the first plan. However, it is limited by the fact that the first action must be correct (to the intended plan), and that it does not allow interleaved activities. But one of the most important limitations of almost every logical approach is that they consider possible to detect the basic actions of a plan. It is, in fact, a very hard nontrivial problem. Finally, there is also the problem of the equiprobability of plans that is unrealistic. For instance, if I dislike tea but love coffee and basic actions *BoilWater* and *GetCup* are observed, it is a lot more probable that it is part of the plan *MakeCoffee* than the plan *MakeTea*. Therefore, at the end of the circumscription process, there is no mean to discriminate the remaining hypotheses. On the other hand, as we will discuss at the end of this chapter, machine learning and probabilistic techniques can alleviate to this problem.

3.4 THE TEMPORAL APPROACHES

In this section, I will present the work of Jakkula & Cook [29]. This model is probably the most well-known activity recognition approach based on temporal constraints in the field. However, there are many other works that possess advantages and disadvantages over this one. We will discuss them at the end of this section, and we will also explain why it is not sufficient alone as recognition approach for cognitive assistance.

3.4.1 JAKKULA & COOK

The model of Jakkula & Cook is built in a multi-agents [76] fashioned architecture where the agents perceive directly the state of the environment from sensor's output raw data. The temporal part is constructed from Allen's intervals based temporal relations presented in Chapter 2 [27]. The recognition algorithm uses data mining [70] and thus is based on the machine learning paradigm. They process raw data to discover frequent sequential patterns. In that case, it enables the discovery of temporal links existing between frequent events. For example, if recorded data tends to demonstrate that every time *Take Tea* happens the kettle is activated soon after, the recognition system will infer a temporal rule from Allen's thirteen relations (*Boil Water* after *Take Tea*).

Since it is a data mining algorithm, to work properly, training data must be gathered prior to process. Supposing that a lot of training data are available, Jakkula & Cook's model works as follows. First, the temporal intervals are found using the timestamp of events and the on/off state. The algorithm that associates these intervals to one of Allens' relations is illustrated below (algorithm 3.1):

Algorithm 3.1: Temporal Interval Analyzer [77]

```
E={set of events}
Repeat
  While (Ei && Ei+1)
    Find pair ON/OFF events in data to determine temporal range
    Read next event and find temporal range
    Associate Allen's relation between events
    Increment Event pointer
Loop Until end of input
```

The algorithm loops until all the pairs of events are compared. Between each pair, it establishes the Allen's relationship from the beginning and end markers of both events.

The second step in their model is to identify frequent activities or events that occur during a day to establish a reduced set of activities. This step is mandatory because there is too much data from smart home sensors, and many potential anomalies are just noise that should be ignored. They accomplish this task using the Apriori algorithm [78] that is shown below (algorithm 3.2):

Algorithm 3.2: Apriori algorithm [78]

```
Ck: Candidate itemset of size k
Lk: Frequent itemset of size k

L1={frequent items};
For (k=1; Lk!=∅; k++)
  Ck+1=candidates generated from Lk;
  For each day t in datasets do
    Increment the count of all candidates in Ck+1 that are in t;
  Lk+1=candidates in Ck+1 with min_support
Return ∪k Lk
```

3.4.2 ANOMALY DETECTION & PREDICTION ALGORITHM ENHANCEMENTS

In their work, Jakkula & Cook not only demonstrated that temporal relationships provide insights on patterns of resident behaviors, but also that it enhances the construction of other smart home assistance algorithms. To do so, they calculate the probability that a certain hypothetic event occurs or not given the observed occurrence of other events temporally related. It is done from the frequency of the nine relationships out of thirteen they determined that could affect anomaly detection: *before*, *contains*, *overlaps*, *meets*, *starts*, *started-by*, *finishes*, *finished-by* and *equals*. The formula to calculate the evidence of the occurrence of an event X is given by the observation of other events (such as Y) that are temporally related (from previous learning phase). The equation below allows such calculus:

(1) $$P(X|Y) = |After(Y,X)| + |During(Y,X)| + |OverlappedBy(Y,X)| + |MetBy(Y,X)| + |Starts(Y,X)| + |StartedBy(Y,X)| + |Equals(Y,X)| \;/\; |Y|$$

That equation gives the likelihood of X considering Y. To combine evidence of X from multiple events that are in temporal relationship with X, we have to improve the equation. Consider the events Z, Y that had been observed in this order, the prediction of X is given by the formula $Prediction_x = P(X)$ that is calculated as follows:

(2) $$P(X|Z \cup Y) = \frac{P\big(X \cap (Z \cup Y)\big)}{P(Z \cup Y)} = P(X \cap Z) \cup \frac{P(X \cap Y)}{P(Z)} + P(Y) - P(Z \cap Y)$$
$$= P(X|Z).P(Z) + P(X|Y).\frac{P(Y)}{P(Z)} + P(Y) - P(Z \cap Y)$$

From the formula, we can detect anomalies and make predictions. If an event X as a probability approaching 1, then it is considered as most likely to occur. On the other hand, if its probability is close to 0, it will be considered as an unusual event and ignore from further prediction. The final step is to use an enhanced version of the *Active LeZi* (ALZ) [79] algorithm for the prediction by adding these discovered temporal rules as input data. This predictor is sequential and employs incremental parsing and uses Markov models. It should be noted that *ALZ* improved could be used for anomaly detection. This could be done by using the prediction as input in an anomalies detection algorithm and by comparing prediction sequence with observations. Thus, if the new observation does not correspond to the expected event, an assisting sequence could be triggered. The add-on to the *Active LeZi* is shown below (algorithm 3.3):

Algorithm 3.3: Temporal Rules Enhanced prediction

```
Input: Output of ALZ a, Best rules r, Temporal dataset
While a!=null Loop
  Repeat
    Set r1 to the first event in the first best rule
    If (r1==a) Then
      If (Relation!="After") Then
        Calculate evidence (EQ 1)
        If evidence>(Mean+2 Std. Dev.) Then
          Make event in the best rule as next predictor output
        Else
        *Get next predicted event
         Look for temporal relations based on frequency
         Calculate evidence, store in a buffer

        If again relation is after Then goto *
        Until no more "After"
        Calculate evidence
        If evidence>(Mean+2 Std. Dev.) Then predict
        Else
        Calculate evidence
        If evidence>(Mean+2 Std. Dev.) Then
          predict this event based on relation
      End If
  Until end of rules
End While
```

Following the creation of this algorithm, they have conducted experiments that can be seen in table 3.4 below. It shows the accuracy of the observed prediction performance on real data sets and synthetic. There is a performance improvement of the prediction of activities of the resident of the intelligent environment. The main reason for a significant error rate is the small amount of data used. The search for knowledge-based temporal rules is a new area of research in intelligent habitats. Note that the use of temporal relationships provided a unique new approach for prediction.

Table 3.2: Comparison of ALZ prediction with and without temporal rules

Datasets	Percentage accuracy	Percentage error
Real (without rules)	55	45
Real (with rules)	56	44
Synthetic (without rules)	64	36
Synthetic (with rules)	69	31

3.4.3 Review with others temporal approaches

The approach presented is innovative and exploit well temporal relations between events. Nevertheless, it has some limitations. First of all, their framework is absolutely dependent of the machine learning techniques. Such techniques can help to personalize algorithms but often require a lot of training in order to be performing well. Besides, we can see that point in the results of their publication (see table 3.2). Moreover, the training is not universal for all residents and thus for each implementation it must be repeated. Song & Cohen [80] has adapted the algorithm of Kautz in order to better address temporal aspects in recognition. They added a rule to fix the beginning and the end of an event with the beginning and the end of its sub events. It improved the usefulness of the temporal constraint of Kautz, but it is not as rich as the work of Jakkula & Cook. Another important model comes from Weida [81]. Similarly to the work presented in this section, his model was centered on the temporal aspect to solve the problem of activity recognition. It is based on description logic, but it modified it by introducing the notion of constraint networks, where network nodes represent concepts in the sense of description logic and where the edges represent the constraints of temporal nature. This model has the advantage over the one of Jakkula & Cook to not depend on learning techniques. It also provides good recognition rate and thus is a valid approach. However, none of these approaches are sufficient for activity recognition because they cannot identify the spatial problem we stated in chapter 2 and their recognition rate could be further improved.

3.5 Spatial constraints based model

Spatial recognition is only beginning to get the researchers' attention even though it has already been recognized as a fundamental aspect of activity recognition algorithms. In this section, we present two approaches that integrate spatial notions. The first one is an assistive system that recognizes only one activity. Nevertheless, it is still interesting because it is a concrete working system. The second one is a novel approach that is based on the natural chemotaxis process of bacteria. It integrates the spatial aspect for activity recognition.

3.5.1 RIMER system

As it has been explained before, recognition approaches based on spatial reasoning is scarce. The work we present in this section comes from Augusto & al. [22] and is named RIMER. This team also works on the problem of technological assistance in smart home. They investigated the integration of spatio-temporal information into smart home algorithms because they believed, as we do, that space is a crucial aspect in monitoring activities. The first thing to know is that RIMER is a Rule based Inference Methodology using Evidential Reasoning that was developed by Yang & al. and published in [82]. Augusto and his team extended RIMER with an active database framework [83] in order to deal with spatio-temporal aspects of human activities monitoring. In fact, it is not a complete recognition approach such as the others we developed in this chapter. As you will see through the description of its functioning, it can identify very simple *situations* based on a kind of expert system. To validate their approach, they addressed a particular case study in which the occupant fainted or fell. Therefore, the spatial integration is mostly about the resident position. To follow the resident position, they combined RFID technology with infrared motion sensors. The resident had a tag attached so when passing through door RFID antenna would detect him, and motion sensors would tell in which room he entered. That is a pretty basic system, but it worked fine for their case study. Before explaining how they can detect if the resident fainted, let's take a look to the rule system.

3.5.1.1 Rule-based design

Active databases are characterized by their Event-Condition-Action (ECA) rules. They are designed to react to incoming information and have the following syntax:

ON <Event>, **IF** <Condition>, **DO** <Action>

The *event* part specifies the signal that triggers the rule whereas the *condition* must be filled in order to react. If the condition is met, the *action* part is executed. However, in smart home, events present uncertainty due to the lack of precision from sensors. There is also such uncertainty in the condition part and in the relation that links both. That is why Augusto & al. used *belief* rule instead of classical one. That kind of rule incorporates a degree of confidence in the statement. In their work, it is merged directly in the rule as follows:

> **IF** at_kitchen_on with *high* confidence **Followed_by** tdRK_on with *medium* confidence **Followed_by** no_movement_detected for 10 units of time
> **THEN** assume with 80% confidence that occupant is compromised

As you can see it is still very straightforward to understand. The events are highlighted in white, and *tdRK_on* is an acronym used to mean *transition* (td) *from room R* (R) *to room K* (K). The *transition* and the *position* state are the tools exploited for spatial reasoning. They also integrated functionalities to deal with time in order to position events in time in their belief rule. They choose to order events using only two temporal relations *earlier than* (<) and *simultaneous* (=). However, events are already ordered using classical logic connectives ($\wedge$, $\vee$, $\neg$) and only from the logic *AND* temporal relations are meaningful (if only one condition *OR* another is fulfilled, there is no need for temporal relations between the two). For that purpose, they introduced two new symbols: $\ddot{\wedge}$, $\overline{\overline{\wedge}}$. So if we have $A \ddot{\wedge} B$, it means A true and later B true too. For $A \overline{\overline{\wedge}} B$, it would means that A and B are simultaneously true.

3.5.1.2 General RIMER operation

The general architecture of RIMER is illustrated on figure 3.2. The two essential components to a rule-based system are the knowledge base and the inference engine. In their work, the knowledge base is a relational database where the rules are generated entirely by experts. So, in case of rules with confidence degree, experts have to exploit their judgment to approximate the real situation. However, they noticed in their paper that we could use machine learning techniques instead to extract that same knowledge. The inference system is classical one where rules have a *weight* to establish a priority in the case that many can be fired at the same time. If further conflicts remain, the evidence theory from [32] is used to discriminate.

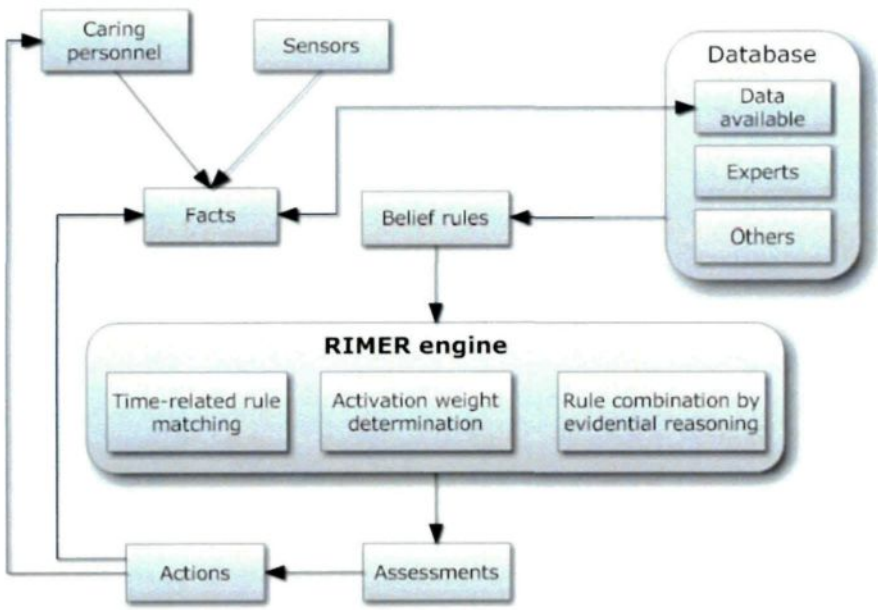


Figure 3.2: General RIMER architecture

3.5.1.3 Example scenario

Consider the problem we described in the earlier sections where an assistive system would try to recognize when the resident as fainted or fallen. An example of rule to detect such a situation in the kitchen would be that:

IF at_kitchen_on $\wedge$ tdRK_on $\wedge$ no_movement_detected
THEN assume the occupant has fainted

However, that rule must be adjusted with uncertainty. In their work, they define four levels of confidence: High (H), Medium (M), Low (L) and None (N). So the grades for *at_kitchen_on* are:

$$A_1^k \in \{H, M, L, N\}, k \in \{1, \ldots, nbRules\}$$

Similarly, the same grades are used for *tdRK_on* (A_2^k) and *no_movement_detected* (A_3^k). So the result confidence is represented in a belief distribution representing those four values for each rule. Consequently, if we update our preceding rule, it would give something like this:

IF at_kitchen_on with (H) Ä tdRK_on with (M) Ä no_movement_detected with (H)
THEN the estimation that the occupant has fainted is
{(H, 0.7), (M, 0.3), (L, 0), (N, 0)}

Here, the belief distribution means the system has a degree of confidence of 70% that the resident as fainted with high possibility and 30% that is as fainted with medium possibility. From these rules, the inference system can decide to assist the resident or not. In that situation, it would report that the resident has fainted to health authorities.

3.5.2 CHEMOTACTIC MODEL

The work we are going to present in this section is surely innovative. Riedel & al. [23] had the idea of creating a better model for activity recognition using spatial aspect from a chemotactic model. The idea comes from the world of bacteria where a process, named chemotaxis, allows bacteria to directionally swim in response to a chemical or other physical gradient [84]. For motile bacteria such as the *Escherichia coli* (commonly named E. Coli), it acts by either attracting the cell to an increasing gradient or repelling from harmful regions by moving in the direction of decreasing gradients [85]. The researches on E. Coli have shown that bacteria sense spatial gradients as temporal changes in attractant or repellant concentration. The cellular chemotatic model of Riedel & al. uses an abstraction of that process to represent and recognize ADL in a smart home.

In the model, a cell is composed of receptor type $\{R_i\}_{i=1}^{n}$ that works to match *molecule* from the environment. An activity is a group composed of cells. A molecule is a spatial symbol $u \in U$ where U is the set of all possible symbols. The notation $|R_i|$ denotes the specified number of receptors for each receptor type. Thus, the total number of receptors of a cell is given by $p = \sum_i |R_i|$. The environmental space E is a two dimensional Cartesian plane where the cells possess a pair *x,y* positioning it in *E*. The place where molecules are conceptually released is set to the origin point. Cells are, however, positioned to (1.0, 0.0) at the beginning. Cells possess a velocity property *v* determining the movement within the coordinate space *E*. In the model, release of molecules *u* in environmental setting *E* increase the concentration and can be detected by cells with a free receptor of the same receptor type. Since the highest concentration of molecules is known in the model, cells know exactly the direction to travel and move toward the attractant. A condition subsists: the cell must not be already in a zone with high concentration.

Chemotactic cells possess a memory associated with the irreversible binding of molecules to receptors that grant them the ability to detect increasing environmental concentration. That memory capability is determined by the fixed maximum number of receptors $|R_i|$ of each receptor type R_i that the cell possesses. If a cell does not possesses a free matching receptor after the increase of an environmental chemical increase (after the release of a molecule u) but still possess receptors of that type, the cell will perform a random walk. When cells move close to the attractant source so the Euclidian distance d between the cell and the origin is less than the high concentration threshold, they perform a random walk irrespective of increasing concentrations. Otherwise, it returns to normal behavior. Cells with a higher degree of similitude to tests sequences will get closer to the attractant source. The molecules representing an activity sequence are *released* into the chemotactic environment consisting of β classes with m cells per class. Then, we find the cell ϕ in Z, where Z is the set of all activity cells, which has a minimum Euclidian distance to the attractant source δ of E according to (2). The minimum distance cell ϕ is then used in the classification decision.

