Doctor of Philosophy

Automated Hazard Identification for Construction Safety Management

February 2015

Department of Architecture & Architectural Engineering
The Graduate School
Seoul National University

Hyunsoo Kim

Abstract

Automated Construction Hazard Identification for Construction Safety Management

Hyunsoo Kim

Department of Architecture &

Architectural Engineering

The Graduate School

Seoul National University

The National Institute for Occupational Safety and Health (NIOSH) suggested several contributing factors to construction injuries and fatalities such as lack of hazard identification, lack of coordinates of activities, inexperienced workers, etc. Among these contributing factors, hazard identification is the first step of safety management. Construction safety management process mainly consists of hazard identification, risk assessment, and risk countermeasure. Although all three steps are important, identifying hazards becomes a fundamental step because unidentified hazards negate the risk assessment and the risk countermeasure process. As a result, an unidentified hazard can cause harm to construction workers who will remain unprotected until it is eliminated.

When workers start their works, they move and orient their activity areas. In this situation, a construction worker encounters a hazard during both moving and working. This study briefly classifies hazards, which a worker

can encounter during moving and working, into two following categories: 1) activity-oriented hazard; and 2) non-activity-oriented hazard. These two types of hazards have different characteristics. Therefore, different approaches are taken to identify each type of hazard. In this regard, the objective of this disseratation develops a hazard identification system that can deal with two types of hazards.

Activity-oriented hazards can be identified by using past accident cases. To retrieve similar past accident cases and apply to corresponding site, indices for generating queries are firstly selected. The indices are extracted from building information model (BIM) and project management information system (PMIS). Based on extracted indices, queries for retrieving past similar accident cases can be generated. After generating queries, search engine retrieves cases and calculates similarity. The results include time and geometric information, as well. Also, they are automatically provided to the workers by using the push system.

The non-activity-oriented hazard system identifies potentially hazardous areas (PHA) on workers' path by using the RTLS of workers and a building information model (BIM). The suggested system identifies PHA by using the deviation between the optimal route (the shortest path), which is determined by extracting nodes from BIM objects, and the real movement path of a worker, which is tracked by radio frequency identification. After this process, PHA can be divided into uncontrolled PHA and already controlled PHA. The information about uncontrolled PHA is provided to safety managers so they can establish proper safety countermeasures and manage the area.

Consequently, this study will contribute to enhance the efficiency of hazard identification of safety management by automated data processing

procedure that can eliminate iterative and repetitive process. The automated procedure can identify a hazard quickly, so that the existing time a hazard without any countermeasure diminishes. It means that the time a worker exposed to risk decreases.

Keywords: Hazard Identification, Safety Management, Information Retrieval, Automated Data Collection (ADC), Real-Time Locating System (RTLS)

Student Number: 2010-30159

Contents

List of Tables

List of Figures

Chapter 1. Introduction

Construction safety management processes consist of hazard identification, risk assessment, and risk countermeasure. Although all three steps are important, identifying hazards represents a fundamental step because unidentified hazards negate the risk assessment and the risk countermeasure processes. This means that risks cannot be assessed and countermeasures cannot be established due to the fact that the risk assessment and the risk countermeasure steps lack awareness of the hazard in the first step. As a result, an unidentified hazard can cause harm to construction workers who will remain unprotected until it is eliminated.

This chapter presents a brief overview of the context under which the research was conducted. Background information in regard to this dissertation is provided. The problem statement, research objective, and research scope are discussed. Finally, the dissertation research procedure is presented at the end of this chapter.

1.1 Research Background

As construction projects become more complex, increase in scale, and involve higher worker mobility, construction-related risks increase. In fact, a

2007 report from the Korea Occupational Safety and Health Agency (KOSHA) indicates that the second highest number of injuries and fatalities among all industries involves construction. Although the accident rate of the construction industry becomes lower, the number of fatal accidents has increased (Korea Ministry of Labor 2007, United States Department of Labor 2012). This high injury rate indicates that there is presently a need for improvement in safety management, despite the industry's previous advances in safety training and accident prevention.

Safety management in the construction industry is different from that in other industries. Performing safety management prior to the occurrence of accidents is essential but difficult, even in the construction of multiple buildings with the same structure, because site conditions change over time and with different work processes (Sacks et al. 2009). Hazard identification, in the process of safety management, provides information that supports decisions on how to control and responds to identified hazards.

Among all of the safety management activities, raising safety consciousness is the most important part of preventing workplace injury, violence and even fatalities. However, considering construction accidents that may occur by trivial mistakes or a momentary absence of management, controlling these is the key factor that determines success of safety management. A systematic approach can decrease these mistakes and cover momentary management absences. To control or manage accident

contributing factors, these factors should be first identified. They can be defined as a hazard which has a potential to harm construction workers.

Current practices for identifying hazards such as human inspection require safety managers' efforts. To identify a hazardous activity or area, safety managers search past accident cases and inspect various places in a construction site. Unfortunately, the number of safety managers is usually fixed, and it is difficult to increase the number of them excessively. In this circumstance, there is a need to introduce automated hazard identification methods that are able to support safety management tasks.

This dissertation introduces a new methodology for construction hazard identification based on information technology (IT). To identify hazards, two methods are suggested according to the origin of a hazard. By using diverse technologies such as information retrieval technology, building information model (BIM), and location tracking technology, the proposed system automatically identifies hazards.

1.2 Problem Statement

Although construction companies have devoted much attention to safety management, safety management is not yet proactive. Especially, current safety management practices are usually performed based on satisfying safety regulations and using checklists. These practices have also helped preventing

accidents, but these approaches are passive. In order to improve this situation, a proactive method that can prevent accidents and eliminate hazards effectively is needed. The following problems should be discussed to establish a proactive safety management method.

(1) Repetitive Occurrence of Accident

Despite the enormous efforts made in regards to safety, the construction industry has had a poor record of preventing accidents (Fredericks et al. 2005). One of the noticeable characteristics of construction accidents is repetitive occurrences (Abudayyeh et al. 2003). The repetitive occurrence of an accident can be defined as an event that happens multiple times in similar situations (similar worker, work, and work condition). Considering these characteristics of construction accidents, there have been efforts to apply past accident cases to safety management in construction projects.

Learning from past events is a fundamental process that helps individuals and organizations improve and not repeat past mistakes (Goh and Chua 2010). Accident cases are the strongest stimuli in planning safety management and raising worker's awareness for safety (Korea Occupational Safety and Health Agency 1999). This is due to the knowledge from past accident cases being directly related to the prevention of future accidents and workers' raised safety awareness (Chung and Jefferson 1998; Lindberg et al. 2010). In particular, as accidents in a specific activity have a high potential to reoccur in the same

scenario, accident cases associated with scheduled activities provide information which predicts risk and establishes safety countermeasures (Ko et al. 2005). Therefore, if past accident cases are analyzed and used in preventing an accident, useful information can be incorporated into the safety management process.

(2) Difficulties of Hazard Identification

To identify enormous hazards that occur during a construction project, traditional practices adopt safety managers' inspection. Safety managers check spaces and activities, and they establish safety plans to prevent accidents if they find a hazard. This inspection is performed by human observers, so that this method needs time, cost, and human resources. Moreover, hazard identification knowledge in the literature and technical documents is not systematically organized, and is mostly in a scattered and repetitive condition (Ding et al. 2012).

The hazard identification process involves time-consuming activities such as retrieving past accident cases and inspecting spaces. The dynamic changes within construction sites also make it hard for safety managers to identify the generation and extinction processes (Sacks et al. 2009). To address these difficulties for identifying hazards on a construction site, hazard identification process should be automated and systemized.

(3) Too Much Emphasis on Activity-related Hazard Identification

The construction industry faces more risk factors than any other industry because each construction site has its own individual characteristics (Scatterman et al. 2008; Hallowell and Gambatese 2009). In addition, construction accidents can be divided into those occurring on working areas and those occurring on non-working areas. The Korea Occupational Safety and Health Agency (KOSHA)'s fatal accident reports (2001-2003) indicate that approximately 20% of accidents in the construction industry occurred when workers were on non-working areas. According to the Health and Safety Executive, 23% of accidents occurred while travelling in the construction industry. The risks that workers face on the movement path (non-working space) are significantly different from the risks that workers face in working spaces; this is due to the performance of other works, the piling of risky materials, and the existence of openings. On construction sites, safety management is performed by focusing on work spaces. For this reason, the level of safety management for non-work spaces is usually lower than for work spaces. Thus, safety managers should consider hazards on both work space and non-work space.

1.3 Research Objectives and Scope

Construction safety managers want to identify a hazard before it occurs. Thus, they examine activities to be performed and drawings for detecting a potential hazard before it activates or generates. However, hazard identification requires safety managers' enourmous efforts since conditions

and environments of an on-going construction project change dynamically. The dynamic changes make identify hazards hard because it is difficult for safety managers to consider all the changes for identifying hazards. Therefore, an automated and systematic approach for identifying hazards is required. The purpose of this study is to develop an integrated construction hazard identification framework to support safety management activities.