(2) $$\phi = arg \min_{g \in Z} d(g, \delta)$$

3.5.2.1 Methodology and experiments

Now that we have reviewed the chemotactic model, let's look at how they implemented it concretely in a smart home infrastructure. The first step to do this was to learn spatial knowledge of all the activities to recognize. To do so, they used the multiple camera tracking system of Nguyen & al. [86] to build a dataset comprising six activities:

getHomeWatchTV, *haveSnackWatchTV*, *atHomeWatchTV*, *readingNewspaper*, *havingBreakfastToast* and *havingBreakfastEggs*. The figure 3.3 shows the smart home architecture and the spatial path for each activity's sequence.

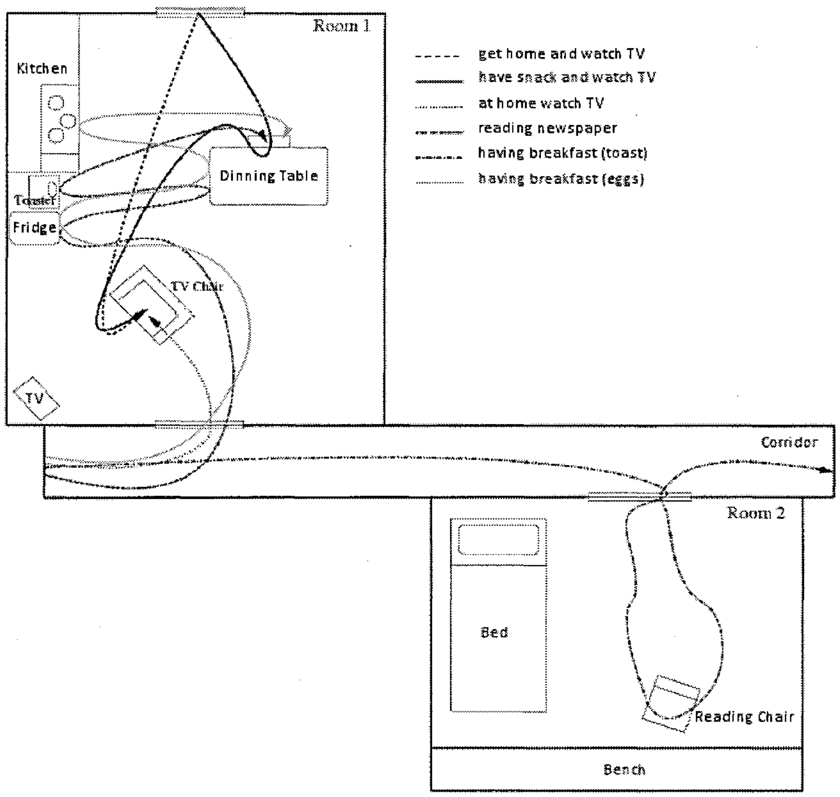


Figure 3.3: Spatial path for each of the six activity sequences [23]

They needed a basis to compare the efficiency of the chemotactic model. To do so, they used Hidden Markov Model (HMM) [87] that allowed them to recognize spatial activity patterns. They divided their smart home into logical spatial zones and created a model that had 156 different spatial states for the HMM construction. A HMM was required for each activity class. The 156 states were corresponding to the set U of the chemotactic model. To test the model, it was required to transform two dimensional coordinates (x, y) into a unique integer that would exist in *U*. From that point, they conducted experiments on the six different activities and demonstrated that their model was significantly better than common HMM built for spatial activity recognition. The table 3.2 shows these results.

Table 3.2: Results for performance comparison of the chemotactic and HMM.

Technique	Precision (%)	Recall(%)
Chemotactic model	99.75%	100%
HMM	90.75%	90.75%

They did not only stop their tests at this step. They also compared it with different numbers of training sequence (1, 2, 5, 10). They demonstrated that the chemotactic model was efficient with low training (90% accuracy with only one training sequence) unlike the HMM. They also conducted tests on simple interleaved activities. The results were similar to the previous tests so it was concluded that chemotactic model could successfully recognize simple interleaved ADLs.

3.5.3 APPROACHES COMPARISON

We presented two approaches integrating spatial aspects. The first one from Augusto & al. [22] was more of an assistive system than a recognition platform so only one activity was recognized, and it was a very simple one (the person has fainted). However, the assistive idea was clever and could be a concrete system rapidly transferable from the laboratory to the population houses. Their experiments were also conducted in a realistic smart home context using non-invasive sensors (RFID, IR motion sensors) on the contrary to Riedel [23] that used only video camera. On the other hand, it was required for the resident to wear a device, and many therapists and researchers consider it too much intrusive [59]. The system itself possesses the advantage of being a rule-based framework that are simple and easy to valid or verify. A major limitation comes from the fact that this system supposes that we can recognize basic action.

The approach of Riedel & al. [23] was an original work from the natural chemotaxis process of bacteria. It provided a major improvement over works directly based on HMM, and the computational complexity was demonstrated to be lower (polynomial). It also partially addressed the problem of interleaved activity. The model gave very good recognition performance. However, it was based on very simple ADLs (watch TV) and thus would not fit for cognitive assistance. Moreover, both Riedel and Augusto works did only integrate the spatial aspect in a very limited way where only the resident position was exploited.

3.6 MACHINE LEARNING AND PROBABILISTIC METHODS

We stated in the beginning of this chapter that every approach could integrate machine learning techniques and probabilistic methods. What are the consequences of such methods? Learning techniques allows personalizing algorithms to the patient routine. However, it often requires a large set of training data to perform well. That is a major downside. That is why it should not be used as standalone recognition approach. However, we think that it can be used as improvement on working existing approaches. Probabilistic methods on other hands can be very interesting when the system's IA cannot reach a conclusion on one plan. For instance, the plans *MakeTea* and *MakeCoffee* would be equally probable in Kautz traditional model. However, it is possible that the resident has a preference for coffee, and it would be more probable that he executes the activity *MakeCoffee* rather than *MakeTea*. Thus, a probabilistic algorithm could discriminate between these two options. Similar results could be obtained from learning techniques. Still, probabilistic methods possess the advantage of being simpler to implement and more universal (from user to user) requiring fewer adjustments. Recognition approaches that are solely based on probabilistic methods such as Hidden Markov Machine (HMM) or Dynamic Bayesian Network (DBN) give very good results in matters of recognition rate. The problem is that they are very hard to ameliorate (update) to recognize more activities. Moreover, often it requires one model for each activity that we want to recognize. Because of this, they are limited to restrain recognition of small activities on a very concise field.

3.7 CONCLUSION ON EXISTING WORKS

A lot of effort was done in the research on smart home, and the technological side has greatly improved over the years. However, we now have technologically enhanced homes that are not smart enough to be useful. That is because the problem of activity recognition inside a smart home is very challenging due to the partially observable, stochastic, dynamic and sequential context that is the hardest possible instance of context in artificial intelligence [63]. In this chapter, we presented a literature review of three categories of recognition approaches: logical, temporal and spatial. We described the advantages and the characteristics of each of them. We also described the limitations of these approaches; such as that none provide perfect recognition rate and that they only partially recognize erroneous plans. Furthermore, many do not address the problem of recognizing activity from low level sensors output but instead suppose that basic actions are easily observable. In addition, none of the approaches address the specific spatial issues that were described in the chapter 2 of this master thesis. As we demonstrated, these issues are realistic and important in cognitive assistance context and therefore, should be addressed. The next chapter will propose a new recognition model that will specifically address these issues.

CHAPTER 4

QUALITATIVE SPATIAL REASONING RECOGNITION MODEL

4.1 INTRODUCTION

The previous chapters had the purpose of reviewing the different works that exist on the spatial aspect in research and more specifically on the recognition algorithms built for the purpose of cognitive assistance. We also have talked, in chapter 2, about general frameworks that exist to infer and reason about space. As we showed, many formals calculus have made their proofs as valid models. One of them, from Egenhofer & Franzosa [25], is based on general topology a mathematical field that is built on a solid foundation resulting from many years of research. It provides a great methodology to compute the relationship that exists between two objects in a space of any dimension. Still, it lacks important properties to be suitable to use as a recognition algorithm for activity recognition. First of all, the framework does not provide a rigorous inferential method to circumscribe a set of activities. In fact, none of the frameworks are directly applicable on this situation and rightly so, because they have not been developed for this purpose. Also, this framework and other are not built to detect problems in the recognition of activity. There is no mean for establishing constraints on spatial relations. The recognition approaches we reviewed, in the previous chapter, possessed many advantages and offered good performances. However, since we try to recognize activity in an applicative context that is very complex and offer only partial information, they do not fill the entire requirements. For instance, none of them address the problems related to spatial aspects (position, distance and orientation) that we described in chapter 2.1.1. As we have seen, these situations are key issues in trying to correctly recognize activities and discriminate between similar hypothesis and potential errors. Moreover, recognition rate of these related approaches is good but not perfect and must be improved on some specific aspects (early detection rate, correct duration, etc.). These aspects will be described later in detail.

Looking to the whole problem of cognitive assistance inside a smart home, it appears that we clearly have work toward us and in particularly in ADLs recognition. In order to overcome some of the challenges that remain in this field, we will describe through this chapter our scientific contribution, which takes the form of a model that we developed for activity recognition based on spatial reasoning. The first part will describe the theoretical aspects of the model, which are based on the current advances of qualitative spatial reasoning (QSR). Secondly, we will detail the concrete implementation of this new model at the LIARA laboratory. In the third section of this chapter, we will present the validation and the tests that we conducted to evaluate the model efficiency. These tests were based on real case scenarios extracted from clinical trials that were previously conducted by the team with AD patients. Finally, we will analyze and compare the results we obtained with existing and new metrics in relation with other models.

4.2 FORMALIZATION OF THE NEW SPATIAL RECOGNITION MODEL

The basic idea of the new model is to reuse the knowledge acquired in the field of qualitative spatial reasoning and apply it to our specific recognition context to improve the algorithm precision. We decided to exploit the topological model of Egenhofer because it is the most robust and the most expressive. However, due to their similarities, we could have used RCC from Cohn & al [31] without much impact on the theoretical part and on the performance (see section 2.6). Our model is based on the eight types of relations from Egenhofer. Altrough, due to the implementation detail, and sensors limitations that we will explain later, it is very hard to distinguish between *covers* relation types and *overlaps* so in reality they have not been integrated. The relations are illustrated as a reminder on figure 4.1 (with a red dashed line are the relations indistinguishable).

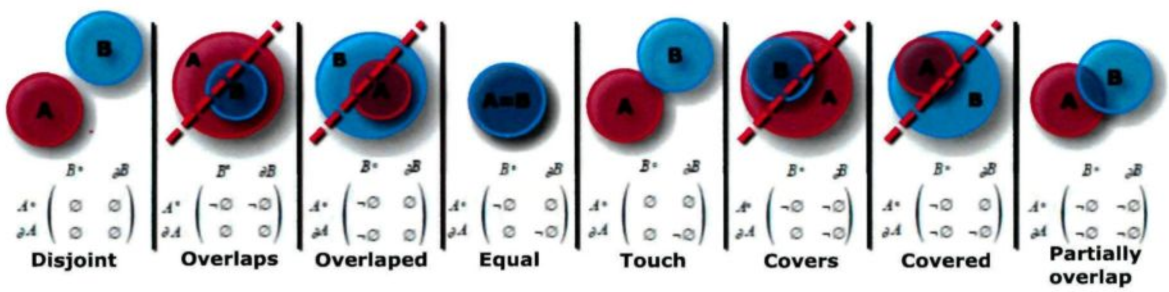


Figure 4.1: Eight relation types from Egenhofer & Franzosa [25]

As argued in chapter 2, qualitative spatial reasoning is best suited for our environment. To determine the relations and the positions, we use real coordinates of the objects. It does not change that it is purely qualitative SR, because the numbers are not used for the spatial representation. For instance, we use the coordinates to find the position of the object cup, but we use it to associate the position with a qualitative criterion: *the cup is on the kitchen counter*. In chapter 2, we also discussed about using the primitive point or region. In our model, we decided to use region as the primitive entity to keep it simpler. From this information, this first part of the chapter will present the theoretical aspect of our new spatial model for recognition.

4.2.1 PRIMITIVES DEFINITION

Before beginning the description of our new QSRR model, we must first pinpoint a thing about the representation of the physical environment. To reason with space, we use C a Cartesian plane where anything can be spatially situated with coordinates $c(x, y)$ where x is the position on the abscissa and y the position on the ordinate. Anything having certain coordinates can occupy a *logical area*. A logical area $a \in A$ is defined as a pair (c, r) where $c \in C$ is the center of the area and r the radius of the circular area comprises between 0 and ε a certain constant. These logical areas are very important; they are used in our algorithm to reason with Egenhofer's relations. An example can be seen on figure 4.2.

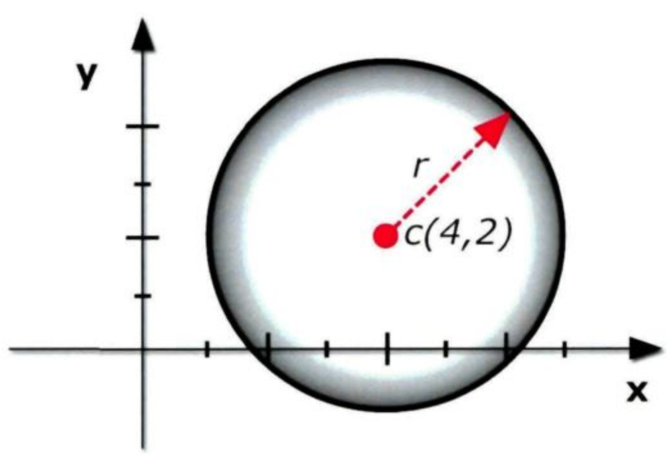


Figure 4.2: Illustration of the *area* concept

Keeping these primitives in mind, we will now introduce our new qualitative spatial reasoning recognition model as a formal structure <C; A; O; Z; S; P; E >. First of all, we have O={Cup, Coffee, Spoon, …} that is the set of all object types that exist in the environment. To each of these general types (concepts) are associated real physical tagged objects (instances). There can be many objects of a type but only one type per object. For example, there can be a type *Cup* that is the type of three objects named *cup1, cup2* and *cup3* but none of this object can be of another type such as *Glass*. An object instance $o_x^a \in I$ the set of instance, is a pair (a, x) where $a \in A$ and $x \in O$. So every object instance covers a certain area at all time that is part of C the Cartesian plan of the model.

Secondly, we must introduce Z the set of different zones inside the environment. The zones divide the smart home environment into logical parts that can be used to reason about space. Let $z_i \in$ Z be the zone that is associated with a logical area *a* predefined as rectangular prism (*rp*) using two opposite points c(x,y) and d(x,y) in the Cartesian plan C: $a^{rp} \in A$ where $(c_x < r_x) \wedge (c_y < r_y)$. The zone covers a mutually exclusive area of the Cartesian plan of the smart home:

$$\bigcap_{i \in Z} Z_i = \emptyset \wedge \bigcup_{i \in Z} z_i = C$$

An example of a set of zones could be Z={(kitchen, (0,0,0), (100, 100, 100)), (Room, ..., ...), …}.

The next parameter of the seven-tuple is S, the resident of the smart home. S is a structure that keeps track of where the person is and at what time he or she got there. It will also be used to establish activities' plausibility.

4.2.1.1 The set of activities P

Now that we have described the basic sets O, Z, S from our seven-tuples < C; A; O; Z; S; P; E >, we must have a word on the primitives that will be part of *P*, the set of activities. Let *K* be a set of constraints in which a constraint *k* is a spatial relation between two logical entities such as $K \subseteq O \times O \cup O \times Z \cup P \times Z$. The set *K* is needed to use Egenhofer's theory. The definition says that a relation can be defined between various types of entities. In other words, $k \in K$ can be defined between two object types (ex: Cup [overlaps] Coffee), between an object and a zone (Cup [disjoints] Bathroom) or between a plan and a zone (Washing hair [disjoints] Room). These relations will later be used by the algorithm to determine plan's plausibility and then circumscribe the library to end up with a hypothesis of the current ongoing activity. A property of the set of constraints is that for all $k \in K$ there is at least one plan *p* it helps the realization.

The set of possible activities *P* represents the knowledge base of our agent. In addition to a set of actions, a plan $p \in P$ is characterized by a set of objects $\{o_1, \ldots, o_n\}$ and a set of spatial constraints $\{k_1, \ldots, k_n\}$ that are integral part of its definition. It should be noted that the agent believes *P* to be exhaustive and will only search within his library. For instance, the plan *MakeCoffee* has a spatial constraint (Cup [covers] Spoon) which literally says that a cup zone should (but not must) contains a spoon and its zone at the end of the activity. Observing this relation or a similar one between a spoon and a cup, the recognition agent would give more plausibility to the plan *MakeCoffee* so it could be determinant to the discrimination of other activities such as *CookPasta*. We will provide more constraints examples between different types of entities in the subsequent sections.

4.2.1.2 The set of events E

The last element of our model is the set E which contains all the events observed in the habitat by the observation agent. An event $e \in E$ is composed of a timestamp linked to a basic observable action structure. Here is an example of what E might look during the recognition process: {(1, Movement, Cup2), (2, Undetected, Glass1), …}. It should be noted that events are inserted on specific instances of object. That is because we observe real instance of objects in the smart home and not *type* of object. However, the recognition agent will reason using the type of these entities. So, in our example, it will infer on the basis that a cup has moved whichever the cup is. Finally, we must say that our basic actions are very simples. In fact, they are only the result of fusion of sensors' raw information at a time t in contrast with basics actions used in models such as Kautz formal theory for plan recognition [28] that often reflect a more complex series of sensors activation (*BoilNoodle*). The advantage is that our basic actions are very easy to identify and yet meaningful. However, an activity sequence will comprise a bigger number of steps for its realization.

4.2.2 OBSERVATION AGENT

Before going into details of the recognition process, it is important to discuss the observation agent functionalities. This agent has the role of coordinating all agents observing the environment. From the various sensors' output, he infers the current position of the resident, and if it has changed, he inserts this as a new event. He also looks at all objects in the smart home to check if one of them has changed position. If an object has moved, the agent asserts from the set of constraints that it is not in a forbidden area. That is, it checks the position issues we described in chapter 2. More precisely, we have to verify that an object logical area is not completely contained in a forbidden zone. If the object is in spatial relation such that he is entirely covered by a forbidden area (covered, overlapped or equal), the agent will raise a flag on this object to be ignored by the recognition agent in his inference process of the ongoing plan. We must do this before beginning to circumscribe the knowledge base in order to ignore some noisy observations that could lead to misjudgment of the activities' plausibility and consequently, restrains the agent from elimination of many impossible activities. In an assisting system, it could have triggered the assisting agent or sent a report to a caregiver responsible.

4.2.2.1 Position issue example

To be sure that our point is rightfully expressed, let's take an example where we have Peter, an Alzheimer's patient, in earlier stage of the disease. Peter wants to make a coffee. Therefore, his first action is to open a panel cabinet and take a cup. Hence, the system observes the event (1; movement, cup). Then, he takes the coffee from the cabinet and begins boiling water. Current event set is E={(1; movement, cup), (2; movement, coffee), (3; turnOn, kettle)}. Then, Peter goes to the bathroom and washes his hands afterward. However, due to memory impairments caused by his illness, he brought back the bottle of shampoo with him in the kitchen. Thus, the new event (4; movement, shampoo) is added to the set E. It is rather simple for a human to see that it is a mistake, because that bottle of shampoo should never be part of any kitchen activity. Nevertheless, in the system, we translate it by verifying the relation of the shampoo with the forbidden areas (the kitchen) that are specified in the library to finally conclude that we should ignore this observation or consider it as an error in the recognition process. If not ignored or classified as erroneous behavior, the system would not have eliminated the activities related to the bottle of shampoo that were obviously incoherent, and it might have led in significant errors in the recognition process by considering plausible activities such as *WashingHair*.

4.2.3 EVALUATION OF ACTIVITIES' PLAUSIBILITY

The activities' plausibility is evaluated every time the recognition agent receives the message that a new event has happened. To determine this plausibility as a percentage of others, each plan first receives a certain number of points at every stage of the recognition process. These points, represented by the symbol δ, are given by the spatial reasoning on the constraints k on each $p \in P$. Then, we can calculate the plausibility points pts_t^δ of an activity at a certain time t that is:

(1) $$pts_t^\delta = pts_{t-1}^\delta * 75\% + \delta$$

The equation means that the current recognition is not only influenced by current state of the smart home but also by past state to a lesser extent. All activities are initialized at 10 points and thus are equally plausible before the first iteration. The 75% constant as been chosen based on multiple tests that gave good results. Increasing it would give more influence to past observations and decreasing it would reduce it. However, that constant is not too critical at the condition that it is not too high (90% would prevent the algorithm to shift rapidly its decision on another plan). Let's take a look at what the recognition agent does to calculate the points δ of an activity.

4.2.3.1 Updating the spatial information

The first thing that happens during recognition iteration is that the spatial analyst agent is called to update the spatial relations between all pairs of objects in the smart home. Since it would mean $O(n^{\log n})$ complexity (n being the number of objects), in reality, these relations are kept into the agent memory and only the relations of objects that have moved are updated. In addition, objects that are at their initial position (their home position) are not considered because they might naturally be in a spatial relationship with other objects that are in a close initial position. As a result, these objects have all their relations with others object as undefined. Otherwise, the relation between two objects is determined by calculating the distance between their positions (center of their disc area) and comparing with their radius and diameter. For instance, if the distance between both centers is bigger than the sum of both areas' radius, we can know for sure these objects are disjointed. If the distance d is smaller than the sum of the radius of both areas (r_1, r_2) check if $d + r_2 < r_1$

then the area with r_1 radius covers the other one. These two examples are illustrated on figure 4.3 below:

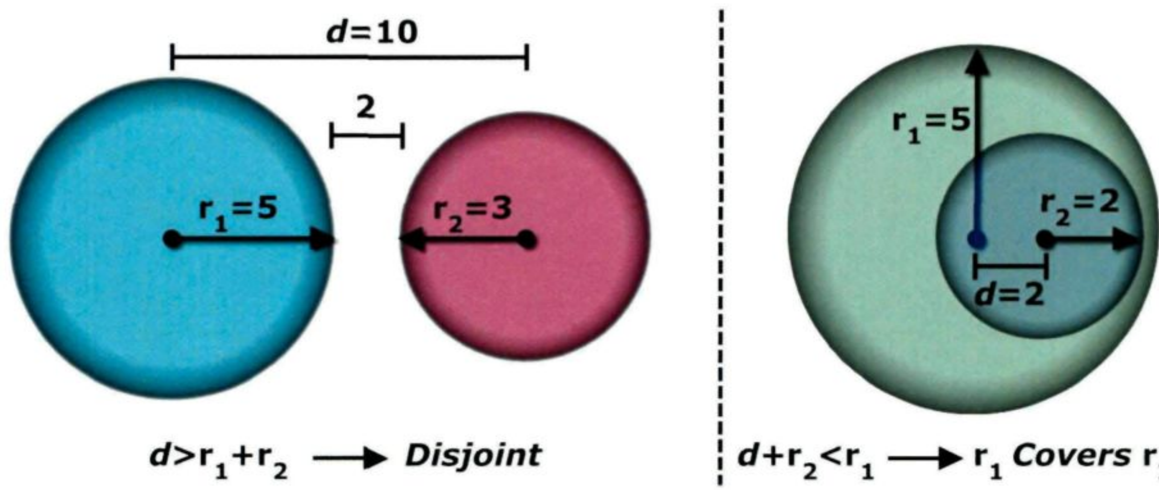


Figure 4.3: Illustration of the spatial relation determination between two circle areas

4.2.3.2 Basic events evaluation

The second step of the recognition process consists to add plausibility points to each plan for the events that have happened. That is the non-spatial part of the recognition process. At this step, the algorithm lookup through the events and adds points for each action recognized that is part of a plan. We say that it is non-spatial because an event could be happening outside of the activity zone or could be an error from the resident. For example, at this step, we would add points for the action (5, Movement, Coffee) because it could be a step part of the plan *MakeCoffee*. However, on later stage of the recognition process, we might discover the coffee has been moved away considering the distance spatial aspects (increasing).

4.2.3.3 Spatial relations between pair of objects

Another agent in the recognition module has the task of assigning the spatial bonus points to every plan from information previously collected. To do so, for a plan, it checks the relations of all the pairs of objects that are part of it. Each of these relations can increase or decrease the plausibility of a plan depending of what is specified in the activities' library. To determine if a relation increase or decrease the plausibility, we must split the eight Egenhofer's types of relation into two groups. It is straightforward to do since only the disjunction between two objects does mean the contrary than the other types. In fact, the seven other are only differentiated by a matter of connection force (in terms of meaning). Therefore, the *disjoint* relation has a negative impact when the others have a positive one. The only problem remaining is to determine when a group should increase the plausibility or when it should decrease it. It is uncomplicated to determine since we have predefined spatial constraints in our library for each plan. Then, if two objects are in disjunction relation and the set K specify a constraint of any other types such as ($object_1$ [partially overlaps] $object_2$), the plan p will lose plausibility. Otherwise we just need to reverse the value from positive to negative. Finally, it is important to understand that some relations are stronger than others. For example, two objects that are in relation "partially overlaps", they have a stronger bond than in relation "touch" but weaker than "equal", "covers" and "overlaps". Using this information, the influence of a relation on the plausibility of a plan will be modulated in function of the relation type that is specified in the agent's knowledge base.

4.2.3.4 Subject position bonus

The last part of the points attribution (δ) step is to influence the score of all activities using the relation of the resident position with the zones of the smart home. In other words, if the subject is in a certain zone where it is common to realize some plans, these activities' score will be enhanced. Again, these constraints are defined in K the set of spatial constraints in the library. To be clearer, let's say the person is in the bathroom during the time t to $t + \varepsilon$. During that interval, the activities having affinities with this zone, such as *TakeBath*, will be more probable. Even though this bonus above is not sufficient to discriminate between activities, it is a good spatial criterion that helps to guide the algorithm in its decision process.

4.2.4 LIBRARY CIRCUMSCRIPTION

To circumscribe the activities' library, we transform the plausibility points (pts) earned by each plan to a percent plausibility (in function of other plans) defined by the following equation:

(2)

$$\varphi = pts \Big/ \sum_{p \in P}^{n} p_n^{pts}$$

In the equation, the denominator is the total plausibility points of all activities. Then, given the plan plausibility ordered from highest to smallest, the set of inferred plan hypotheses $P_h \subseteq P$ is defined as follows:

(3)
$$P_h = \bigcup_{p_i}^{P} Equal(p_1^{\varphi}, p_i^{\varphi})$$

We will always take the one with the highest plausibility because we want at least one hypothesis. Then, the next activities will be considered only if their plausibility φ is equal to highest plausibility. The function *Equal* compares the plausibility rounded to the units. So, if *MakeCoffee* is 30% plausible, and *MakeTea* is 29.5% plausible, they will be considered as equal since the difference is too small to take a decision. Figure 4.4 graphically illustrates the steps of our model that we have been explaining. In the next section, we will give a complete example using our model.

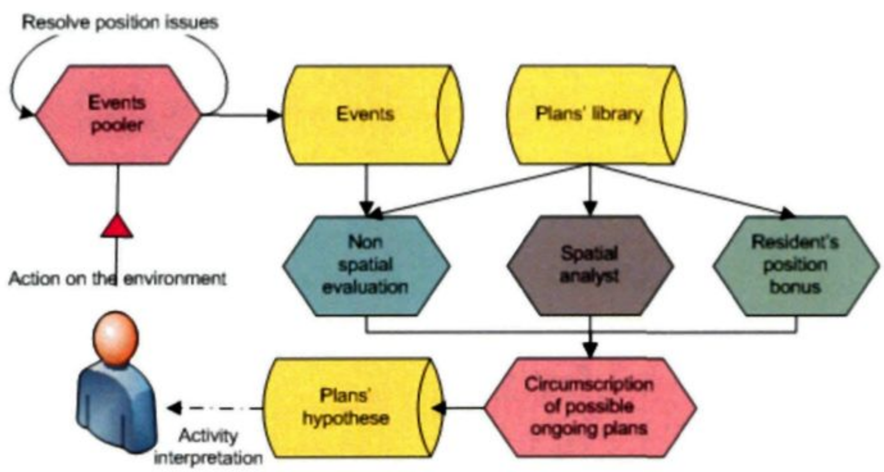


Figure 4.4: Summary of steps done by our algorithm to recognize an ADL

4.2.5 EXECUTION EXAMPLE

This section will display with a simple example how it helps to determine the ongoing activity with the spatial criterions. Let P={*MakeTea* (*MT*), *MakeCoffee* (*MC*), *ReadBook* (*RB*)} be our example activities' library with all initial plausibility of 10 points.

Since the activities are all equal, they each have 33.3% chance to be the ongoing activity. Again, let's suppose that we have Peter living in his smart apartment doing is everyday life activities. His position, detected by infrared sensors, is believed to be right in front of the kitchen counter (no other event has been observed). Considering that information, the algorithm checks the set of spatial constraints K to find relations of type $Z \times P$ and find that both the *MT* and *MC* plan possess the relation (*kitchen [covers] MT/MC*). Consequently, during the first recognition iteration, a resident's position bonus of 10% is added to the two first activities (*MT, MC*). Therefore, the points become {*MT:* 11, *MC:* 11, *RB:* 10} because 10*110% equals 11. That new information gave a slightly better plausibility of 34.3 % for *MT* and *MC* compared to 31.3% for the activity *RB*. The result is that the plan *ReadBook* is ignored and *MakeTea* and *MakeCoffee* are both considered as possible (this iteration at least). Then, Peter takes a coffee cup from one of the cabinets and deposes it on the countertop of the kitchen. The observation (1, movement, cup) is added to the set E and the recognition process begins. The first step consists to calculate points for this iteration's events. Since the action *TakeCup* is part of both *MT* and *MC* activities, they receive 10pts for the event. Next, spatial module checks if there are objects in relation, but it founds nothing this turn. Consequently, their actual plausibility points at the end of the second iteration are {*MT:* 20.1, *MC:* 20.1, *RB:* 7.5} given after the following calculations $(11*75\%+10)*110\% \cong 20.1$ and $10*75\%+0 \cong 7.5$ (the 110% is the bonus for the resident position). As you can see, *RB* has loss points from its initial plausibility. That is correct because we give the 10 points at

> The relation type for position constraints is not so much important, and we always use "covers" or "disjoint".

the beginning in order to avoid a score of zero plausibility for a plan. The equation used here is the one we explained in the previous section (1). As a percentage, their plausibility has now become: *MT(42%)*, *MC(42%)* and *RB(16%)* dismissing further the plan *ReadBook* from assumptions. Two activities remain as equally plausible, but then the system observes (2, movement, tea) so obviously Peter seems to prepare tea. Then, the system observes (3, movement, coffee) which would have re-established the initial plausibility to be approximately equal between the two activities. Let's take a look at the calculation supposing that we do not take spatial relations between objects into account. The observation 2 gives 10 points to the activity *MT* but zero to the other two. Thus, the new plausibilities look like this:

MakeTea	=	(20.1*75%+10)*110% ≅ 27.6
MakeCoffee	=	(20.1*75%+0)*110% ≅ 16.6
ReadBook	=	7.5*75%+0 ≅ 5.6

After that, we would observe the third event that would add 10 points to the plausibility of *MakeCoffee* given us *MT:*22.8, *MC:24.7* and *RB:*4.2 (or in percent respectively 44.1%, 47.8%, and 8%). The conclusion would be that the resident is making coffee. However, with the spatial analysis, the relation (tea [partially overlaps] cup) would have been observed between the items and this relation would have increased the plausibility of the plan *MakeTea*. Let's now take back the plausibility points as they were before the last two observations: {*MT*: 20.1, *MC*: 20.1, *RB*: 7.5}. The second observation would then have added 10 points as previously but now the spatial relation would have given a 10 points bonus totalizing 38.6 plausibility points at the end of the iteration for *MakeTea* and unchanged amount of points for both *MakeCoffee* (16.6) and *ReadBook* (5.6).

Then, the observation (3) where the coffee is moved would again increase the points of *MC* by ten. However, the spatial analysis would see a disjunction between the coffee and the cup giving a penalty of 10 points. Thus, the final iteration would give the following plausibility points:

MakeTea	=	(38.6*75%+0)*110% ≅ 31.8
MakeCoffee	=	(16.6*75%+10-10)*110% ≅ 13.7
ReadBook	=	5.6*75%+0 ≅ 4.2

At the end, the system would observe the plausibility is much higher for the latter plan and then reduce the set of plan hypotheses to $P_h = \{MakeTea\}$. In reality, the action of taking the coffee done by Peter could have been a misjudging error due to AD so it was put back to its initial position. The example shown was a really simple version of what might happen in the reality. However, it was really clear in that case, without observing the distance between related objects that we would not have been able to adequately discriminate through the different hypotheses.

4.2.6 REVIEW OF OUR NEW QSRR MODEL

This first section of the chapter 4 had the goal of presenting the theoretical contribution of this master thesis. It presented a new recognition approach based on Egenhofer topological framework for qualitative spatial reasoning [25]. The new model use QSR to better evaluate plans' hypotheses during the recognition process and to deal with spatial issues that may arise during the realization of an ongoing activity. The next section will describe the implementation of this new model inside the smart home of the

Laboratoire d'Intelligence Ambiante pour la Reconnaissance d'Activités (LIARA) of the *Université du Québec à Chicoutimi* (UQAC).

4.3 IMPLEMENTATION AT THE LIARA

This section will describe the context in which we have implemented our new model. It will begin by a description of the smart home infrastructure. In that section, we will talk a little bit about the sensors used and their role in the model. Also, we will detail the software that was developed to test the algorithm. Finally, the section will conclude on a description of the library and its implementation.

4.3.1 THE LIARA INFRASTRUCTURE

The LIARA laboratory possesses a new cutting edge smart home infrastructure that is about 100 square meters and possesses around a hundred of different sensors and effectors. Among the sensors, there are infrared sensors, pressure mats, electromagnetic contacts, various temperature sensors, light sensors and eight RFID antennas. We also have many effectors, including an Apple© iPad, many IP speakers around the apartment, a flat screen HD television, a home cinema theater and many lights and LEDs hidden in a strategic position, and that we can control at distance. The figure 4.5 shows a cluster of images from different parts and angles of our smart home.