This dissertation is concerned with the means of identifying hazards and their countermeasures in the both pre-construction stage and construction stage. For this purpose, it is convenient to divide into construction hazards into two categories. One is the activity oriented hazards as the first factor can cause harm. In many cases, workers are wounded during performing their jobs. Consequently, the accident spot belongs to the activity area. Another is non-activity oriented hazards. Construction workers travel on the route to the daily work site. When a worker moves around a construction site, he/she faces potentially hazardous objects or conditions. The suggested framework should be able to deal with hazard identification of both types of hazards. The three primary objectives of this study are as follows:

(1) To establish an automated information retrieval framework for identifying activity-oriented hazards which can provide the most related accident case and its countermeasure to the end-user (worker). A three-step approach is taken to achieve the purpose:

- Review current safety management information retrieval systems

- Find an information retrieval method that fits the purpose of this study
- Establish a safety management information retrieval framework

(2) To develop a framework that automatically identifies hazards on the movement paths through worker location tracking.

- Select a location tracking system and prove its accuracy in order to apply it in construction sites
- Develop an algorithm that identify obstacles using the deviation between workers' location logs and optimal routes
- Develop a filtering module that can select potential hazardous areas among identified obstacles by the aforementioned algorithm

(3) Validate the framework with a case study by using an existing database of accident cases.

Consequently, this study will contribute to enhance the efficiency of hazard identification of safety management by automated data processing procedure and help identify hazards. It would lighten the burden of safety managers for identifying hazards. Moreover, the automated procedure can identify a hazard quickly, so that the existing time a hazard without any countermeasure diminishes.

The primary objective of safety management in construction industry is the prevention of construction accidents. Among the steps of safety management, hazard identification is the first step and should precede other steps as hazard identification drives the subsequent safety management steps. If a hazard is not identified, then it negates the whole safety management process. Therefore, it is important to identify hazards for preventing accidents.

Hazard identification for preventing construction accidents is the main scope of this dissertation. There are various types of construction accidents such as worker injury, crane accidents, equipment failures, building collapse, etc. Among these types, this dissertation focuses on workers and their injury. There are various construction types such as building, civil, plant, etc. This dissertation only deals with building construction. Generally, a construction worker can get in an accident during performing an activity and moving for performing an activity. The suggested hazard identification method is developed considering the sequence that an accident occurs to a worker.

1.4 Dissertation Outline

This dissertation is organized into six chapters—including the instruction— and a set of appendices containing support information and the accident cases DB. The research components and procedure of each chapter are as follows, and the outline is shown in Fig. 1-1.

Following this introduction chapter, reviews on hazard identification are presented in Chapter 2.1 and 2.2. Based on classification of hazard origins, characteristics and current practices of two types of hazards are analyzed. Based on analyzed characteristics and current practices of two types of hazards, a potential method that can help to identify each type of hazards is selected.

Chapter 3 presents a method for the identification of activity-oriented hazards using accumulated past accident cases. To extract the most suitable case among enormous cases, this chapter illustrates indices selection process including extracting, classifying, weighting, as well as similarity calculation.

Chapter 4 starts with the analysis results in Chapter 2.4. Supported by this work, a conceptual model is presented and assumptions for developing algorithms are suggested and validated in order to develop a detailed non-activity oriented hazard identification framework. Then, requirements for developing non-activity oriented hazard identification framework are introduced. Among technologies implemented in this framework, radio frequency identification (RFID)—one of the primary technologies used—is analyzed, and an accuracy improvement method for workers' location is developed. Finally, a non-activity-oriented hazard identification framework is established using assumptions, requirements, and the accuracy improvement method.

In Chapter 5, the suggested hazard identification framework is validated with the use of a case study. Initially, a validation strategy for two different methods applied for identification of hazards is introduced. By following the suggested strategy, the activity and the non-activity-oriented hazard identification frameworks are validated.

Chapter 6 summarizes the study and concludes by examining the contributions of the completed research. It also includes recommendations for future continuation of this study.

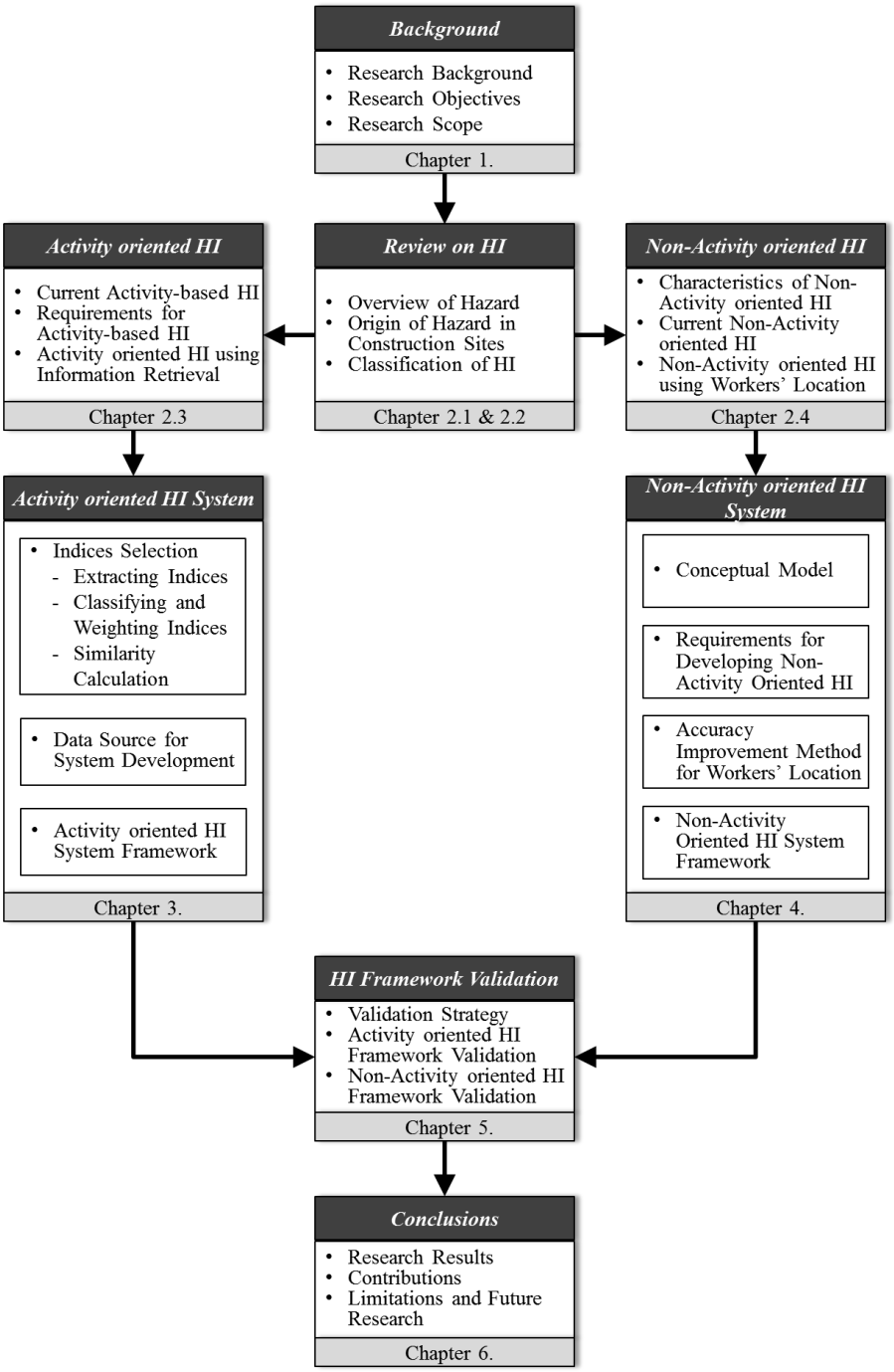


Fig. 1-1. Disseration Outline

Chapter 2. Preliminary Study

2.1 Hazard and Safety Management

All the engineers wish to make their job site safe. Since the beginning of the construction industry, safety management has been one of the main issues. However, the accident record of construction industry is still poor, despite enourmous efforts to improve construction safety. This section presents the fundamentals of hazard and analyzes hazard causes.

2.1.1 Definitions

To begin to manage hazards in construction, one must first understand the actual nature of hazards. Over several decades, considerable confusion has arisen with regard to the concepts of safety, risk, and hazard. The major source of confusion is that many people tend to interchange the words as if the terms were synonymous. To avoid conceptual confusion, definitions of the terms are investigated and defined.