Figure 4.5: Pictures of the LIARA's smart home

The main image is the kitchen. At the bottom from left to right you can see: a tagged cup (RFID tag), the dining room, an RFID antenna, the HD television. From top right to bottom can be seen: the server, the bathroom and the library. The server is a Dell© industrial blade computer, and it is the one in charge of processing the information. We also have an AMX© system to control multimedia hardware such as the DVD player, the television and the IP speaker. As shown on figure 4.5, the iPad is embedded in the refrigerator. It controls the habitat for the experiments or to test the equipment or to assist the resident with the help of videos when he is in the kitchen. Talking about assistance, the television can also be remotely controlled from computer (or AMX) for that same purpose. Our respective offices and a meeting room are built around the intelligent home. Moreover, we can see inside the apartment with the windows mirror specially designed for our experiments.

4.3.1.1 Smart home software

We developed software for the means of controlling our smart home. A screen shot of this software showing the overall smart home can be seen on figure 4.6. The graphical interface of this software allows us to see different part of the smart home or the overall picture. In each of these interfaces, we can see the state of many sensors such as infrared sensors, light sensors, etc. We also can see an approximate location of the objects in the smart home (rounded rectangle, only appears if RFID antennas are activated) and the current position of the resident (in front of the kitchen counter on figure 4.6). From this software, we can enable/disable any light in the apartment. We also can control the oven and even shut it down in case of emergency. The most important feature is certainly the scenario recording capability. That allows us to record everything that happens in the smart home in real-time until we stop it and then play it back again. During recording, not only the state of sensors is recorded, but also the RFID signal sent by objects to the antennas. In addition, it also saves everything those effectors do in the smart home. So if a light flash, the television is opened or the speakers play audio, it will be recorded. That useful feature had been exploited in the experiments of this project. More details will be given in the third part of this chapter.

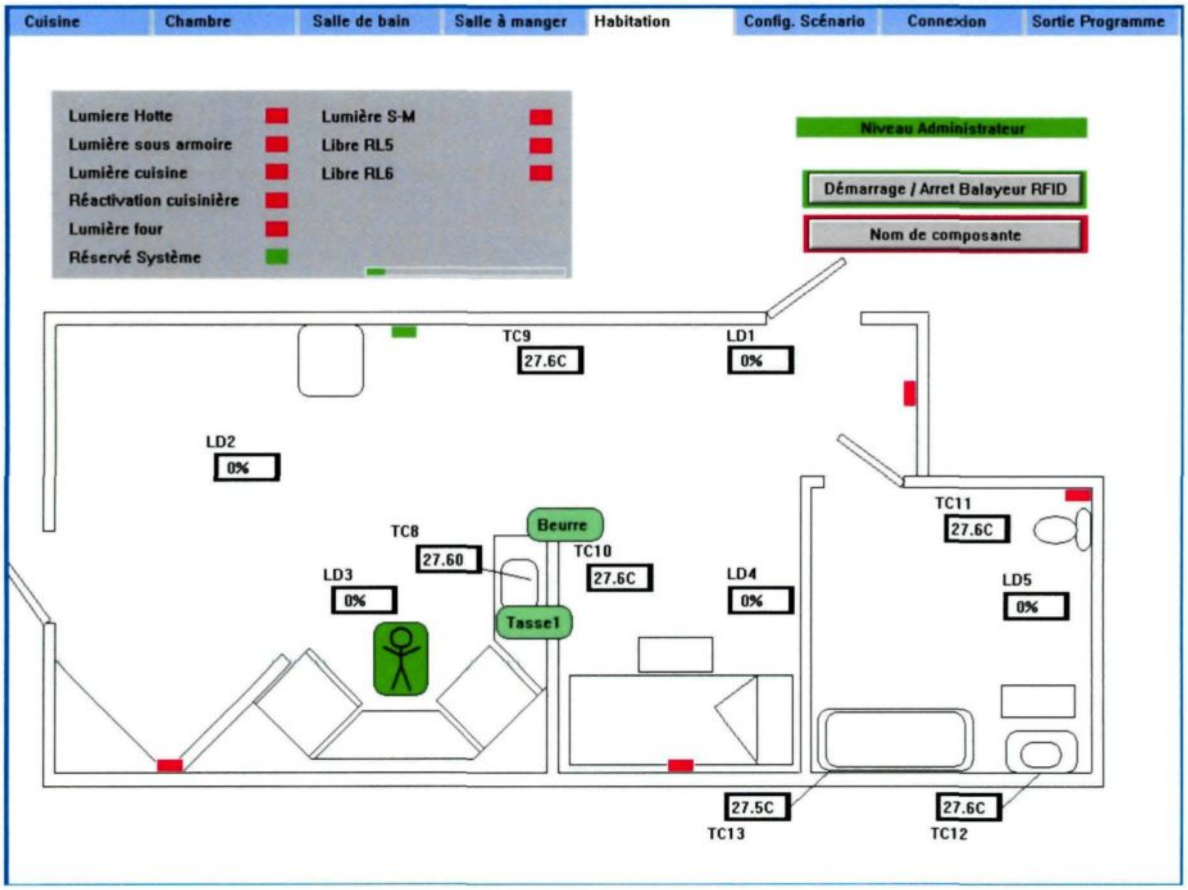


Figure 4.6: Screen shot of the smart home controlling software

4.3.1.2 Sensors to locate the resident

In the implementation of the model, we required information from two types of sensors to locate the resident. First, we used pressure mats. This kind of devices is very precise and allowed us to know with a hundred percent certitude that the resident was in a certain position (unless that an intruder is in the house). However, the pressure mats do not cover the entire floor of the infrastructure because they are costly. Therefore, we had to use another technology to locate the resident on area where there was no pressure mats (the most part of the smart home, in fact). For that purpose, we used infrared motion sensors.

These sensors are, however, only active for a short amount of time, and alone they are not much precise. For that reason, a locating agent was programmed to accompany the recognition software. That agent uses a rule-based inference engine that allows him to track the position of the resident from the series of motion sensors activation and from information from other sensors such as the tactile mats. Sometime, the agent might lose track of the resident, in that case he will consider that he has not moved. Yet, if that situation happens, no bonus will be given to activities for the resident position.

4.3.1.3 Using RFID technology

For the implementation of our software, we primarily used the RFID technology. We used passive RFID tags (class 3) that are self-powered and possess no memory. That kind of tag works with the help of the RFID antennas. The antennas send a pulse in the form of radio waves every few milliseconds, and then they listen to receive any response from the surrounding tags. A passive tag is in a dormant state, and it is receiving one of these waves that wake it. Thereafter, it draws energy from that same wave to emit at its turn a radio wave containing information on its ID. Of course, when comparing to active tags that use their own power to emit and that are always awake, passive tags have a reduced range and precision. However, they are much smaller and thus better adapted for our situation. Passive tags are also characterized by their low cost (often less than a penny) compared to other possible solutions. These two qualities are what allowed us to tag all objects in the smart home. The kind of tags we used is so robust, that the objects tagged are washable without damaging the tags. Consequently, it is realistic to propose such utilization for them and could be used in a resident home. One of these tags can be seen on figure 4.5 at the bottom leftmost corner.

To deal with the positioning of objects, we could also have used GPS system or video cameras. The GPS is a great system, but it is not precise enough for identifying relation between close objects inside home. Video cameras sure offer the most complete information and the utmost precision, but as Patterson [68] highlighted computer vision is far from being able to deal with the complexity of that amount of information in a foreseeable future. Another reason to use RFID technology is that it is less invasive than any other means to position objects (glove, camera, etc.). It is well-known to AD specialists that elders do not like intrusions into their privacy. Such intrusiveness could even accelerate the degradation of their cognitive abilities or lead the patient to reject the proposed assistive services.

The RFID antennas have a detection zone ranging from a few centimeters to several meters (even for passive tags). However, the more they are adjusted to be sensitive (long-range detection), the less they are precise. For this reason, we decided to give them a bubble of detection of approximately 1 meter (radius). With this setting, we can detect the position of multiple objects.

4.3.2 ALGORITHM IMPLEMENTATION

The implementation of our algorithm was made in the Java object-oriented programming language. We used Java, because we wanted our recognition software to be portable on different platforms and to be compatible with the most technologies as possible. The software communicates to the server SQLServer database of the smart home but have its own local MySQL database that contains the activities' library (and the spatial constraints). The figure 4.7 shows the general architecture of the software on a multi-agents perspective.

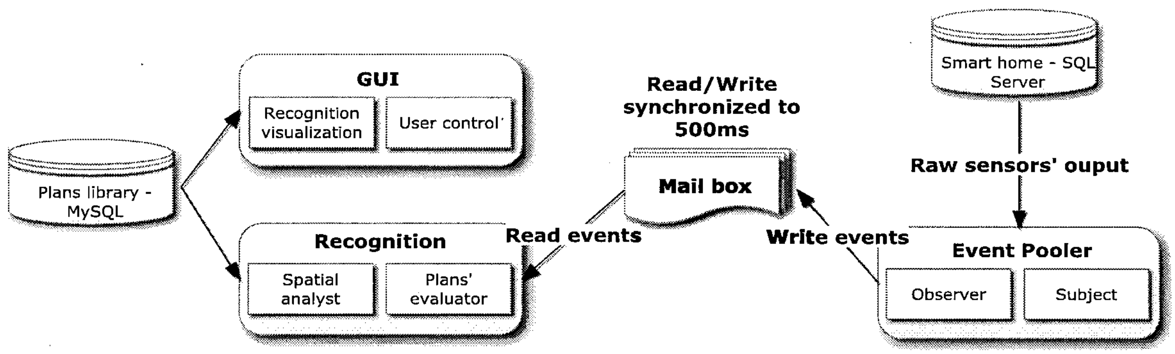


Figure 4.7: Multi-agents software general architecture

As you can see on the picture, there are mainly three components that are built as distinctive threads. We chose to build from the multi-agents paradigm [15] to have better control on the resource usage. For example, the *Event Pooler* agent must take exactly 500 milliseconds to check what has happened and create corresponding events. Most of the time it only takes less than 100 milliseconds, but if it was not limited it would probably update too often querying too much the server database. Using multi-agents approach also allows us to enhance a particular part of the algorithm easily or to change technology because each agent corresponds to a general task in the software. As you can see, the event pooler is the only one to communicate to the smart home database, and the recognition agent on its side is the only one to communicate with the plans' library. The *event pooler* agent has the task of transforming basic information into concrete events that can be used by the AI, or simply by the *recognition* agent. The two of them communicate through a simple mailbox where one write and the other read. It is important that both agents keep the same pace because events are only important for a limited time (e.g. the event "Cup moved" would not be important an hour after it happened in the smart home).

4.3.2.1 Graphical user interface

The software developed was constituted of a graphical user interface made from the Java Swing library that can be seen on figure 4.8. This was managed on a separate thread that allowed it to work during the recognition process at anytime. To conduct a recognition experiment, the user had to choose the plans he was going to execute (the recognition agent does not know it!!!), then select if he wanted a report to be generated (more about it later) and press the "Begin recognition" button. As you can see, the hypotheses (all the plans) are all visible during recognition. Then, the algorithm gives the real time plausibility of each ADL of being the ongoing activity as it is believed by the recognition agent. At the beginning, all the ADLs are equally plausible but as observations are made, the recognition agent will increase some and decrease other. The little rectangle with the "Status" label is here as an indicator. In fact, if the recognition agent correctly identifies the activity that the user selected beforehand, the status will be green; if it almost recognizes it, it will be yellow; and if it does not recognize it at all it will be red.

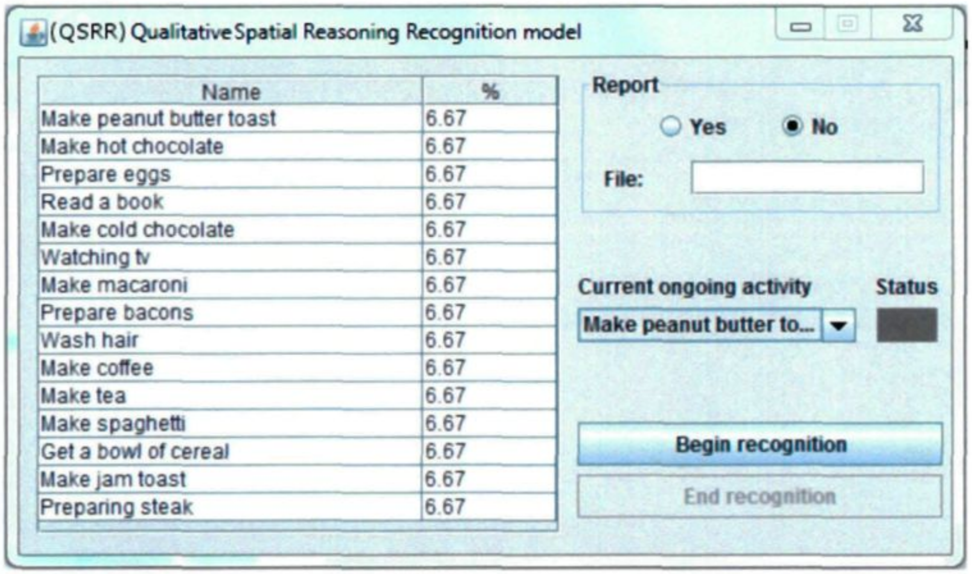


Figure 4.8: The GUI of the developed software

4.3.2.2 Foreword on calculation complexity

Recognition algorithm must be fast enough to process all the smart home information and quickly take decisions to help the resident. The software was built with the optimization of code in mind. It is absolutely mandatory for a recognition algorithm to execute iteration in polynomial time. The algorithmic complexity was kept in a polynomial time of $O(n^3)$. However, to do that, memory space (and efficiency) had to be sacrificed. Yet, memory is not an issue on a modern personal computer in normal programming situations so it was for the best.

4.3.2.3 Object positioning in space

In our implementation, we tagged every object in order to position them in the Cartesian space of the smart home. However, it is impossible to do in two dimensions when it is only detected by one antenna since antennas only give us the signal power returned by tag. When it happens, we simply suppose the object is in front of the antenna and place it in space from that same antenna. However, most of the time, in the kitchen, objects are detected by at least two antennas. Since they cover only 180 degrees, two antennas are enough to do a triangulation calculus to get the position of an object. To do so, we measure the distance of the objects from each antenna. Then, we use it as a radius to create two virtual circles around the antennas. Using the equation of both circles, we can find the coordinates (x, y) where the circles intersect. We are certain to obtain only one intersection point because the other one will be on the other 180 degrees not covered by the antennas. This process can be seen on figure 4.9.

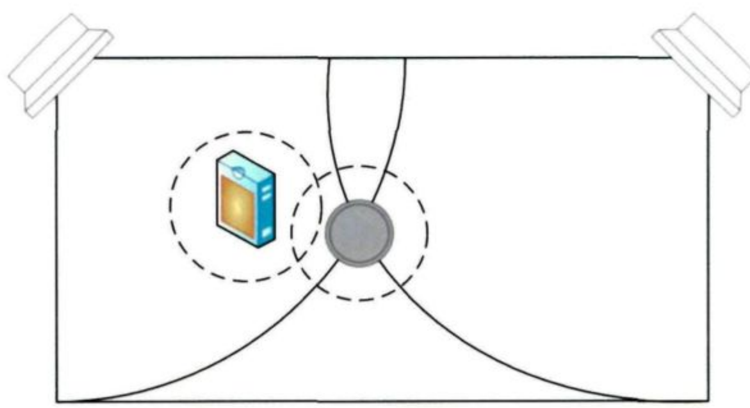


Figure 4.9: An object position being determined with triangulation from two antennas

The picture also shows partially overlap relation between objects. To reason about space, each object is associated with a virtual area. These areas were defined in 2D to reduce the complexity in order to improve the precision of the results.

4.3.2.4 Choosing the right shape for virtual areas

We tested different shapes of regions (convex hull, disc, elliptic) and we rapidly concluded that convex hull would be far too complicated to implement and not necessarily the best choice to be considered to use. Besides, elliptic shape would have been a good solution (probably the better), but as we cannot know the orientation of the objects for sure, we would not be able to determine the orientation of the axis of the ellipse in a Cartesian plane. Thus, it has to be eliminated too (most convex hulls would have required a mean to determine the orientation too). Therefore, we created the regions in the shape of a projected disc under the objects. The radius of an object's area is about the size of the diameter of that corresponding object. The elongated objects (spoons, forks, etc.) are no exceptions to this condition. We use their longest diameter as the radius. Another advantage of circle regions over other types is the calculation complexity of the spatial relation determination. We explained before how we could determine the relation simply by comparing the distance between the center points. This method would not have been sufficient for elliptic shaped regions nor for convex hull. We would have been forced to compare intersection points for other shapes that would have been lot more expensive in calculation time. As we said before, calculation complexity is an important factor to take into consideration when building recognition algorithm.

4.3.2.5 Event pooling

The event pooler agent is a very important fellow that has the hard task of merging information extracted from the smart home database into meaningful events. Events can be of many kinds. The subject may have moved from a room to another, a door might have opened or an object might have been moved. This agent shares a commons mailbox with the recognition agent. It writes events as letters that can be of different basics type and put them into the mailbox. However, we must be sure that the mailbox does not overload so both agents must execute their task in about the same speed. In fact, they are configured to perform an iteration each 500 milliseconds. If they go faster, they are put to sleep. That is one of the reasons why calculation complexity was very important in this project (and for most recognition algorithms).

Because of the RFID degree of sensibility, there is one more condition for the observation agent to create a new event for object movement. That condition is that the sensors' interpretation (not the pure output) must stay stable for at least 1 second. Then, for example, if a *cup* becomes undetected during few milliseconds, it will be ignored (situation that happens very often with RFID technology).

4.3.2.6 Report generation

We said before that the user could choose to generate a report. This functionality was implemented to facilitate the compilation of data and better understand what happen during the process of recognition. A report includes various information on the test such as the date and time of beginning, the duration, the first recognition, the total recognition time and the important events that happened. An example of report for the realization of a recognition test on the ADL *Make coffee* can be found on appendix A. A notable thing about the report module is the use of the visitor design pattern. This design pattern allows some classes to be visited by visitor classes. It is very useful to separate an algorithm from the structure on which it operates. We used it because the report generator needed information from a lot of classes. So, basically, it can gather the information by visiting the classes without ever touching to a real object. Of course, many more design patterns were used in the development of this software, but this one was notable because it is rarely exploited so it was decided that it could be interesting to have a word about it. The concrete implementation of the visitor pattern for the report generator can be seen on figure 4.10:

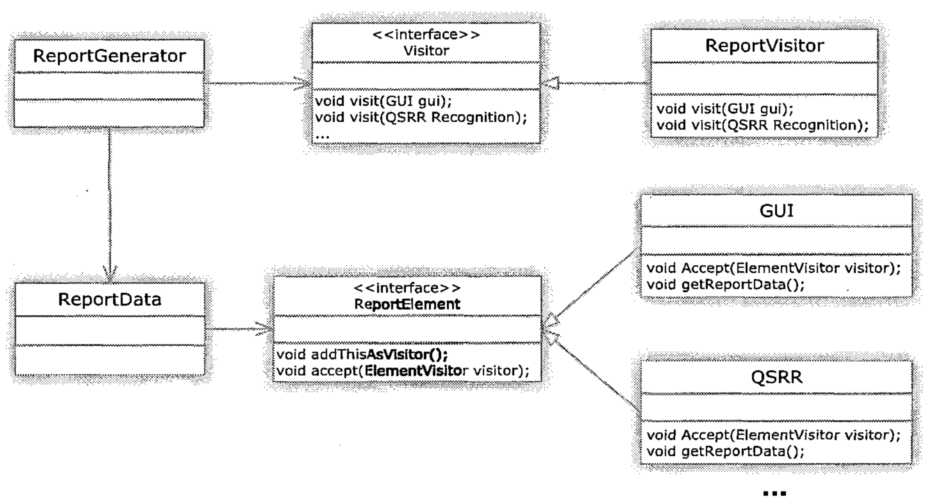


Figure 4.10: The visitor pattern implemented for the report generator

4.3.3 LIBRARY IMPLEMENTATION

The key functionality of the QSRR model we presented resides in the activities' library so it is important to talk about it a bit. We have defined 15 activities into the library and most of them cover similar kitchen activities. For instance, we have *MakeCoffee*, *MakeTea*, *MakeHotChocolate* and *MakeColdChocolate* that all share common steps such as *boiling water*, *pouring milk* or *stirring with a spoon*. We also added less traditional activities such as *Read a Book* or *Watch TV* that not much algorithms attempted to recognize. For all these activities, we created about 25 different object types in the knowledge base that were associated to real physical objects in the smart home. Constraints were defined on the object type and not on the real objects. That feature allows us to add, change or eliminate real objects in the smart home without influencing the spatial constraints defined. We also divided the smart home into virtual zones that can be seen on figure 4.11. The divisions are defined logically. So for example, A1 is the television zone and A2 the dinning zone while B3 is the bookcase zone.

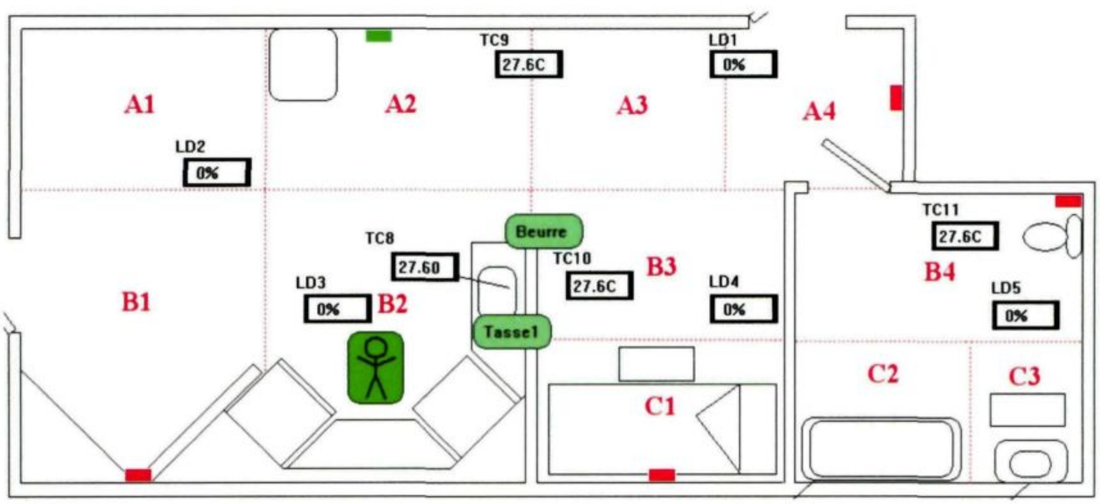


Figure 4.11: Logical zones of the LIARA smart home

These zones, represented by the letter A1 to C3, are used to define spatial constraints on an activity and an object type. For example, there is a disjoint relation with the bedroom zone (C1) for the plan *Washing Hair*, because it is fairly reasonable to assume one will never wash his hair in his bedroom. We have chosen to build the library and the constraint on plans in a MySQL relational database that was directly configured locally on the working station (not on the smart home servers). However, the structure of the library is very simple and could have been mounted on directly in the programs or in a simple text file. Moreover, we load the entire library in the memory at the launching of the recognition software. We chose to implement it in MySQL database because it was not more trouble and because if we are to improve the structure, it would be easier. Also, if the plans' library grows too big, it will become a major limitation to the *load in memory* model. Below you can see the tables of the database and their relations (figure 4.12):

Figure 4.12: Visualization of the library implemented on MySQL

4.3.2.1 Managing the library

At this step of the explanation, you might be thinking that such a library full of constraints would be long to build. The time to build the complete activities' library with all the spatial constraint was about a day. It might seem a significant time, but this is a onetime task because the library will not have to change even if we use the algorithm in another smart home. All that will need to be updated in another smart home is the concrete objects used and the defined logical zones.

Moreover, as it is shown on figure 4.13, two small software were developed in order to manage the library. The one at the left allows creating the spatial relation constraints between two objects on a plan. It also allows looking rapidly if it already exists for an activity. The software at the right of figure 4.13 was created to add constraint between objects or plans and a smart home zone. Of course, the software interfaces are a bit rough, but it is only used by the team to inputs spatial constraints during the project. However, it would not be hard to develop a good and easy to use constraints manager in the future if that type of library was to spread to concrete smart home.

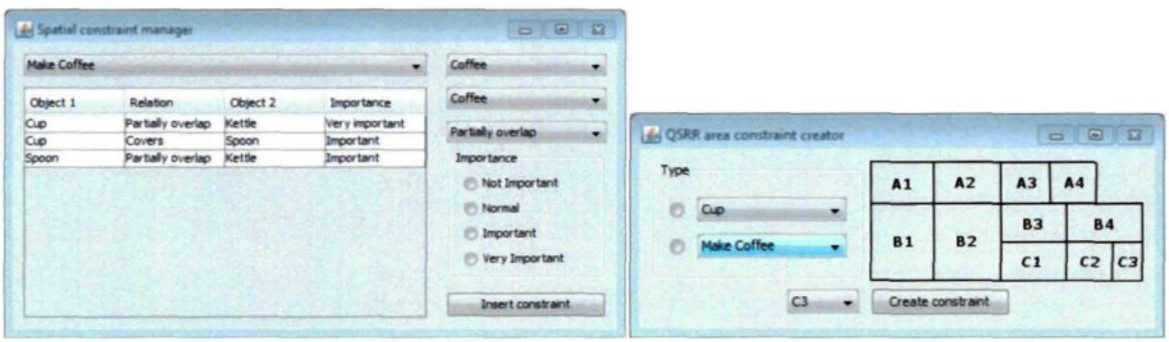


Figure 4.13: Tools to manage the library

Finally, one might be wondering what happen if constraints are forgotten to be defined in the design process. The answer is simple. It could reduce the recognition performance depending on how much constraints have been forgotten or badly designed.

Nevertheless, if most of the constraints are correctly defined, the recognition performance will remain almost unaffected. In conclusion, the library is robust, simple to design and transferable to any smart home with little adjustments.

4.4 EXPERIMENTAL PROTOCOL

In order to test our new algorithm, we had to define realistic scenarios of activity. These scenarios were based on clinical trials that were conducted with a neuropsychologist researchers and undergrad students at the LIARA laboratory [6]. These clinical studies were not conducted purposely to valid the model presented in this master thesis, but the data gathered about activities realization by AD patients was easily reusable. This section will describe the clinical trials and the tests we built to test the model.

4.4.1 THE CLINICAL TRIALS

The goal of these clinical trials was to experiments with the various types of prompt for assistance in tasks. The first step was to determine an ADL that would be carried out by the subjects. To do so, we collaborated with our colleague, a neuropsychologist researcher, which helped us determining the appropriate test. We chose to use the well establish cognitive test named the *Naturalistic Action Test* (NAT) [58]. The NAT has purposely been designed to assess patients with neurologic afflictions. The original NAT featured three activities, but since we wanted to conduct experiments with AD person we had to keep the realization time short. For that reason, we decided to focus our efforts into one activity. That activity needed to be simple to put in place and to require a little organization. We noticed that a lot of examples in the literature implied kitchen activities such as cooking [60], washing hands [11] and preparing tea or coffee [88]. We also needed at least few steps for the chosen activity and to be shorter than fifteen minutes. For these reasons and because it is a well-known activity by patients from all age and both sex, we chose to conduct experiments on the activity *MakeCoffee*.

4.4.1.1 Recruitment of the subjects

To conduct the clinical trials, we completed an ethical protocol from which we obtained two certificates of approvals. The trial was divided into two distinct phases. In the first phase, we wanted to gather useful data about the normal performance of the activity. So, for this phase, we recruited a set of normal subjects on a voluntary basis. To do so, publicity has been made in regional newspapers, as well as on the campus. The participants had to be aged between 18 and 77 years old. The only inclusion criterion was the absence of any type of cognitive impairments. The participants were also asked to refrain from any alcoholic or drug use, for the 24 hours preceding the test. A total of 20 participants were selected to perform the activity.

For the second phase, we entered in formal collaboration agreement with the Centre de santé et services sociaux (CSSS) of La Baie and the Maison Le Phare of Jonquière. They both helped us recruiting Alzheimer's patients. We obtained an adequate group of cognitively impaired subjects ranging from mild to middle stages of development of the Alzheimer's disease.

4.4.1.2 *Protocol of activity realization*

The experimental protocol for the activity realization worked as follows. The participant takes place on a chair; all the material needed for the completion of the activity was laid on the table in front of him. The participant was free to use anything he wants to complete the task and can do so in the order that pleases him. As instruction, the participants were only asked to prepare a cup of instant coffee with sugar and milk. There was no time limit, and every test was camera-recorded for further analysis. Clinical trials were all conducted into the LIARA laboratory. Normal subjects like those suffering from Alzheimer's disease came to our laboratory for the realization of the experiments. The figure 4.14 shows an Alzheimer patient during the execution of the test.

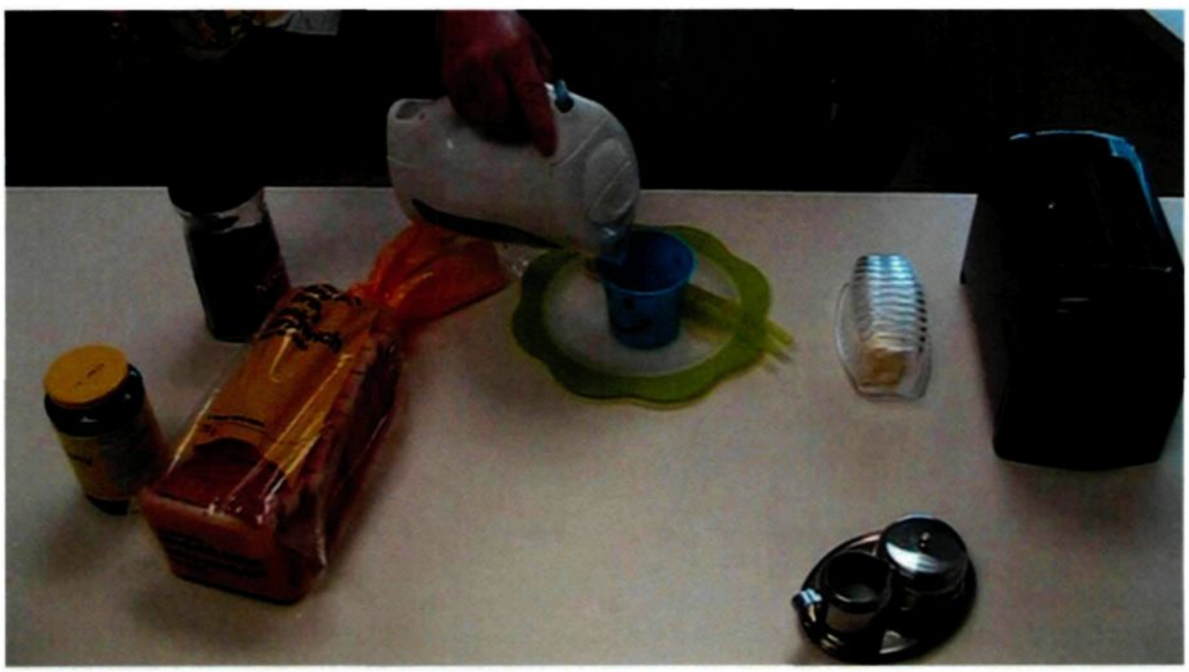


Figure 4.14: An AD patients doing the NAT activities

4.4.2 TESTS REALIZATION PROTOCOL

From the observation of a little less than fifty recorded execution sequences, we created different representative scenarios to recognize with our new algorithm. Each scenario was recorded with our smart home software three times following always the same protocol. Before recording a scenario, we replaced objects of the smart home at the same initial position that was used for all tests. The initial position of objects was chosen to follow a perfectly normal house organization. Thus, they were not set in a particular way that could help the recognition. The person doing the activity test followed a strict methodology. It always began by the person entering the smart home then going to the activity zone and beginning the activity. A complete description of all these scenarios is joined as appendix B.

4.4.3 SPATIAL ISSUES

Many scenarios included mistakes that were extracted from our experiments with AD's patients and normal subjects. For those scenarios, the problem was documented to be able to repeat it exactly when recording. We will not describe the problems in detail because we already covered it in the section 2.1.1. A total of six different scenarios included an *Absence/Presence* issue where in particularly in three of them an object was missing and in the three other an object was replaced by another one. Three others scenarios included a *grouping* issue and two more included an *object position* issue. Finally, even though it is not really a problem for activity realization, five scenarios exploited the subject position issue in order to gauge the algorithm's performances. See appendix B for more detail on the problems introduced and on scenarios.

4.5 RESULTS PRESENTATION

We have conducted rigorous experiments over a week implying different types of activities to assess the new algorithm we developed. All scenarios have been tested three times following the experiment protocol described in the earlier section. We did a total of 78 tests on all 26 different scenarios inside our new smart home. From those scenarios, there were a total of five distinctive activities, which we wanted our algorithm to recognize: *Make coffee*, *Make tea*, *Make hot chocolate*, *Read a book* and *Prepare a cold bowl of cereal*. Of this number, we more extensively tested *Make coffee* and *Make tea* in part because of the clinical trials that were conducted on the activity *Make coffee*. A total of 33 tests had a spatial problem, and 45 were normal execution of ADL. The total duration of the combined tests was about 285 minutes. This section will discuss about the result we obtained from this period of experiments and compare it with other existing works.

4.5.1 METRICS DEFINITION

Before presenting the results, we have to define a few metrics for the comparison purpose.

Recognition rate (RR): The most popular metrics that is used for evaluation of recognition approaches. This metrics measure a unique at end conclusion of a recognition algorithm.

So, most of the time it is calculated by executing the complete activity sequence and then asking the algorithm to infer what was the activity. For example, if I make coffee and at the end I ask the algorithm to infer what I did and it concludes that I made coffee it will be a 100% recognition rate (1/1). If I make coffee ten times, ask the algorithm to infer the activity each time at the end of the realization and out of the 10 time it infers 9 times that the activity was *Make* coffee, then its recognition rate would be 90% (9/10). However, some authors use it as what we call in this master thesis the *correct duration rate*.

Correct duration rate (CDR): That metric represent the total duration while the algorithm was able to correctly identify the ongoing activity during the total realization time. That means that at each time *t* of an execution sequence we ask the algorithm to infer the ongoing activity. Because the ADLs do not last the same duration, it is given as a percentage of the complete activity sequence. (e.g. Activity A is recognized 1 minute out of 2 minutes duration length, that would give a 50% CDR). Of course, we can cumulate it from numerous activities. For instance, if activity B was recognized 1 minute 30 out of the total duration of 2 minutes, it gives a CDR of 75%. Then, we could evaluate the mean CDR for the algorithm as CDR of activity A+B/2 = 50%+75%/2 = 66,7%.