(1) Safety

Safety has been defined in a variety of ways. Safety is the state of being safe, the condition of being protected from unlikely to cause, risk, or injury such as physical, social, financial, political, emotional, occupational or other types of consequences of failure, damage, error, accidents, harm or any other

event which could be considered non-desirable (Bahr 1997).

(2) Risk

Hazard addresses only the severity or end result. Risk combines the concept of severity of the accident consequence and the likelihood of it occurring. In the simplest terms, risk is the combination of the probability (or frequency of occurrence) and consequence (or severity) of a hazard (Lee et al. 2012a). Every safety managers want to make their jobsites zero-risk but that is a practical impossibility. Because we cannot totally avoid or eliminate risk, all the workers and managers try to shrink it as much as possible. This can be done by lowering either probability or the severity of the hazard, or both.

(3) Hazard

A succinct definition is that “a hazard is a condition that can cause injury or death, damage to or loss of equipment or property, or environmental harm (Roland and Moriarty 1990)”. To be more particular, a hazard is an unsafe physical condition that is always in one of three modes: Dormant (unable to cause harm), Armed (can cause harm), Active (causing injury, death, and/or damage) (MacCollum 2007). Fig. 2-1 illustrates the flow how a hazard changes. Conditions create a hazard, and a hazard exists before it is eliminated. Changes of circumstance arm the hazard within the limit of the action mode. The line between an armed hazard and an active hazard is called the hazard elimination. From this point, one miss step that is not eliminated or controlled can activate the hazard. As the result of an activated hazard, damages and/or

injuries can occur. This analysis provides the changes of circumstances that can arise and how they affect workers, activities, and conditions.

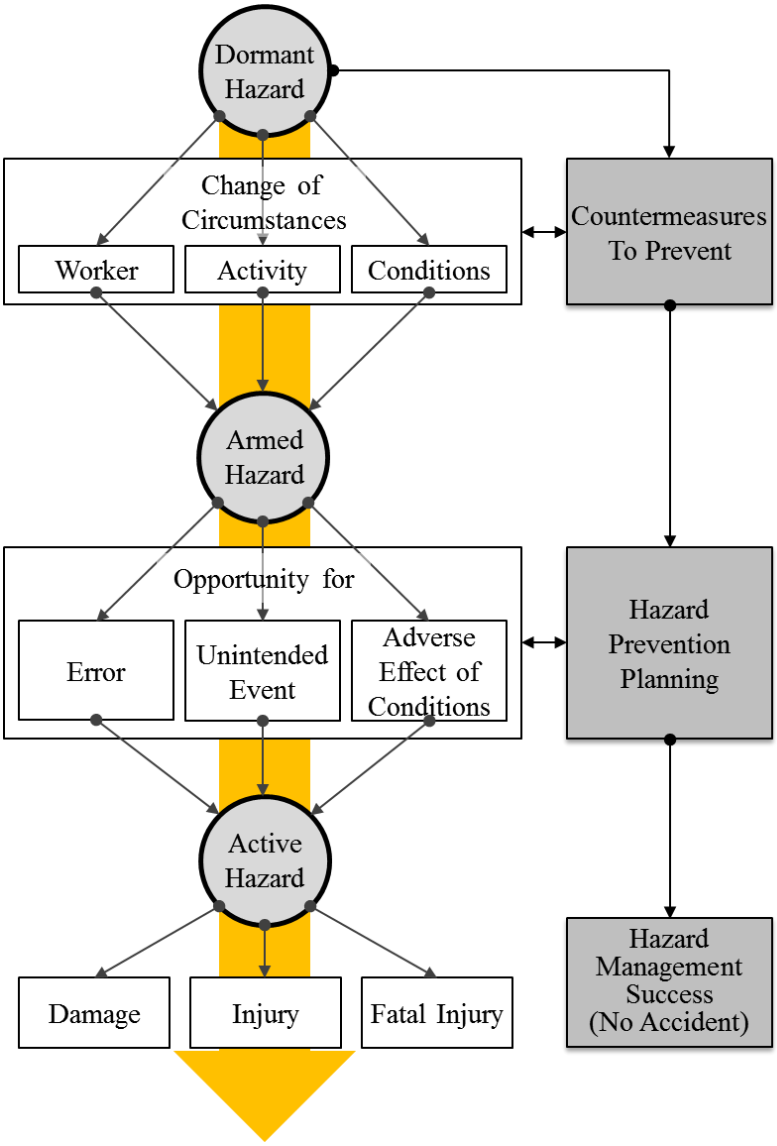


Fig. 2-1. Three Modes of Hazard (modified from Bahr 1997)

A hazard has its life cycle. As shown in Fig. 2-2, the cycle has three phases that consist of hazard occurrence, hazard existence, hazard elimination. By various reasons, a hazard occurs. For instance, an opening on the floor is made for easy equipment transportation between two adjacent floors during a construction phase. Once a hazard occurs, it exists before it is eliminated. During it exists, the hazard is neglected without any countermeasures or it is controlled or safeguarded against approaches of workers. A hazard is eliminated when the cause of a hazard perishes. For example, after an opening–whether it is safeguarded or not–is closed up by workers, the opening does not exist and it is no more hazard.

All the hazards follow this process and it gives an insight how to identify and manage hazards on construction sites. By understanding the life cycle of hazard, hazard identification can be easily performed. In the next section, how to identify hazard based on the concept of life cycle of hazard will be presented.

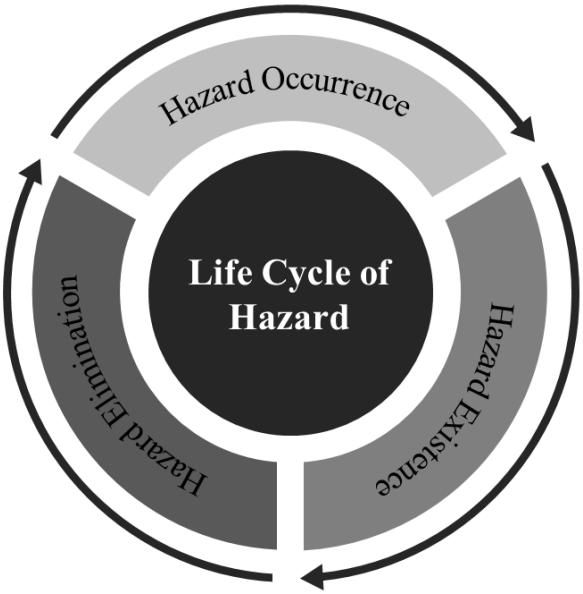


Fig. 2-2. Life Cycle of Hazard

2.1.2 Construction Accident

This subsection will briefly review the root cause of construction accidents based on Heinrich's and Bird's domino theories. Based on the review result, current construction safety management practices are analyzed.

(1) Causes of Construction Accident

One of the most widely known theories is Heinrich's Domino Theory (1941). He suggested five metaphorical dominoes labeled with accident causes. Five dominoes are Social Environment and Ancestry, Fault of Person, Unsafe Act or Mechanical or Physical Hazard (unsafe condition), Accident, and Injury. Heinrich defined each of these dominoes explicitly, and gave

advice on minimizing or eliminating their presence in the each sequence. Heinrich's Domino Theory states that accidents result from a chain of sequential events, metaphorically like a line of dominoes falling over.

Although Heinrich's Domino Theory helps to understand accident processes, this theory was revised because of emphasizing blame on individual excessively, not considering the managerial fault. Bird (1973) revised Heinrich's Domino Theory and suggested revised Domino Theory. Bird's five dominoes are Management Structure, Operation Errors, Tactical Errors, Accident, and Injure. He suggested that the root cause of an accident is "lack of management control".

Eliminating "Lack of control" situation is the most fundamental solution for preventing an accident (An 1994). However, it is impossible for construction managers to eliminate all the "lack of control" situations. Construction site conditions dramatically change overtime and there are lots of components and workers. Therefore, a method that can decrease or minimize "lack of control" situations is required.

(2) Current Construction Safety Management Practices

The purpose of safety management is to identify hazards, eliminate or control them, mitigate risks. Safety management mainly consists of six steps. At first, a manager selects an activity for setting the range or scope. Based on the range, expecting harms and hazards are identified. Identified hazards are estimated and risk of each hazard is calculated. A manager evaluates risks and

establishes countermeasures that can mitigate risks. If controlled risks are acceptable–there is obviously risk but the risk can be acceptable, then the safety plan is performed. In the opposite case, a manager develops a response development, and the risk is estimated and evaluated again until the risk becomes acceptable. Fig. 2-3 illustrates this process.

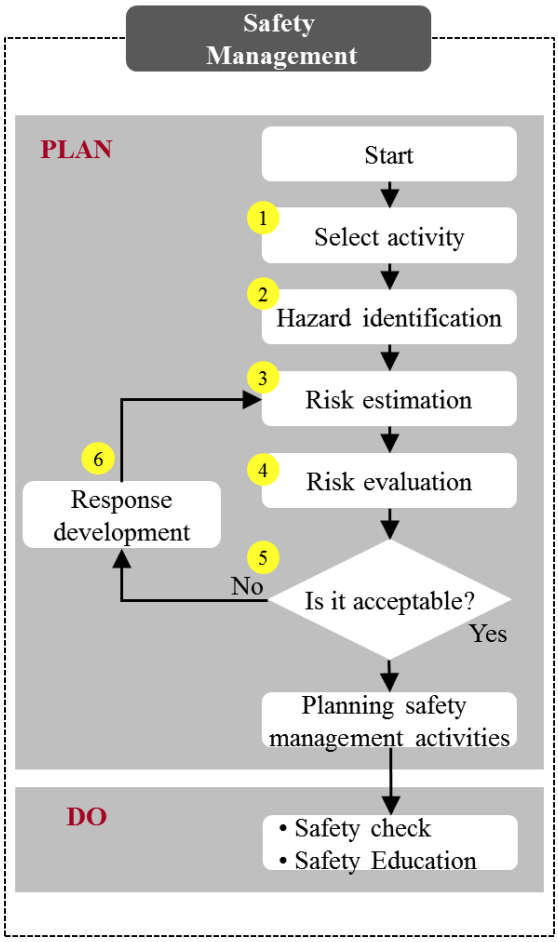


Fig. 2-3. Safety Management Process (modified from Ahn 2007)

Construction safety management practices are different from other industries because construction industry has own characteristics such as discontinuity, uncertainty, non-repetitive project, and inherent dynamic changes. Based on these characteristics, construction accident causes can be summarized as Table 2-1.

Table 2-1. Construction Accident Causes and Countermeasures

Accident Cause	Related Cause	Countermeasure
Specificity of the working environment	• High environment effect due to outdoor construction • Difficulty of predicting risk	• Use of safety information that can respond effectively to frequent changes in working conditions* • Establish and enforce strict safety plan*
Risk of work itself	• Simultaneous operations of various activities	
Instability and fluidity of employment	• Almost daily workers • Lack of opportunity for safety training	• Provide safety information to improve safety awareness and workers' recognition to accidents*
Lack of workers' safety awareness	• Lack of workers' safety awareness due to fatigue accumulation	
Problems caused by sub-contractors	• Poor safety management system	• Cost rationalization • Support safety training* • Ensure proper construction period
Excessive construction contracts	• Unreasonable demands for construction price, construction conditions, construction period	

*: solutions that can be managed by the level of safety managers

Among solutions for preventing an accident in table 2-1, solutions that can

be managed by the level of safety managers are as follows: establishing strict safety management plan; continuous on-site inspection; and construction safety training. To produce actual results from these solutions, information related to safety should be provided.

Safety management starts from hazard identification, and almost of safety management activities are collecting and analyzing safety information. Collecting safety information activities include hazard identification. Current information used in safety management process is mainly safety technical standards, safe work instructions and accident prevention measurement. Safety information formed as rule-based regulations does not consider the dynamic characteristics of a construction site. Moreover, because safety managers perform their jobs to satisfy these rules, the rule-based information may not be effective in non-typical construction projects.

(3) Potential Role of Construction Accident Cases

A large amount of information and knowledge has been accumulated since the beginning of the construction industry. It is widely regarded that the appropriate use of the accumulated information of the construction industry is the key for a successful project. However, as vast volumes of information became available, the issue of “information overload” has been magnified (Demian and Balatsoukas 2012). Korea Occupational Safety and Health Administration (KOSHA) has accumulated enormous past accident cases. Construction companies have also accumulated accident cases which occurred

in projects they have done.

All accidents are undesirable events and caused by unconsciousness or ignorance of safety managers or workers. To prevent an accident, safety information that can make a safety manager or a worker identify a hazard should be provided (Ahn 1994). Safety information that can be used to prevent an accident consists of safety technical standards, safe work instructions, and past accident cases. Among these, past accident cases are the strongest stimuli and can provide direct information for predicting a risk of an activity. Considering a characteristic that construction accidents occur repetitively under similar conditions, providing similar past accidents can be an effective method for preventing accidents (Ye 1998).

Despite the potential of construction accident cases, there are two problems for utilizing past accident cases. First is which a past accident case is the most similar to a corresponding site. There are already large amount of accident cases accumulated. Moreover, conditions of a corresponding site change continuously. In this situation, selecting an appropriate accident case is very difficult. Second is that the most accumulated accident cases occurred in performing activities. Although the selected accident case can prevent an accident which may occur on an activity area, other types of accidents which occurred in non-activity area, such as walking, are hardly identified by accident cases.

2.2 Hazard Identification

There are enourmous studies focused on the management of hazardous areas. Most of these studies presented methods to manage hazards or hazardous area effectively. For example, Blackmon and Gramopadhye (1995) proposed positive backup alarm that is activated when a worker is in possible danger. Navon and Kolton (2006, 2007) suggested automated monitoring algorithm of fall hazards. In addition, Lee et al. (2012b) developed a labor location-based system of warning laborers when they go near hazardous areas.

These studies can alarm when a worker approaches to a hazardous area, but they do not suggest how to identify a hazardous area. To operate these methods, hazard identification or hazardous area identification should be first performed. However, identifying hazards may require great efforts of safety managers. In this manner, this dissertation overviews hazard identification and applicability of an automated hazard identification procedure to safety management.

In other words, to manage risk, hazards should first be identified, and then the risks should be evaluated and determined to be acceptable or not. Hazard identification the first step for safety management, thus it is important to make construction sites safe. In this section, a brief overview will be provided to present importance and origins of hazard identification and classified hazard identification (by construction phase).

2.2.1 Importance of Hazard Identification

Hazard identification is a crucial part of the safety management. It is really impossible to safeguard a system or control risks adequately without first identifying the hazards. Within the context of how hazards lead to accidents, it indicates that managing hazardous events is a fundamental aspect of construction safety management. In the construction industry, risk assessment is the practical means by which hazardous events are managed.

However, unidentified hazards negate the risk assessment process and the whole safety management process. If a hazard remains unidentified, risks cannot be assessed because the unidentified hazard is not incorporated in safety management process. Moreover, countermeausres for the unidentified hazard cannot be developed and implemented if those involved are not aware of the hazard in the first place. Thus an activity related with the unidentified hazard will remain unprotected, or will feel a level of protection exists that is not justified by the risk assessment (Carter and Smith 2006).

2.2.2 Hazard Classification by Area Type

Among various hazard classification methods used in the construction industry, classification by hazard type is mainly used. It consists of fall, slip, struck-by hazard, etc. This classification has a benefit for dealing with hazards since hazard type is useful information to establish countermeasures that can prevent an accident. Despites its usefulness, this method does not have a strength for identifying a hazard because how a hazard is generated is not

fully explained by using hazard type.

An accident occurs when a worker' act and hazardous condition are combined. Therefore, to identify a hazard, both workers' act and hazardous condition should be considered simultaneously. When workers are inputted to their work, they move to their work area, arrive at their work area, and start their activity. Workers' actions and their location can be a clue for identifying hazards.

The function of space on a construction site explains how a worker acts on a specific space. Among several efforts to categorize construction space types, Riley and Sanvido (1995) classified construction spaces into 13 work-space types by observing the construction process. Based on Riley and Sanvido's (1995) classification, Choi et al. (2014) define six functional space types by merging the construction space types that perform the same functions and the space types are as follows: 1) object space; 2) working space; 3) storage space; 4) set-up space; 5) path space; and 6) unavailable space.

Riley and Sanvido (1995) suggested hazard area and protected area, and Choi (2014) merged them to unavailable space. However, it is difficult to apply their space classification methods to this dissertation since other spaces such as working area (space), personnel path, and path space, can include hazardous conditions.

Based on the space classification suggested by Choi et al. (2014), this dissertation categorizes construction space in a construction project activity oriented area or non-activity oriented area depending on its function as described in Fig.2-4.