Almost correct duration rate (ACDR): Duration while the algorithm was not able to clearly determine the correct ongoing activity but could identify 2 or 3 activities as equally plausible, including the current ongoing activity. That metric is complementary to the CDR and often the hypotheses that the recognition algorithm cannot discriminate are really similar ADLs (e.g. *Make Coffee & Make* Tea). It is also presented as a percentage of the complete duration of the activity sequence.

Early detection rate (EDR): This metric measure how fast is an algorithm in determining the correct ongoing activity. It is calculated by taking the time at which it correctly identified the ADL for the first time and dividing it by the total duration. Consequently, it is given as a percentage and the lower it is the better it is. For example, if an algorithm infers the correct activity after 20 seconds of the realization sequence totalizing 1 minute 40, the EDR will be 20/100 = 20%.

4.5.2 THE RECOGNITION RATE (RR)

First of all, we must establish a certain fact about the recognition success to adequately compare with other recognition approaches. Probabilistic methods such as Dynamic Bayesian Network (DBN) or different types of Hidden Markov Machine (HMM) are often the bringer of the better scores in the matter of recognition rate. Pure probabilistic like those give scores near the 100% roof. However, as we discussed in the related works section, they suffer from other weaknesses, such that they are hard to update and the calculation complexity is high. In comparison, if we had used the recognition rate metric to evaluate our QSR model, we would have had to calculate the single at end conclusion of our algorithm of which plan would be more plausible. In that way, for our 78 tests, we would have obtained the incredible result of 100% recognition rate. Of course that does not mean our algorithm is perfect. To be fair, on a bigger library with a lot of spatially similar activities or with a badly constructed library, the algorithm could have ended up wrong quite a few times. For example, if we had inserted 20 activities to make different kinds of spaghetti but in a really similar way, the algorithm would not have differentiated between all these activities easily. However, since the premise is that the activities are alike, we can infer that the type of errors the resident will make will be also similar for all these activities. Thus, perfect recognition is not needed to provide coherent assistance.

Nearly all experiments that we can find in the literature on activity recognition use the recognition rate metric to evaluate the performance of their approach. That is because most often their algorithm analyzes the complete activity sequence to determine what has happened only at the end. We believed that it is not appropriate since the goal is to be able to identify an *ongoing* activity directly during its execution. That is why we prefer to use the *correct duration rate*, the *almost CDR* and the *early detection rate* metrics.

4.5.3 Analysis of the CDR results

The first thing to notice about the results is the *correct duration rate* that can be seen on figure 4.15. The figure shows that all activities were recognized for more than 60% of their execution time. Remember what we said about the experimental protocol in section 4.4.2, for each scenario realization, the person had to enter the smart home from the door at the beginning of every test. This means that for an activity, there will be about 10 seconds delay before anything occurs in the smart home. And even after that, for some activities not much happens at the beginning (in a scenario of *Make coffee* we did wait until water was boiling before doing anything else). That is quite encouraging for the CDR of the new algorithm.

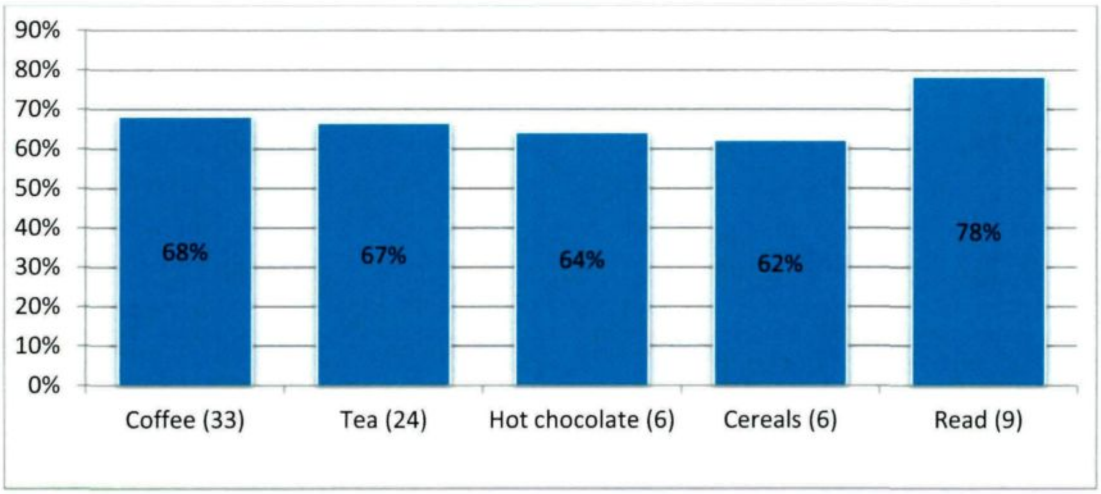


Figure 4.15: This picture shows the *correct duration rate* for each activity

Not surprisingly, the activity *Read a book* gave the best score with our recognition algorithm (we did read for about 1 or 2 minutes in each scenario). This can be explained by the simplicity of this activity that does not require many steps and also considering that no other activity is alike to this one. However, as highlighted in [65], as simple as it is for a recognition platform based on RFID sensors, it might be tricky to figure out this activity with a camera based framework. In fact, the issue of recognizing this activity from an image stream under a broad range of orientations, positions, and lightning conditions is far away from the capabilities of computing vision. The first scenario on this activity was to take a book from the shelf and read it on the bed. The second was to read it on the couch. In the last scenario, the subject was reading on the couch for some time then moving on his bed to continue his reading. We did this activity in different zones of the smart home to test and demonstrate the flexibility of our recognition model as part of the *subject position* issue. Subsequently, our QSRR model not only gives good performance, but allows recognizing the same activity in different part of a smart house without adjustment. The only thing required for such recognition into another part of the resident's home is to do not add a position constraint on the book for this zone. That is because if an object is forbidden in a zone, it will be ignored of the recognition process.

Of course, since reading a book is a very simple activity, we also tested other activities in different zones. We made coffee in every part of the kitchen counter, and the differences in the results were insignificant. We also tried different zones for the activities *Preparing a bowl of cold cereals* and *Preparing hot chocolate*. For each of them, there was at least one scenario on the kitchen counter and another on the kitchen table. Again, there was not any significant difference in the results.

4.5.4 Discussion on the ACDR results

We purposely defined similar scenarios to see if our approach could be used as a standalone recognition algorithm in a real-life context. The four activities that were very similar were *MakeCoffee, MakeTea, MakeHotChocolate* and *MakeColdChocolate* (the later one was more similar to *MakeHotChocolate*). The results showed that when the algorithm was not able to recognize one of the three activities for which we conducted tests, about 8-10% of that time he believed that these activities were equally probable. That is the *almost correct duration rate* metric that we described earlier, and that could be seen on figure 4.16

(in red). These results mean that when the algorithm was uncertain, most of the time it was on track. As we said many time, perfect recognition is not always necessary to provide adequate assistance. For example, if the ongoing activity was *MakeCoffee* but the algorithm believed that it was *MakeTea* and the person forgot to boil the water before pouring it into a cup, it would have been possible either way to provide the adequate assistive prompting.

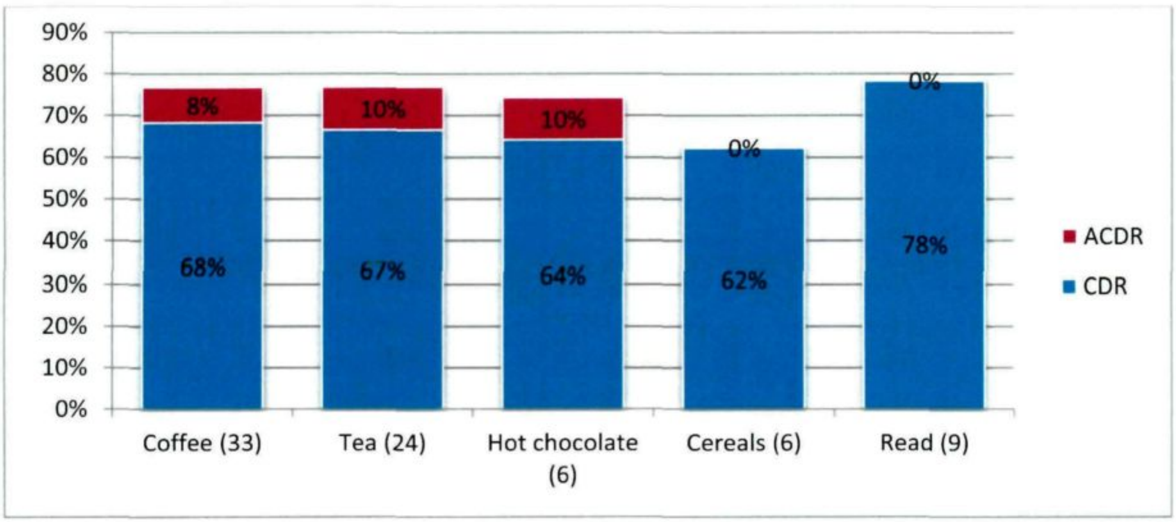


Figure 4.16: The CDR in blue and the ACDR in red

We believe that even though we got promising results that our algorithm could benefit from integrating other criterions. For instance, our longest scenario took about 7 minutes to realize. In this scenario, the subject was making a coffee without any mistake, and so our algorithm gave good results (about 87% CDR and 4% ACDR). However, even if this activity was spatially correct, taking that much time to make coffee might reflect an abnormality. Thus, we believe that our future recognition algorithms should also deal with temporal aspects.

4.5.5 Early detection rate

To achieve the goal of assisting AD's patients inside a smart home, an important thing to consider is the speed at which a recognition algorithm can identify what is going on. That is measured by the metric we explained earlier named the *early detection rate*. Some others work used this really important metric such as Nguyent & al. in their paper [89]. The early detection rate (EDR) can be defined as the percent of activity execution time elapsed before the first correct recognition of the ongoing ADL. The figure 4.17 shows the EDR for all tests, for normal tests and for erroneous one.

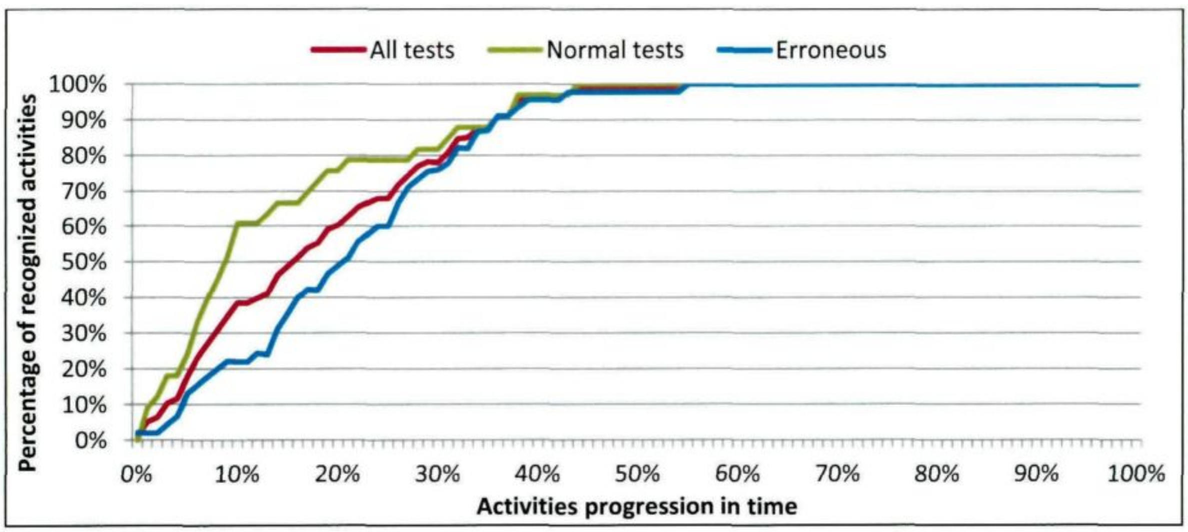


Figure 4.17: EDR in percentage of activity recognized compared to progression

The vertical axis represents the percentage of activities that were recognized for the first time (the number of activity recognized divided by the total number of activities). The horizontal axis represents the progression of the tests. It is written as a percentage to establish a common metric for all tests (they did not take the same time). The first thing to notice is how fast the algorithm is. At 30% of the tests' progressions about 80% of all activities were identified. That is rather excellent considering that many scenarios begin very slowly. At 35% of progression of the tests more than 90% of the ongoing activities have been correctly identified and 100% after 55% of progression. Furthermore, this graphic show that the erroneous tests' EDR was really similar to the normal tests. That means our algorithm is robust enough for the spatial issues we wanted him to deal with.

However, EDR metric has some flaws that we must take into account to accurately understand the performance shown in the graphic. The most important one can be found directly in the definition of the metric. In fact, that metric is calculated from the first correct recognition in a test. The problem is that it might have been only a kind of luck for the algorithm to conclude a fraction of time on the correct ongoing ADL. We can see from the variation of the EDR that ranged from 0.4% (best) to 54.5% (poor). It is clear that recognition before 1 or 2 percent of advancement of an activity is due to a factor or luck. It does not mean that the EDR is not a good metric. In fact, the overall EDR is a good indicator when a large enough set of experiments are conducted (we cannot always be lucky!). In conclusion, it might not be the most precise metric for evaluation of performance, but it is the most representative to evaluate algorithm speed according to the specific needs related to cognitive assistance.

4.5.6 ERRONEOUS SCENARIOS

After having looked at the results of our experiments, it is worthy to talk a bit about the precise results of the erroneous tests that we conducted. Nine tests from 3 different scenarios of *Make coffee* were including organizational error, which is doing the accurate steps using wrong tools (common cognitive errors for AD patients). Tool's replacements were using a glass instead of a cup, using a fork rather than a spoon and using both replacements in the third scenario. Only the third scenario gave disappointing results with around 36% CDR. However, his ACDR was around 18%. Other two scenarios were about the same as normal execution of the activity *MakeCoffee*. That is due to the fact that the spatial relation of distance between objects gave good enough hint to the algorithm to conclude that it was the wrong object in the correct execution sequence of making a coffee.

Another set of nine tests was conducted from 3 different scenarios, including a realization error. That error was simply to skip one step in the standard execution of *Make coffee*. The three forgotten steps were: adding sugar, filling milk, adding coffee. Surprisingly, even the scenarios where adding coffee was skipped gave us good results. The overall CDR was 65% with a ACDR of 11%, and the EDR was about 19%. The reason is that we did not require having coffee in the activity zone to discriminate between the similar plans. With all the other objects present, it was different enough from other activity hypotheses. That clearly shows the additive value of using the distance criterion.

With the activity *Make tea*, we tested several scenarios that would have probably led to errors without considering the spatial aspects. In the first one, the coffee was taken from the cabinet and then put back to be replaced by the tea. With the QSRR, it was easy to identify the mistake and recognize the plan. In the second kind, the coffee was taken, but forgotten on the kitchen counter. The QSRR was able to recognize *Make tea* because the relations between the objects of this plan were *stronger* than the relations of *Make coffee*.

Finally, the last kind of problem was the intrusion of a disturbing object in the realization zone of an activity. For instance, in a scenario, the subject when to read during the time waiting for the water to boil. Then he returned in the kitchen with the book that he put on the counter. Not only, our algorithm successfully recognized *Make tea*, but he was also able to recognize *Read a book* in between the steps of *Make tea*. Thereafter, he successfully ignored the book lying beside the kettle. That last example show that it is even possible to recognize simple interleaved activities with the inclusion of spatial criterions in recognition approaches. Of course, much work remains to say that interleaved activities cause no more problem. Luckily, elder and more specifically people with cognitive disease tend to realize only one activity at the time reducing the challenge of the interleaved activity recognition.

4.5.7 EXPERIMENTS DIFFICULTIES

Before concluding on the experimental part of this chapter, it could be useful to review the problems we encountered during the tests. We decided to document the problems here to act as a guide for others that would like to conduct similar experiments. First, the positioning of the subject into our smart home was not always precise due to the limitation of inference from infrared motion sensors. Therefore, sometime the algorithm believed the person was somewhere else than in his activity zone (even though we put some restriction on the position inference process). This led in a decrease of CDR and occasionally in a decrease of overall certainty of the algorithm. However, there are many solutions for better precision in the patient positioning inside a smart home such as [90, 91].

Secondly, during the first tests trying to recognize the activity *Make coffee* we observed two times the algorithm recognizing *Make spaghetti* instead of the common result we expected. We discovered that it was due to the detection of the pasta in equal topological relation with the spaghetti sauce. This misled the algorithm to add a lot of plausibility to the wrong activity. These two objects were always positioned in the kitchen cabinet side by side so sometime they were seen by our RFID antenna and then believed to be on the kitchen counter instead of on their correct position (in the cabinet). This problem, as hazardous as it may seem, can be solved straightforwardly by adding a shield on (or inside) the cabinet. The shield can be a simple aluminum sheet, and it should be more than enough to block undesired antennas activity.