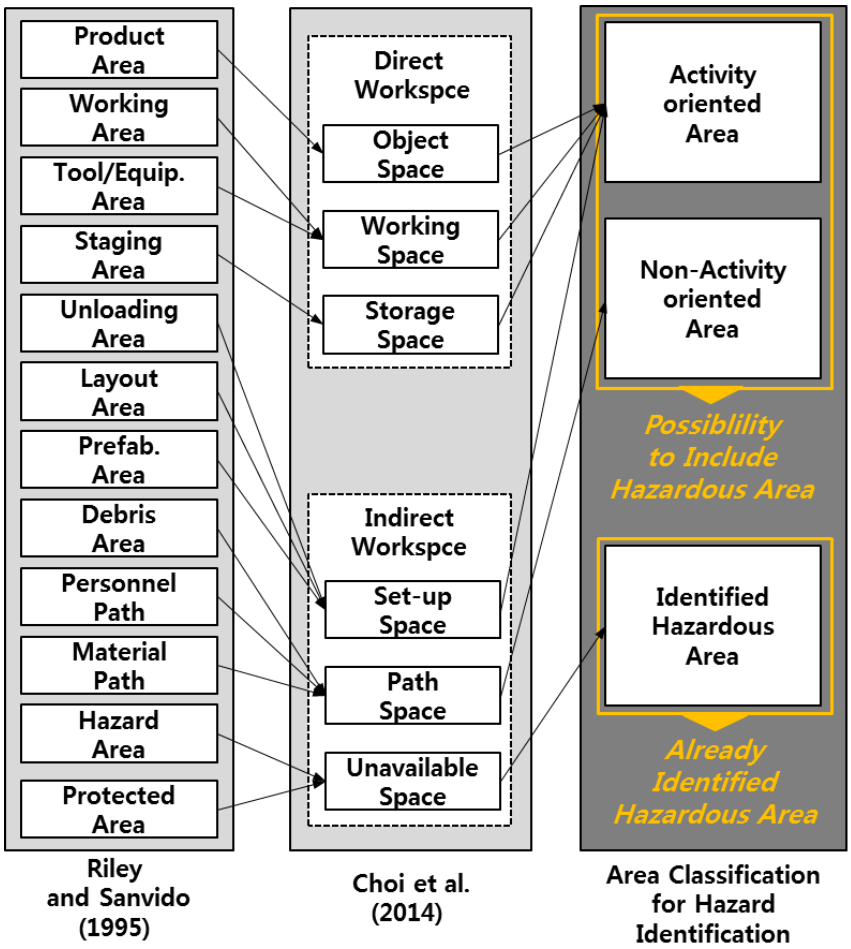


Fig. 2-4. Area Classification for Hazard Identification

Activity oriented area is associated with the execution of specific activity

in a direct way, and the location and size of the activity area is determined by the geometric features of the related object or construction plan for the activity. Activity oriented area includes: (1) object space, which is the area occupied by a building component itself such as a wall, doors, or windows; (2) working space, which is the area required for crews or equipment to execute a specific activity that contributes physical changes in a construction project; (3) storage space, which is the area for storing materials for each activity execution in a construction project; and (4) setup space, which is the area for operating the overall construction project, such as a tower crane and lift car;

Non-activity oriented area has either an indirect relationship with the execution of a specific activity or is associated with the execution of multiple activities. The location a non-activity oriented area is determined by site layout or predefined spatial relationships with the activity oriented area. Non-activity oriented area includes: path space, which is the area required for the movement of resources, such as a laborer or equipment material. Identified hazardous area means an area already identified as hazard or protected for preventing an accident.

Considering origin of hazard in construction sites, hazard identification should include both activity oriented and non-activity oriented. Construction hazards are related to many influence factors, such as kind of activities, construction techniques, environmental conditions and shape of building. Therefore, this dissertation is concerned with the means of identifying hazards

and their countermeasures in the both pre-construction stage and construction stage. For this purpose, it is convenient to divide into construction hazards into two categories.

As forementioned, construction workers' acts can be broadly divided into two types. One is "performing an activity" and another is "moving to performing an activity". In this regard, hazards in construction sites can be classified by two causes: 1) Activity oriented; and 2) Non-activity oriented hazard.

This classification can illustrate the cause of hazard generation which is the key factor for identifying hazards. Because each type of hazard has its own characteristics, a hazard identification method should be selected considering its characteristics.

Workers are exposed to various hazards. As shown in Fig. 2-5, hazards can be classified by two aspects. One is the activity oriented hazards as the first factor can cause harm. In many cases, workers are wounded during performing their jobs. Consequently, the accident spot belongs to the activity area. Another is non-activity oriented hazards. Construction workers travel on the route to the daily work site. When a worker moves around a construction site, he/she faces potentially hazardous objects or conditions such as openings, unprotected dangerous edges, piled materials. For example, an opening is mainly used to transfer materials and equipment between floors. If an opening

remains unprotected, a worker can fall and be injured by this fall.

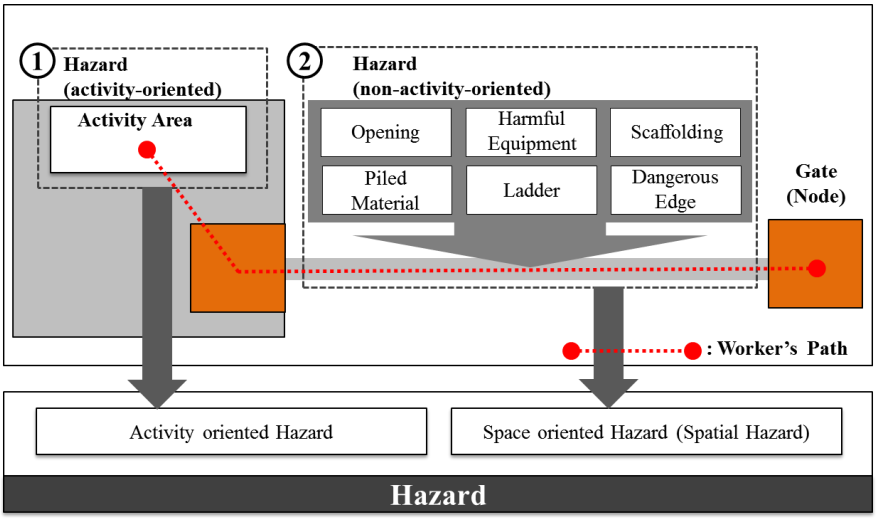


Fig. 2-5. Origin of Hazard in Construction Sites (modified from An 2007)

2.2.3 Hazard Identification by utilizing Information

The time a hazard occurs in a construction site varies. A hazard can occur before and during construction phase. Therefore, hazard identification should be performed considering the characteristics of a hazard. Some hazard can be identified in pre-construction phase. This kind of hazards can be identified before occurrence by using a construction schedule or drawings. Other hazard is difficult to be identified because it can be searched after it occurs. In this case, a hazard is identified by safety managers' observation (inspection) and workers' report.

As shown in Fig. 2-6, hazard identification activity starts before hazard

occurrence (pre-construction phase) and lasts throughout construction phase. In the pre-construction phase, safety managers analyze a construction schedule and drawings, and they search potential hazards that can occur in their construction sites. These hazards are mainly activity-oriented and the similar past accident cases can be searched by using accident cases.

Hazards on paths that can be identified after they occur are identified by human observers' inspections. However, investing every place continuously requires enormous efforts and time. Thus, a method overcoming this problem is needed. This disseratation focuses on laborers' hazard perception. The concept is that workers deviate from their own movement routes when they perceive a hazard on their routes. This deviation can be detected by using a location sensing technique and BIM. The next two sections describe related research and techniques to identify each type of hazard.

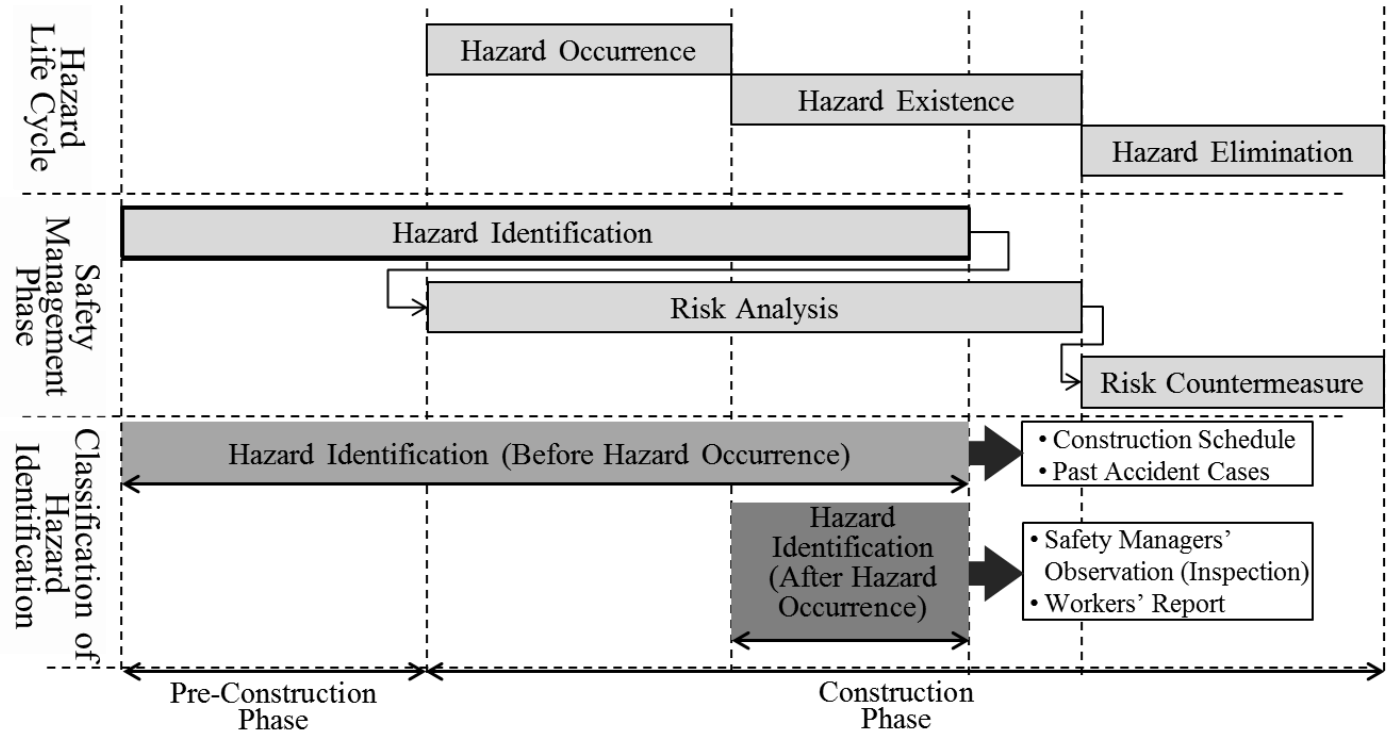


Fig. 2-6. Hazard Identification Classified by Construction Phase

2.3 Hazards on Activity Areas

A large amount of information and knowledge on the construction industry has been accumulated. It is widely understood that the appropriate use of the accumulated information is the key for a successful project. Therefore, searching for necessary information is one of the most important steps in information and knowledge management. Information retrieval is a candidate to be utilized in searching for useful accident cases (Kartam 1997) since the accident context in the construction industry can be easily structured and stored due to the repetitiveness of accidents (Ye 1998).

Fig. 2-7 illustrates how a construction schedule can be used for identifying hazards. A construction schedule consists of various information including activity names, duration, equipment, workers, etc. By extracting information, a query can be generated to search past similar accident cases that can help prevent an accident on the construction site. This section will review current accident case retrieval method and present a method to overcome them in detail.

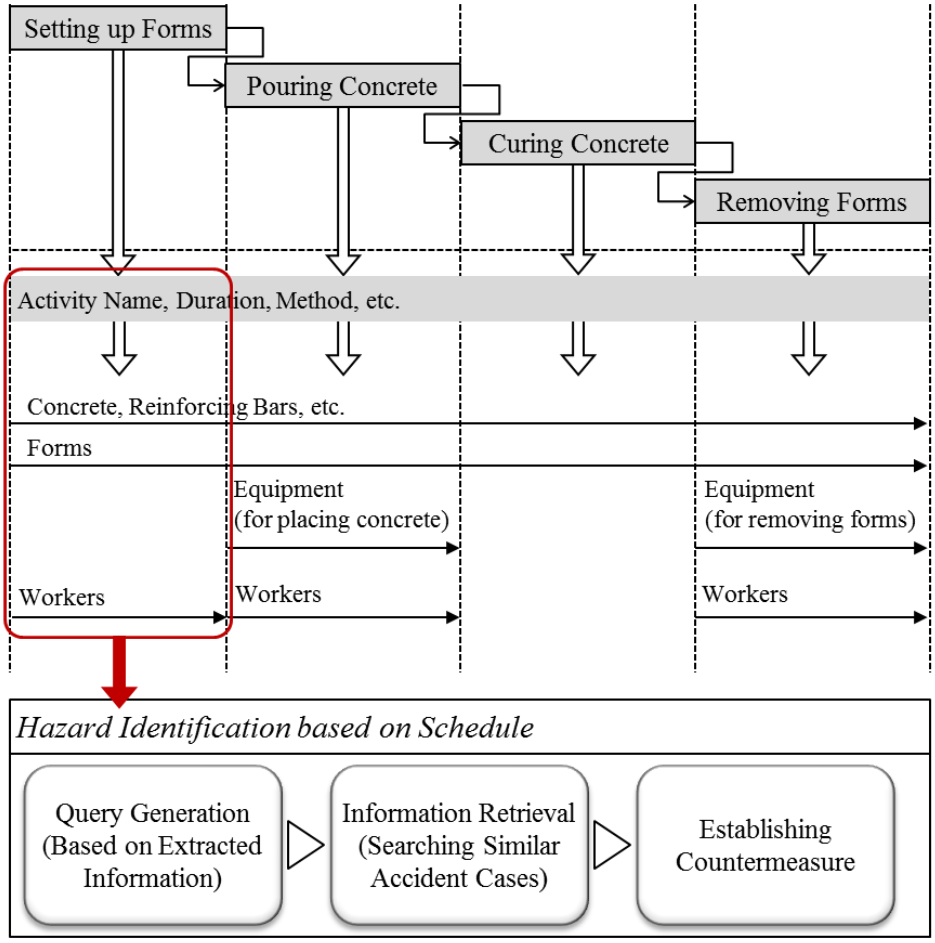


Fig. 2-7. Hazard Identification based on Construction Schedule

2.3.1 Hazard Identification using Construction Information and Accident Cases

Many researchers have made an effort in finding a proper method (e.g. CBR, expert system, information retrieval, etc.) to be able to acquire required information. Studies used CBR to estimate cost (Yau and Yang 1998; Karshenas and Tse 2002; Koo et al. 2010), support making a decision

(Morcous et al. 2002; Chua and Loh 2006), and establish a schedule (Tah et al. 1998; Ryu et al. 2007). The expert system is also used in various fields such as material (Bai and Amirkhanian 1994), water and sewer network (Ortolano et al. 1990; Shepherd and Ortolano 1996), and claim management (Diekmann and Gjertsen 1992).

In the construction industry, information retrieval can be broadly divided into text-based (also including numbers) and image-based retrieval. The former has been used in searching legal information (Mahfouz 2009), domain knowledge (Rezgui 2006; El-Diraby et al. 2005), and cost information (Woldesenbet and Jeong 2012). The latter has been used to retrieve construction drawings and related information (Yeh et al. 2012).

Even though these techniques have improved the use of accumulated information, there had been relatively few studies in the safety information area. To identify construction hazards, many researchers have made an effort based on various methods. Information retrieval–based methods have been suggested. Kartam (1997) developed the integrated knowledge-intensive prototype for safety management. This system integrates safety database and scheduling software such that safety information will be tagged onto each activity. Chua and Goh (2002) devised a CBR approach to retrieve useful cases for safety planning and hazard identification. Zhao et al. (2009) suggested a learning hazard and operability (HAZOP) expert system based on case-based reasoning. Carter and Smith (2006) suggested a web-based

application including knowledge and experience of all personnel within a company. Goh and Chua (2010) developed a hazard identification model using case-based reasoning (CBR). It was established based on accident cases. Despite their strenghs on hazard identification, these methods have a limitation on usablility. These did not provide a function that reflects construction progress or workers' conditions.

By contrast to the construction industry, the expert system has been adopted to improve and automate hazard identification in the chemical industry (Catino and Ungar 1995; Suh et al. 1997a; Suh et al 1997b). These systems were developed to shorten time and manpower for detailed hazard identification.

Aside from CBR and expert system, information retrieval is another candidate to be utilized in searching for accident cases. The context of accident in the construction industry can be easily structured and stored due to the repetitiveness of accidents (Ye 1998). Many countries have management agencies that handle safety and health. Government organizations (e.g., Occupational Safety and Health Administration (OSHA, USA) Health and Safety Executive (HSE, UK), and KOSHA (Korea Occupational Safety and Health Agency (KOSHA, South Korea), etc.) have provided accident cases to prevent the occurrence of similar accidents. Furthermore, they also provide web-based search engines in order to support the improvement of safety management. It can be observed that application of information retrieval to

hazard identification is feasible, but the current approaches do not appear to be suitable for the construction industry.

2.3.2 Current Accident Case Retrieval Systems

Current accident case retrieval systems (e.g., information retrieval systems of OSHA, HSE, and KOSHA etc.) have provided useful accident cases. The algorithm for accident case search by KOSHA, OSHA and HSE is simple and easy to use. The system begins with an input of a query in a search box. The algorithm finds an item (accident case) or items with specified properties among a collection of items. Finally, the result (relevant item) is presented to the user. This process is familiar to general and untrained users. However, as shown in Fig. 2-8, this process is not effective when a user wants to search for vast quantities of information. A user needs to perform repetitive retrieval works (e.g., generating queries, inputting queries, selecting a result, providing a result to a relevant worker, etc.) in order to search for various past accident cases that can be used to prevent accidents in the current project. Nonetheless, the task of a safety manager is not considered in the retrieval procedure and algorithm. Despite the advantages of these systems, they face some limitations in terms of the user and usability, query generation and providing results, and the retrieval method.