4.6 SUMMARY AND DISCUSSION

This section has the goal of resuming the main characteristics of the new algorithm. Table 1 below present the most important information from the experiments conducted:

Table 4.1. Key statistics of the experiment

	Total	Normal	Erroneous
Total testing duration	285.17 minutes	161.17 minutes	124 minutes
Recognition rate (RR)	100 %	100 %	100 %
Correct duration rate (CDR)	68 %	70 %	65 %
Almost CDR (ACDR)	8 %	8 %	7 %
Best CDR	90 %	90 %	90 %
Poor CDR	31 %	45 %	31 %

Early detection rate (EDR)	17 %	15 %	19 %
Best EDR	0.4 %	0.5 %	0.4 %
Poor EDR	54.5 %	31 %	54.5 %

We will not describe further the table, since we already talked a lot about these metrics. However, it could be interesting to compare the main characteristics of our new model with the most important related works. The table 4.2 below was made up for that purpose. The CDR and EDR are not part of the table because it was impossible to find information on the four compared algorithms for these metrics.

Table 4.2: Characteristics comparison of important approaches

	Kautz[28]	Jakkula & Cook [29]	Augusto[22]	Riedel[23]	Our QSRR
Type	Logic	Temporal	Spatial	Spatial	Spatial
Machine learning	No	Yes	No	Yes	No
Erroneous recognition	No	Yes	No	Tolerate	Yes
Spatial issue	N/A	No	N/A	N/A	Partially
Interleaved recognition	No	Partially	No	Partially	Partially
Recognition level	Very High	Low	High	Low	Low
Correctness of the library	Critical	Trivial	Very Important	Trivial	Important
Library construction	Complex/ Short	Simple/ Very long	Complex/ Short	Simple/ Moderate	Moderate/ Moderate
Library reusability	Complete	Low	Complete	Impossible	High
Tested in a smart home	No	Yes	Yes	Yes	Yes
Activity complexity	Complex	Moderate	Simple	Simple	Complex
Recognition rate (RR)	100 %	65 %	100 %	99.75 %	100 %

The first thing to notice is the color code that was used to denote the effects of a feature on the qualities of an algorithm from excellent and good to moderate and bad. The erroneous recognition is a criterion that tells if the algorithm can deal with erroneous plans. Riedel's algorithm does not, but it is not critical for the activity realization to be correct. The recognition level compares the granularity of the basic input of the algorithm. For example, Kautz's work is a high level recognition approach since it recognizes from high level action such as *Make Noodle*. The correctness of the library criterion evaluates the importance of having a complete and correct library (without any mistake) for an algorithm. Of course, for Kautz, this is critical. For machine learning algorithm, it is not important. Our algorithm can perform very well even though the library is not complete and correct, but some types of error can have dramatic effects on the recognition of a particular activity. The library construction compares the difficulty and the duration of the construction process of the algorithms' library. Some algorithms, like Kautz's and Augusto's, have complex library construction but short in time duration (once the information is gathered). Machine learning approaches possess easy to build a library but require a lot of training to be performing well. As we demonstrated in the implementation section, our library is moderately easy to build and require only moderate time. The library reusability evaluates how easy it would but to reuse the library from smart home to smart home. Our library is not completely transferable because it must be little adjusted to the smart home hardware. However, it is much easier than Riedel and Jakkula & Cook approaches. Next, the activity complexity is how hard to recognize the ADLs are (granularity of ADLs). For example, some approaches recognize *PrepareMeal*, other *PreparePasta* and some

MakeSpaghettiPesto that could all represent the same activity on different a granular level. Finally, the recognition rate is very high for most approaches. Jakkula & Cook recognition rate might seem lower, but it can be explained by the fact they used only temporal relation for recognition and because their algorithm lacked enough training data.

4.7 CHAPTER CONCLUSION

This chapter had the objective of presenting a new qualitative spatial reasoning recognition model arising from the formal topological QSR framework of Egenhofer & Franzosa [25]. In the first part of the chapter, we described a theoretical model that used the relation extracted from the QSR framework. In particularly, we explained how we would exploit the distance and the position aspect in the model for the recognition of ADLs adds the identification of potential problems. We also demonstrated how the model works by providing a complete execution example from a scenario.

The second part of this chapter described the implementation of the model at the LIARA smart home. It described the infrastructure we used and the various sensors inside the apartment. It also reviewed the usage of the information and the choice of technology. Then it described the software developed and how we use it. We passed through the most important modules in order to have a global idea of the work that was done. This section described, in particular, the software architecture, its implementation details, the encoding of the plans' library and the tools used for activities building.

In the third section, we described the clinical trials, conducted by the LIARA's team, which served as a basis to define our representative test's scenarios. The section also described the experimental protocol that was used to validate our approach and evaluate its performance.

Finally, the last section presented the results we obtained. To do so we defined four metrics: the *recognition rate* (RR), the *correct duration rate* (CDR), the *almost correct duration rate* (ACDR) and the *early detection rate* (EDR). For each of these metrics, we described our results in different angles. In particularly, we have shown that the RR was not an adequate metric to evaluate recognition algorithms for cognitive assistance. The section allowed us to see that our algorithm provides very good performance in a realistic context, and it stressed how using spatial aspect in recognition approach was important. The next and the last chapter will conclude on the project by describing our approach drawback and future works that should be done in that precise field of research.

CHAPTER 5

GENERAL CONCLUSION

The master's thesis research project presented through the fourth previous chapters brings new possible solutions to the problem described in the introduction. The main intention was to investigate the spatial properties that could take part of the complex process of recognizing the ongoing activity of daily living of a person with cognitive impairment. Through the chapter 2, we were able to assess the importance of taking into account important spatial information (position, distance, orientation) for recognition of ADLs and for identification of error made by the AD patients in some specific situations (Absence/Presence issue, Object positioning, Grouping problem, etc.).

We also saw in chapter 2 that many formal frameworks provide good hints on the way we should deal with spatial aspects in general. However, these frameworks were not designed to address the specific issue of recognition of ADLs from various sensors in smart home context. Afterward, in chapter 3, we reviewed the various existing models that tried to address this problem and in particularly the one that integrates spatial aspects in their inference process. We found that the approaches that deal with the fundamental spatial aspects incorporate it in a very limited way, and thereby plenty of important information is lost. In order to find answers to this problem and to the limitations of the reviewed previous models, we exploited the eight basic relations that exist between two spatial entities from Egenhofer & Franzosa [25]. From their framework, we developed a formal model that is composed of inference methods and a plans' library incorporating spatial constraints between logical entities of many types (objects, area, etc.). In particular, this new structure of spatial constraints allows recognizing the ongoing ADL in situations that could not have been otherwise. The new model allows increasing the recognition performance in comparison to earlier works while improving the detection of mistakes committed by the resident due to his cognitive impairment. This project was conducted following a strict methodology of research. For each phase, we established precise objectives that we will review in the next section.

5.1 REALIZATION OF THE OBJECTIVES

The first objective had the goal of acquiring specialized knowledge of the field of study surrounding the problematic topic of this master thesis. It was realized in a first time by accomplishing a review of the important classical existing works in activity recognition (Carberry [8], Kautz [28], Bouchard & al. [12] and Patterson & al. [10]), those including temporal aspects (Wobcke [74] and Jakkula & Cook [29]) and more particularly, those dealing with the spatial aspects (Augusto [22], Riedel [23], Lymberopoulos [24]). Meanwhile, an important study of the tools that exists in AI to conceptualize and reason with space as also been conducted. Notably, formal frameworks from Cohn & al. [31], Egenhofer & Franzosa[25] and Guesgen [43] have been explored in order to better understand the stake of spatial aspects in recognition models. That literature review allowed finding possible solutions that led us to the contribution of this master thesis.

The second objective consisted in an extension of a formal framework of qualitative spatial reasoning and to its exploitation inside a new recognition model in order to partially solve the spatial problem explained in the introduction chapter. From the literature review, we chose to adopt the topological framework of Egenhofer & Franzosa [25]. This model was the most powerful, and its formal definition based on mathematic topology allowed to easily extend it. Specifically, this phase gave birth to the theoretical definition of a new extended recognition model allowing anomalies detection related to the distance and position relations of objects, resident and area of the smart home. Consequently, it answers to the issues raised.

The third objective aimed to implement this new model in a concrete environment as a real functioning instance of a spatial recognition model. This was done by developing new software programmed in Java object-oriented programming language coupled to a plans library built on a MySQL database. That software was interfaced with the cutting edge smart home infrastructure of the LIARA laboratory of the UQAC. In that phase, we exploited raw data from various sensors such as radio-frequency identification tags, infrared motion sensors, electromagnetic contacts, etc. The developed software implemented a multi-agents architecture that allowed creating a recognition agent implementing the theoretical model and taking basic observations as input from another agent interpreting the pure information from sensors.

Finally, the last objective of the project consisted to validate the implemented model of spatial activity recognition. The purpose was also to evaluate its ability to deal with the spatial issues described in chapter 2. For this purpose, data extracted from clinical trials, conducted in parallel by the team with normal and Alzheimer subjects, was used to define a set of scenarios of realization sequences from different ADLs. A total of 78 execution tests were conducted on the 26 different scenarios defined, and 33 of these tests incorporated one of the spatial problems described. The results were analyzed and compared with other approaches to complete this objective.

5.2 Review of the developed model

In summary, our approach proposed to integrate knowledge extracted from the topological model of Egenhofer & Franzosa [25]. That information was primarily incorporated as new discriminative information in plans' library. From that point, we defined an inference model to reason with that information and use it to formulate hypotheses about the ongoing ADLs in the smart home.

We thoroughly tested the implemented algorithm and measured its performance. We obtained a recognition rate of 100% meaning that for all 78 tests, the algorithm was able to conclude at the correct ongoing activity. However, many other approaches achieve a perfect recognition rate, so we defined new metrics to evaluate more accurately the performance of the algorithm. The first one was the *correct duration rate* (CDR) that instead of measuring the success of an algorithm with its conclusion at end of the activity sequence, it measures the success rate of the conclusion of the algorithm at each time t of an activity sequence. That is the metric of the *ongoing* recognition rate. We saw that our new algorithm gave very good results being able to recognize the ongoing activity for 65% of the realization time (or about 185 of the 285 minutes of experiments).

We also defined a metric with the goal of measuring the speed at which a recognition algorithm can make a correct conclusion. That metric called the *early detection rate* (EDR) measure the time of the first recognition divided by the total activity sequence (e.g. first time we recognize *MakeCoffee* at 12 seconds of a 2 min activity would be 12/120=10% EDR). This metric is very important because we not only want an algorithm that offers good performance, but one that could provide prompt assistance when problems arise. Again, our algorithm gave promising results and was able to firstly recognize more than 80% of activity before 30% of time elapsed in the tests' sequences.

5.3 LIMITATIONS AND FUTURE WORKS

Despite the promising results obtained during the experimental phase by testing various scenarios drawn from the clinical trials conducted in parallel by the team, the qualitative spatial reasoning recognition proposed in this paper contains some limitations, which were identified during the stages of validation. The first notable example is from the spatial criterions we enunciated at the beginning of this master thesis. That is our model does not take into account the fundamental orientation aspect that could play a significant role in both activity recognition and mistakes identification. For instance, if a resident open the tap of the kitchen for a time and move a glass under, we cannot know if he wants to fill it with water to drink (stable orientation, opening up), or if he is beginning to wash his dishes (orientation varies a lot in time). That limitation is primarily due to the sensors we used that were not sufficient to know the orientation of the objects for sure. RFID tags could have been used for this purpose by affixing tags to each side of an object, but the precision would not have been good enough and with many objects, there would have been too much interference to obtain accurate information.

Another limitation of the model we can identify is the rigidity of the spatial relations that are defined within our model. In fact, these relations are determined in a rigid way. We take the coordinates of both objects, and we verify the distance between their centers. That means that the relation is determined by rigid condition such as, for example: *IF distance between center > radius of object1 + radius of object2 THEN disjoint*. This approach, although very interesting, poses a problem in situations where the relationship is unclear. For instance, if the distance between both object is 2.01 and the radius are both equal to 1, the relation will be considered as 100% disjoint (2.01>1+1). When we consider that the RFID tags are not precise sensors, such a small difference has no real meaning, and therefore, we should be able to consider the possibility that objects touch or partially overlap. An interesting avenue to address this issue in future developments is to investigate the side of the fuzzy logic and possibility theory, which are ways that will define spatial relations in matters of degree of truth and degree of possibility. This will reflect the inaccuracy that may arise from the implementation of the model in real smart home infrastructure. Therefore, we would not simply determine a simple static relation between two objects, but a degree of belonging to the different relations (e.g. 70% disjoint, 20% touch, 10% partially overlap).

5.4 Personal assessment on this research

In conclusion, I would like to use few last words to do a brief personal assessment of my initiation to the world of research. The journey made throughout this project was quite a hard and constant work. However, it was very rewarding, worthy of all these short nights for which I traded hours of sleep for acquisition new precious knowledge in the targeted area of expertise of spatial reasoning for activity recognition. I was able to successfully conduct this project because of its stimulating nature. As a member of a formidable multidisciplinary team, I have been lucky enough to participate in the arising of a new smart home infrastructure and to be the first to conduct experiments in it. This experience allowed me to develop important new skills such as a rigorous research methodology and communication skills. This rewarding experience also allowed me to make a modest contribution to the scientific community in my field of research that I presented at the occasion of notorious international conferences [32, 33]. After such a positive introduction to research, I only look toward pursuing doctoral studies in relation to my new field of expertise. I would love to continue to develop my knowledge and my experience in the near future.

APPENDIX A

GENERATED REPORT EXAMPLE

In section 4.3.2.5, we described a module that generates a report after a recognition test has been conducted. These reports helped us in the compilation of the results and in the definition of new metrics. Below you can see a report example from a test conducted where the ongoing activity of the resident was *Make Coffee*.

```
Trying to recognize: MAKE COFFEE     Date:19 juillet 2011
****************************************************************
Start time: 04:12:20
End time:   04:15:18     Total time: 2 min. 58 sec.
****************************************************************
First recognition:  04:13:07
Total recognition: 2 min. 7 sec. (71.0 %)
Overall confidence: 54.0 %
****************************************************************

EVENTS:

04:15:17 (ALMOST) - Make coffee (27.55)
     2- Make hot chocolate (27.55), 3- Make tea (14.11)

04:14:56 (RECOGNIZED) - Make coffee (60.59)
     2- Make tea (22.42), 3- Make hot chocolate (9.7)

04:14:53 (ALMOST) - Make coffee (47.91)
     2- Make tea (47.91), 3- Make hot chocolate (0.47)

04:13:07 (RECOGNIZED) - Make coffee (74.4)
     2- Make tea (11.69), 3- Make hot chocolate (11.69)
```

APPENDIX B

TEST SCENARIOS FOR EXPERIMENTS

In this appendix, we will describe the different scenarios that were used to conduct the experiments for this project. The scenarios are grouped under five singular activities of daily living totalizing twenty-six scenarios. All scenarios began by the resident entering in the smart home by the same door (in the lobby).

B.1 Make coffee

This ADL was the most exploited for the experiments. It supposes that the resident wants to make instant coffee, and that he normally follows these steps: 1. Boil water, 2. Get a cup, 3. Put coffee in the cup, 4. Put sugar in the cup, 5. Pour boiled water in the cup, 6. Pour milk in the cup and 7. Stir with a spoon.

Objects involved: cup, coffee, kettle, milk, sugar, and spoon.
Normal scenarios:

#1-	Best situation possible, executed directly in front of an RFID antenna on the center of the kitchen counter. With particular attention with the object position during the realization.
#2-	Directly executed in front of an RFID antenna on the center of the kitchen counter with **no** particular attention to object position during the realization.
#3-	Similar to #2, but realized on the rightmost part of the kitchen counter beside the refrigerator.
#4-	Kettle is on the left most part of the kitchen counter while the preparation of the cup is at the opposite.
#5-	Objects are quickly stored after having been used.

Erroneous object scenarios (Absence/Presence issue):

#1-	The resident wrongly replaces the cup by a simple glass to make his coffee.
#2-	The resident wrongly replaces the spoon by a fork.
#3-	The resident wrongly replaces both the cup and the spoon respectively by a glass and a fork.

Forgotten object scenarios (Absence/Presence issue):

#1-	The sugar is forgotten in the realization of the activity.
#2-	The milk is forgotten.
#3-	The coffee is forgotten.

B.2 Make tea

The steps of this activity are normally as follows: 1. Boil water, 2. Get a cup, 3.Put tea in the cup, 5. Pour hot water in the cup and 6. Stir with a spoon.

Objects involved: tea, cup, kettle and spoon.
Normal scenarios:

#1-	Normal execution on the center of the kitchen counter.
#2-	During the time the resident wait for water to boil, he goes in his room and read for 1 min before returning to complete his tea (subject position issue).
#3-	The tea is quickly stored after being used.

Scenarios with distance issues (Grouping):

#1-	Coffee is taken, but the resident store it back right after to take the tea.
#2-	Both the coffee and the tea are taken out but the coffee in farther from the other object than the tea.
#3-	Milk and chocolate powder are taken out and grouped on a part of the kitchen counter. Then the tea is taken out and grouped with the cup and the kettle.

Scenarios with wrong positioning (Object position):

#1-	Subject read while waiting for water to boil, but after he brings the book with him to complete the activity *MakeTea*.

#2-	Subject goes to the bathroom while waiting for water to boil and bring back with him a bottle of shampoo.

B.3 Read a book

Objects involved: A book.
Normal scenarios at different places (subject position):

#1-	Reading on the bed.
#2-	Reading in the living room.
#3-	Reading in the living room. Then, moving to the bedroom to read on the bed.

B.4 Make hot chocolate

The ADL *MakeHotChocolate* is similar to *MakeTea* and *MakeCoffee*. Here are the normal execution steps: 1. Boil water, 2. Get a cup, 3. Put chocolate powder in the cup, 4. Pour boiled water in the cup, 5. Pour milk in the cup and 6. Stir with a spoon.

Objects involved: Kettle, cup, chocolate powder, milk and spoon.
Normal scenarios at different places (subject position):

#1-	Preparing hot chocolate on the kitchen counter.
#2-	Preparing hot chocolate on the table in the dining room.

B.5 Preparing cold cereals

The normal execution steps for this activity are: 1. Get a bowl, 2. Fill it with cereals, 3. Pour milk in the bowl and 4. Eat with a spoon.

Objects involved: Bowl, creals, milk and spoon.
Normal scenarios at different places (subject position):

#1-	Preparing cold cereals on the kitchen counter.
#2-	Preparing cold cereals on the table in the dining room.

BIBLIOGRAPHIE

[1] U. Nations, *World population ageing 2009*: United Nations, Dept. of Economic and Social Affairs, Population Division, 2010.

[2] J. Diamond, "A report on Alzheimer disease and current research.," *Alzheimer Society of Canada,* pp. 1-26, 2006.

[3] D. Patterson, *et al.*, "Intelligent ubiquitous computing to support Alzheimer's patients: Enabling the cognitively disabled," in *In UbiCog '02: First International Workshop on Ubiquitous Computing for Cognitive Aids*, 2002.

[4] C. Ramos, *et al.*, "Ambient Intelligence: the Next Step for Artificial Intelligence," *IEEE Intelligent Systems,* vol. 23, pp. 15-18, 2008.

[5] M. J. Wooldridge, *An introduction to multiagent systems*. Chichester, England: Wiley; 2nd edition (July 7, 2009), 2009.

[6] M. Van Tassel, *et al.*, *Guidelines for Increasing Prompt Efficiency in Smart Homes According to the Resident's Profile and Task Characteristics* vol. 6719: Springer, 2011.

[7] J. Boger, *et al.*, "A decision-theoretic approach to task assistance for persons with dementia," presented at the Proceedings of the 19th international joint conference on Artificial intelligence, Edinburgh, Scotland, 2005.

[8] S. Carberry, "Techniques for Plan Recognition," *User Modeling and User-Adapted Interaction,* vol. 11, pp. 31-48, 2001.

[9] K. Z. Haigh, *et al.*, "The independent lifestyle assistant\&\#8482; (I.L.S.A.): AI lessons learned," presented at the Proceedings of the 16th conference on Innovative applications of artifical intelligence, San Jose, California, 2004.

[10] D. J. Patterson, *et al.*, "Pervasive computing in the home and community," in *Pervasive Computing in Healthcare*: CRC Press, 2007, pp. 79-103.

[11] J. Hoey, *et al.*, "Automated handwashing assistance for persons with dementia using video and a partially observable Markov decision process," *Comput. Vis. Image Underst.,* vol. 114, pp. 503-519, 2010.

[12] B. Bouchard, *et al.*, "A keyhole plan recognition model for Alzheimer's patients: First results," *Applied Artificial Intelligence,* vol. 21, pp. 623-658, 2007.

[13] P. Roy, *et al.*, "A hybrid plan recognition model for Alzheimer's patients: Interleaved-erroneous dilemma," *Web Intelli. and Agent Sys.,* vol. 7, pp. 375-397, 2009.

[14] P. Roy, *et al.*, "Challenging issues of ambient activity recognition for cognitive assistance," *Ambient Intelligence and Smart Environments,* pp. 1-25, 2010.

[15] B. Bouchard, "Un modèle de reconnaissance de plan pour les personnes atteintes de la maladie d'Alzheimer basé sur la théorie des treillis et sur un modèle d'action en logique de description," Ph.D., Université de Sherbrooke, Sherbrooke, 2006.

[16] V. R. Jakkula, *et al.*, "Knowledge Discovery in Entity Based Smart Environment Resident Data Using Temporal Relation Based Data Mining," presented at the Proceedings of the Seventh IEEE International Conference on Data Mining Workshops, 2007.

[17] J. C. Augusto and C. D. Nugent, "Smart homes can be smarter," in *Designing Smart Homes: Role of Artificial Intelligence*. vol. 4008, Berlin: Springer-Verlag Berlin, 2006, pp. 1-15.

[18] H. Laprise, *et al.*, "Creating tools and trial data sets for smart home researchers: experimenting activities of daily living with normal subjects to compare with Alzheimer's patients," in *International Conference IADIS e-Health*, Freiburg, Germany, 2010, pp. 1-8.

[19] H. Pigot, *et al.*, "The intelligent habitat and everyday life activity support," in *Simulations in Biomedicine V*. vol. 7, Z. M. Arnez, *et al.*, Eds., Southampton: Wit Press, 2003, pp. 507-516.

[20] A. G. Cohn, "The Challenge of Qualitative Spatial Reasoning," *ACM Computing Surveys,* vol. 27, pp. 323-325, 1995.

[21] P.-O. Rocher, *et al.*, "A New Platform to Easily Experiment Activity Recognition Systems Based on Passive RFID Tags: Experimentation with Data Mining Algorithms," *International Journal of Smart Homes,* pp. 1-18, 2011.

[22] J. C. Augusto, *et al.*, "Management of uncertainty and spatio-temporal aspects for monitoring and diagnosis in a Smart Home," *International Journal of Computational Intelligence Systems* vol. 1, pp. 361 - 378, 2008.

[23] D. E. Riedel, *et al.*, "Spatial Activity Recognition in a Smart Home Environment using a Chemotactic Model," presented at the International Conference on Intelligent Sensors Networks and Information Processing, 2005.

[24] D. Lymberopoulos, *et al.*, "Extracting spatiotemporal human activity patterns in assisted living using a home sensor network," presented at the Proceedings of the 1st international conference on PErvasive Technologies Related to Assistive Environments, Athens, Greece, 2008.

[25] M. J. Egenhofer and R. D. Franzosa, "POINT-SET TOPOLOGICAL SPATIAL RELATIONS," *International Journal of Geographical Information Systems,* vol. 5, pp. 161-174, 1991.

[26] A. G. Cohn, "Qualitative Spatial Representation and Reasoning Techniques," presented at the Proceedings of the 21st Annual German Conference on Artificial Intelligence: Advances in Artificial Intelligence, 1997.

[27] J. F. Allen and P. J. Hayes, "A common-sense theory of time," presented at the Proceedings of the 9th international joint conference on Artificial intelligence - Volume 1, Los Angeles, California, 1985.

[28] H. A. Kautz, "A formal theory of plan recognition and its implementation," in *Reasoning about plans*: Morgan Kaufmann Publishers Inc., 1991, pp. 69-124.

[29] V. R. Jakkula and D. J. Cook, "Enhancing Smart Home Algorithms Using Temporal Relations," in *Technology and Aging*. vol. 21, A. Mihailidis, *et al.*, Eds., Amsterdam: I O S Press, 2008, pp. 3-10.

[30] B. Gottfried, *et al.*, "Spatiotemporal Reasoning for Smart Homes," *Designing Smart Homes,* vol. 4008, pp. 16-34, 2006.

[31] A. G. Cohn, *et al.*, "Qualitative Spatial Representation and Reasoning with the Region Connection Calculus," *Geoinformatica,* vol. 1, pp. 275-316, 1997.

[32] K. Bouchard, *et al.*, "Qualitative Spatial Activity Recognition Using a Complete Platform Based on Passive RFID Tags: Experimentations and Results
Toward Useful Services for Elderly and People with Disabilities." vol. 6719, B. Abdulrazak, *et al.*, Eds.: Springer Berlin / Heidelberg, 2011, pp. 308-312.

[33] K. Bouchard, *et al.*, "A new qualitative spatial recognition model based on Egenhofer topological approach using C4.5 algorithm: experiment and results," presented at the International Conference on Ambient Systems, Networks and Technologies, Niagara Falls, Canada, 2011.

[34] A. Herskovits, "Space and the preposition in English: regularities and irregularities in a complex domain," Ph. D., Stanford University, 1982.

[35] G. Retzschmidt, "VARIOUS VIEWS ON SPATIAL PREPOSITIONS," *Ai Magazine,* vol. 9, pp. 95-105, Sum 1988.

[36] B. H. Arnold, *Intuitive concepts in elementary topology*. Mineola, N.Y.: Dover Publications, 2011.

[37] M. Skubic, *et al.*, *Spatial language for human-robot dialogs*. New-York, NY, ETATS-UNIS: Institute of Electrical and Electronics Engineers, 2004.

[38] K. D. Forbus, *et al.*, "How Qualitative Spatial Reasoning Can Improve Strategy Game AIs," *IEEE Intelligent Systems,* vol. 17, pp. 25-30, 2002.

[39] J. F. Allen, "Maintaining knowledge about temporal intervals," *Artificial Intelligence and Language Processing,* vol. 26, pp. 832-843, 1983.

[40] A. Desolneux, *et al.*, "Gestalt Theory and Computer Vision
Seeing, Thinking and Knowing." vol. 38, A. Carsetti, Ed.: Springer Netherlands, 2004, pp. 71-101.

[41] A. Mukerjee and G. Joe, "A qualitative model for space," presented at the Proceedings of the eighth National conference on Artificial intelligence - Volume 1, Boston, Massachusetts, 1990.

[42] A. Morales and G. Sciavicco, "Using Temporal Logic for Spatial Reasoning: Spatial Propositional Neighborhood Logic," presented at the Proceedings of the Thirteenth International Symposium on Temporal Representation and Reasoning, 2006.

[43] H. W. Guesgen, *Spatial reasoning based on Allen's temporal logic*: International Computer Science Institute, 1989.

[44] A. Borrmann, *et al.*, "Towards a 3D Spatial Query Language for Building Information Models," presented at the Int. Conf. of Computing and Decision Making in Civil and Building Engineering, 2006.

[45] M. J. Egenhofer and A. U. Frank, "Towards a Spatial Query Language: User Interface Considerations," presented at the Proceedings of the 14th International Conference on Very Large Data Bases, 1988.

[46] A. Herskovits, *Language and Spatial Cognition*: Cambridge University Press, 1987.

[47] D. Pullar and M. J. Egenhofer, "Towards Formal Definitions of Topological Relations Among Spatial Objects," presented at the Third Internationnal Symposium on Spatial Data Handling, Sydney, Australia, 1988.

[48] M. J. Egenhofer, "A formal definition of binary topological relationships," presented at the 3rd International Conference, FODO 1989 on Foundations of Data Organization and Algorithms, Paris, France, 1989.

[49] J. L. Kelley, *General topology*. New York: Ishi Press International, 2007.

[50] M. J. Egenhofer, *Spherical topological relations* vol. 3534. Berlin, ALLEMAGNE: Springer, 2005.

[51] N. M. Gotts, *et al.*, *A connection based approach to common-sense topological description and reasoning*. Peru, IL, ETATS-UNIS: Hegeler Institute, 1996.

[52] B. L. Clark, "Individuals and points," *Notre Dame Journal of Formal Logic,* vol. 26, pp. 61-75, 1985.

[53] B. L. Clark, *A Calculus of Individuals Based on "Connection"* vol. 22, 1981.

[54] D. Hernandez, *Qualitative Representation of Spatial Knowledge*: Springer-Verlag New York, Inc., 1994.

[55] S. Katz, *et al.*, "Studies of Illness in the Aged," *JAMA: The Journal of the American Medical Association,* vol. 185, pp. 914-919, September 21, 1963 1963.

[56] T. Giovannetti, *et al.*, "Naturalistic action impairments in dementia," *Neuropsychologia,* vol. 40, pp. 1220-1232, 2002.

[57] C. Baum and D. F. Edwards, *Cognitive performance in senile dementia of the Alzheimer's type: the Kitchen Task Assessment* vol. 47, 1993.

[58] M. F. SCHWARTZ, *et al.*, *The Naturalistic Action Test: A standardised assessment for everyday action impairment* vol. 12. Hove, ROYAUME-UNI: Psychology Press, 2002.

[59] S. Giroux, *et al.*, "The Praxis of Cognitive Assistance in Smart Homes," presented at the BMI Book, 2009.

[60] J. Modayil, *et al.*, "Improving the recognition of interleaved activities," presented at the Proceedings of the 10th international conference on Ubiquitous computing, Seoul, Korea, 2008.

[61] C. F. Schmidt, *et al.*, "The plan recognition problem: An intersection of psychology and artificial intelligence," *Artificial Intelligence,* vol. 11, pp. 45-83, 1978.

[62] Y. Wærn and R. Ramberg, "People's perception of human and computer advice," *Computers in Human Behavior,* vol. 12, pp. 17-27, 1996.

[63] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach*: Prentice Hall, 2010.

[64] M. P. Georgeff, Lansky, A. L., "Reactive Reasoning and Planning," in *AAAI-87 Proceedings*, 1987, pp. 677-682.

[65] D. J. Patterson, *et al.*, "Fine-Grained Activity Recognition by Aggregating Abstract Object Usage," presented at the Proceedings of the Ninth IEEE International Symposium on Wearable Computers, 2005.

[66] J. Boger, *et al.*, *A planning system based on Markov decision processes to guide people with dementia through activities of daily living*, 2006.

[67] R. Goldman, *et al.*, "A New Model of Plan Recognition," *Artificial Intelligence,* vol. 64, pp. 53-79, 1999.

[68] D. J. Patterson, *et al.*, "Inferring high-level behavior from low-level sensors," in *Ubicomp 2003: Ubiquitous Computing*. vol. 2864, A. K. Dey, *et al.*, Eds., Berlin: Springer-Verlag Berlin, 2003, pp. 73-89.

[69] M. Weiser, "The computer for the 21st century," *Scientific American,* vol. 265, pp. 66-75, 1991.

[70] I. H. Witten, *et al.*, *Data Mining: Practical Machine Learning Tools and Techniques*: Elsevier Science & Technology, 2011.

[71] H. Kautz and J. Allen, "Generalized plan recognition," in *National Conference on Artificial Intelligence (AAAI)*, 1986, pp. 32-37.

[72] D. Py, "Reconnaissance de plan pour l'aide à la démonstration dans un tuteur intelligent de la géométrie," Ph. D., Université de Rennes, Rennes, 1992.

[73] Nerzic, "Two methods for recognizing erroneous plans in human-machine dialogue," presented at the Proc. of National Conference on Artificial Intelligence (AAAI '96), 1996.

[74] W. Wobcke, "Two logical theories of plan recognition," *Journal of Logic and Computation,* vol. 12, pp. 371-412, Jun 2002.

[75] J. McCarthy, "Circumscription - {A} Form of Non-Monotonic Reasoning," *Artificial Intelligence Journal,* vol. 13, pp. 27-39, 1980.

[76] G. Weiss, *Multiagent Systems*: MIT Press, 2000.

[77] V. Jakkula and D. J. Cook, "Mining Sensor Data in Smart Environment for Temporal Activity Prediction," in *KDD'07*, San Jose, California, USA, 2007.

[78] V. Jakkula and D. J. Cook, "Learning Temporal Relations in Smart Home Data," in *Proceedings of the Second International Conference on Technology and Aging*, Canada, 2007.

[79] K. Gopalratnam and D. J. Cook, "Active LeZi: An Incremental Parsing Algorithm for Sequential Prediction," presented at the FLAIRS Conference, 2003.

[80] F. Song and R. Cohen, "Temporal reasoning during plan recognition," presented at the Proceedings of the ninth National conference on Artificial intelligence - Volume 1, Anaheim, California, 1991.

[81] R. Weida, *Terminological constraint network reasoning and its application to plan recognition*: Dept. of Computer Science, Columbia University, 1993.

[82] J. Yang, *et al.*, "Belief Rule-Base Inference Methodology Using the Evidential Reasoning Approach - RIMER," *In Proceedings of IEEE Transactions on Systems, Man, and Cybernetic,* pp. 266-285, 2006.

[83] J. C. Augusto and C. Nugent, "The use of temporal reasoning and management of complex events in smart homes," in *Proceedings of ECAI 2004.*, 2004, pp. 778-82.

[84] J. Adler, "Chemotaxis in Bacteria," *Annual Review of Biochemistry,* vol. 44, pp. 341-356, 1975.

[85] R. Bourret and A. Stock, "Molecular information processing: Lessons from bacterial chemotaxis," *J. Biol. Chem.,* p. R100066200, 2002.

[86] N. Nguyen, *et al.*, "Recognising and Monitoring High-Level Behaviours in Complex Spatial Environments," in *In IEEE International Conference on Computer Vision and Pattern Recognition (CVPR*, 2003, pp. 620-625.

[87] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," in *Readings in speech recognition*, W. Alex and L. Kai-Fu, Eds.: Morgan Kaufmann Publishers Inc., 1990, pp. 267-296.

[88] T. Gu, *et al.*, "An unsupervised approach to activity recognition and segmentation based on object-use fingerprints," *Data Knowl. Eng.,* vol. 69, pp. 533-544, 2010.

[89] N. T. Nguyen, *et al.*, "Learning and Detecting Activities from Movement Trajectories Using the Hierarchical Hidden Markov Models," presented at the Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 - Volume 02, 2005.

[90] J. Martinez Del Rincon, *et al.*, *Tracking human position and lower body parts using kalman and particle filters constrained by human biomechanics* vol. 41, 2011.

[91] D. Hauschildt and N. Kirchhof, "Improving indoor position estimation by combining active TDOA ultrasound and passive thermal infrared localization," pp. 94-99, 2011.