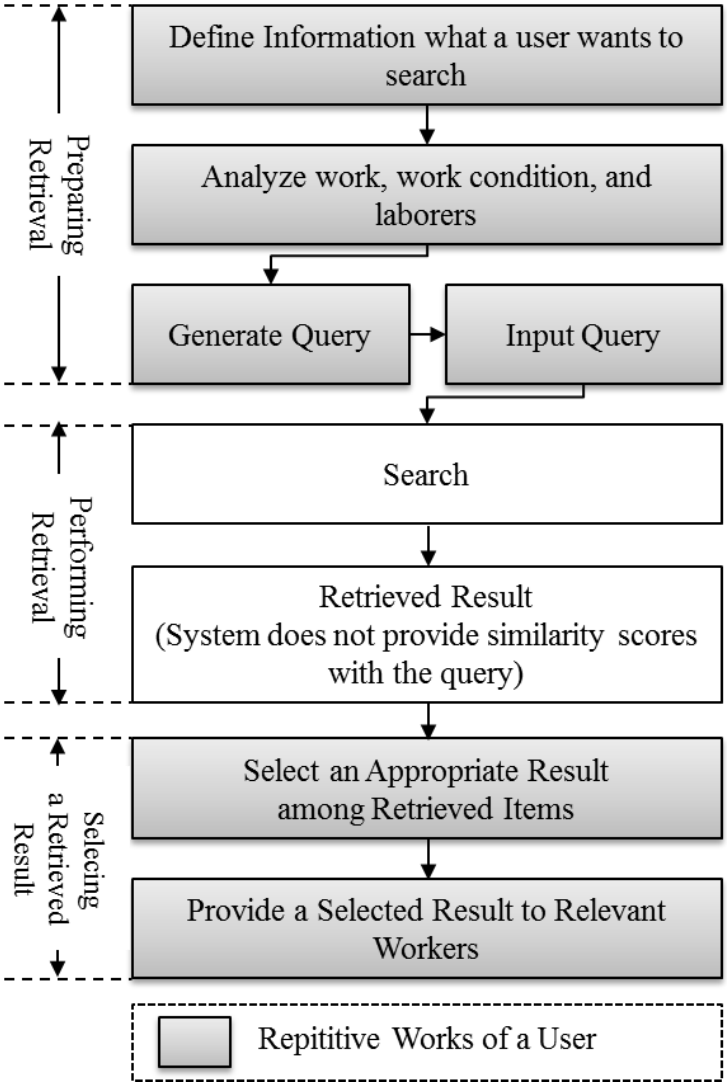


Fig. 2-8. The Procedure of Information Retrieval in Existing Systems

2.4 Hazards on Movement Paths

Although activity-oriented hazards can be identified by using a past accident case retrieval system, identifying non-activity oriented hazards is another problem. Because most accumulated accident cases are caused by activity oriented hazards, a retrieval method is difficult to apply to non-activity oriented hazard identification. This section will discuss the characteristics of non-activity oriented hazard identification. Based on this, current practices and their limitations are presented. A potential solution and its related works are reviewed.

2.4.1 Characteristics of Non-Activity Oriented Hazard Identification

Non-activity oriented accidents are usually caused by the combination of workers and spaces, especially paths. The level of management for moving processes or paths is usually lower than for work spaces. A condition of a construction site changes over time, which can make it difficult for safety managers to identify the generation and extinction processes of hazards on the movement paths. Hazard identification on the movement paths is generally performed by human observers, and requires additional efforts (e.g., more managers, time, and cost).

Non-activity oriented hazard identification performed by a human inspectors has an inherent problem in terms of timing. Fig. 2-9 illustrates an example of the problem. If a human observer searches hazards starting from

the basement and moves along the red line, he/she can identify a hazard generated (occurred) before inspection. However, a hazard generated after inspection is not included in the safety management process. The yellow star mark on the first floor in the Fig. 2-9 is generated after a safety manger inspects. Until a safety manager repeats inspection, the hazard is not identified. Therefore, a method that can minimize duration– a hazard exists without any identification and safeguard-is required.

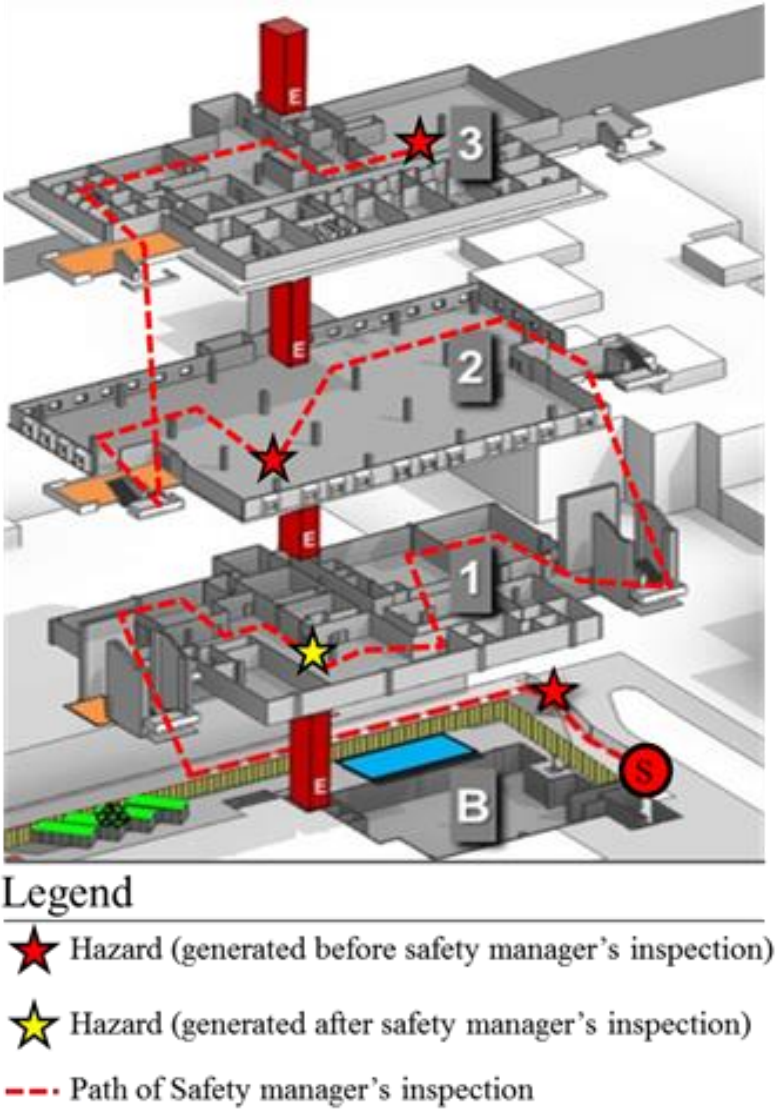


Fig. 2-9. An Example of Non-Activity Oriented Hazard Identification Performed by a Human Observer

2.4.2 Current Non-Activity Oriented Hazard Identification

To prevent accidents, hazard identification is necessary (Korea Occupational Safety and Health Agency 2001-2003). Unidentified hazards mean that hazards are not included in the safety management process. There are various factors causing accidents in a construction site. These factors are managed by safety managers on site, where the hazard identification is usually carried out before an activity begins.

The traditional way of identifying construction hazards is through a rule-based checklist established based on accident cases and best practices. Traditional hazard identification methods have failed to identify all of the hazards that should have been identified (Carter and Smith 2006) because hazards are generated by a combination of unexpected conditions in various times and places as a construction project changes dynamically and progresses. As a result, it is almost impossible to identify all of the hazards before the beginning of construction. Ineffective hazard identification can result in unsafe site conditions and construction processes (Le and Skitmore 2012).

To identify construction hazards, many researchers have made an effort based on various methods. Information retrieval–based methods have been suggested by Carter and Smith (2006), Goh and Chua (2010), and Kim et al. (2013). Carter and Smith (2006) suggested a web-based application including knowledge and experience of all personnel within a company. Goh and Chua (2010) developed a hazard identification model using case-based reasoning

(CBR). It was established based on accident cases. Kim et al. (2013) introduced an information retrieval system that also used accident cases. This system can help to identify a hazard by using a formalized method based on accumulated accident cases. However, current accident cases are accumulated based on activities. The results of these methods are mostly text-based cases and the coordinates of a potential hazard are difficult to determine. Therefore, managers should devote an additional effort to convert text-based cases to visual information.

In this regard, visualization has been applied to the design phase to help all construction participants in identifying hazardous activities or areas (Li and Skitmore 2012). Recently, a BIM-based 4-dimensional model used to detect potential on-site safety hazards. Despite the fact that some research has been conducted on improving construction hazard identification by using visualization technologies, there is still the problem of applying the visualization technologies for hazardous area identification in an on-going project. The main reason is that it is difficult for an established visualization model to reflect the dynamic changes of a construction site because even a well-established BIM model does not fully present. Another problem is that a safety manager checks every visualized area in a computerized system to identify the potential hazards on a real site.

A real-time location tracking system (RTLS) can be an alternative to solve these two problems. Workers' location logs include lots of information that

can represent workers' moving paths. Moreover, if there is proper data processing, workers' location logs can be also used in safety management (Navon and Kolton 2006). Navon and Kolton (2006), and Lee et al. (2012) suggested RTLS-based safety management systems that track workers' locations. Generally, there are two methods for tracking location: one is global positioning system (GPS) and another is radio frequency identification (RFID) (Yelamarthi et al. 2010; Oloufa et al. 2003). Construction conditions can be broadly divided into indoor condition and outdoor condition. GPS—one of the most common location tracking techniques—can be effectively used in outdoor conditions. However, this technique is difficult to apply to construction sites because received signal strength may lower in an indoor condition.

On the other hand, RFID has enough positioning accuracy in an outdoor condition as well as an indoor condition if the receiver and transmitter devices are sufficiently installed on a construction site. Construction hazards can be found in both indoor and outdoor conditions. Therefore, the RFID technique that can be applied to both conditions is used in this study. By analyzing workers' locations we can get information, such as the point where a worker is currently located and a worker's path. Based on this information, this paper suggests a hazardous area identification system.

2.4.3 Location Sensing Methods

This chapter introduces a location-sensing selection procedure based on comparative overview. Literature and current approaches about how the accuracy of a selected location-sensing technology can be improved are reviewed.

(1) Comparative Overview

One of the most widely used RTLT technologies is the global positioning system (GPS) that locates a target object by using a network of 24 satellites (Garmin 2013). GPS is quite effective in locating and tracking objects in outdoor environments (Ni and Liu 2004) but has an inherent weakness in locating indoor objects. This has limited the effectiveness of GPS in building construction projects where many activities are conducted indoors. Therefore, there is a need for a method that can be applied to indoor and outdoor conditions.

There are diverse locating methods that can determine the location of objects in both indoor and outdoor conditions. These methods consist of location measurement methods (e.g., triangulation, scene analysis, and proximity), distance measurement methods (e.g., angle of arrival (AOA), received signal strength indication (RSSI), time of arrival (TOA), and time difference of arrival (TDOA)), and technologies (e.g., Infrared, IEEE 802.11, Ultrasonic, ultra wide band (UWB), and

RFID). These methods have problems, such as multi path, obstacle (shadow area), and signal loss (Jin et al. 2006; Han et al. 2012).

IEEE 802.11b products are easy to set up for general indoor conditions as well as construction sites. Nevertheless, they do not guarantee high-accuracy location-sensing. Ultrasonic has a relatively higher accuracy than other technologies. Despite its higher accuracy, an ultrasonic-based system requires a great deal of infrastructure and cost in order to be highly accurate. Another problem is that it is difficult to install this system on construction sites. Construction sites have diverse noises, including high-frequency sound, which can cause interference with ultrasonic wave. Although UWB presents the highest accuracy, it is difficult to be applied to outdoor construction activities because the signal transmission distance is approximately 20m.This short signal transmission distance means that a huge number of readers are required for covering a whole construction site and it is not cost-effective.

RFID is relatively cost-effective and has high capability of information transfer. Despite its strengths, it is heavily influenced by environments which cause multi-paths, obstacles (i.e., shadow area), or signal loss (Jin et al. 2006; Han et al. 2012). This is because the accuracy of location sensing largely relies on the strength of the tracking signal (Lee et al. 2012), which can vary depending on the

tracking environment. RFID is a small electronic device that consists of a small chip and an antenna.

An RFID-based RTLS can store and retrieve location data by utilizing a number of tags (which are attached to target objects) and readers (i.e., tracking equipment). To locate a target object, a reader receives data emitted from a tag attached to the target object. After transmitting and receiving the location data, an RFID-based RTLS can calculate the coordinates of a target tag.

Many researchers have proposed the use of RFID for indoor object location tracking because of its good tracking performance and cost-effectiveness. For example, Ni et al. (2004) developed an RFID-based indoor location sensing systems that can improve the overall accuracy of location sensing by applying reference tags. Song et al. (2006) developed a mathematical model that estimates the signal strength within a construction site. Similarly, Skibniewski and Jang (2007) suggested a framework for automated real-time location tracking of construction materials based on RFID signals combined with ultrasonic waves. By providing supplementary tracking tools (e.g., a signal strength fingerprint map), these approaches are effective in tracking indoor objects when site conditions are well pre-determined. When installing the indoor tracking systems (i.e., determining the suitable location of the readers), a signal strength fingerprint map has

strong potential to improve the overall accuracy of RTLS. However, the tracking environments in a building construction site can vary significantly as the project progresses. Therefore, this paper suggests an enhanced indoor RTLS that can improve the accuracy of location tracking in building construction projects.

(2) Location Sensing Accuracy Improvement Methods Focusing on Workers

Tracked objects have their own characteristics (e.g., direction, range, and velocity). Thus, if we understand the characteristics of the target, then it can be involved in an RTLS (Mullen 2005; Son et al. 2013). There are a few researchers dealing with movement characteristics of a target for location-sensing (Lei et al. 2013; Son et al. 2013). One main example is the research of Son et al. (2013). They suggested an enhanced asset locating system for vehicle pooling in a port terminal. This system focused the range and speed of a transfer crane. A transfer crane has three types of path, range, and velocity (x-, y-, and z-axes). Moreover, each axis has a range that can be movable.

Unlike vehicles in a port terminal, construction workers do not have a specific direction. A worker can move in all directions. One similar characteristic between a construction worker and a vehicle in a port terminal is having a velocity range. If information about the velocity range of a worker is incorporated into the location-sensing engine, a

log located on outside of a target's range means that a measured log can be considered an error. Based on the consideration of this characteristic of tracking a worker's location, this paper selects the velocity of workers as a primary factor to mitigate estimation errors, and it excludes workers' paths.

(3) NLOS Error Mitigation Methods

NLOS environments can heavily decrease the accuracy of location sensing. Uses of the reference tags and the assistant tags are two primary examples that can increase the accuracy of location sensing without the additional installation of readers. The concept of the reference tags is introduced in LANDMARC (Ni et al. 2004). It serves as a reference point in the system. The use of it does not need a large number of expensive RFID readers, instead use requires extra RFID tags, which are cheaper than RFID readers. It can also make location-sensing results more accurate and reliable. In spite of reference tags have advantages, they should be installed on floors so that they can interrupt construction activities. In terms of constructability, an assistant tag can be an alternative. Though the concept of the assistant tag is very similar to that of the reference tag, the primary difference is the installation place.

In this paper, assistant tags are attached to workers' safety helmets, and tags follow workers' movements. When a tag is in an NLOS

environment, another tag that can be detected in an LOS environment can communicate with the first tag (in the NLOS environment) and readers. This approach has been attempted to convert an NLOS environment to an LOS environment (Cho et al. 2008; Lee et al. 2012). By utilizing the concept of assistant tags, interruptions for construction activities can be minimized and additional effort for installing reference tags or readers is not required. Thus, considering the characteristics of construction projects, assistant tags are select to overcome the NLOS environment. However, they do not suggest methods such as how to determine an environment and to select a proper assistant tag. In this paper, current problems to apply assistant tag concept will be addressed.

2.5 Summary

Hazard identification on construction sites should be performed with consideration of hazard characteristics. To understand characteristics of hazard, definitions and life cycle of hazard are analyzed. Safety management process–includes hazard identification–is also examined. Based on this, the origins of hazard in construction sites can be divided into two kinds: activity oriented hazard and non-activity oriented hazard.

Each type of hazard can be identified by using different methods. Activity oriented hazards are generally repetitive, thus using past similar accident cases can help identify a hazard in a corresponding site. Although there are abundant accident cases that can be used to prevent an accident, extracting useful accident cases is difficult. Current accident case retrieval systems have tried to provide accident cases and these systems obviously have advantages.

However, they make a user perform repetitive retrieval works (e.g., generating queries, inputting queries, selecting a result, providing a result to a relevant worker, etc.). This problem can be subdivided into three kinds: 1) user and usability; 2) query generation and providing results; and 3) retrieval method.

Safety managers and safety management focus mainly on the activity areas. However, more than 20% of construction accidents occur on worker's

movement path. The combination of a worker and a hazardous area can become an accident. To identify a hazard on the path, current practices have adopted human observer-based inspection. This method requires additional efforts (e.g., more managers, time, and cost) and has an inherent problem in terms of timing.

Although there are various techniques that are suggested to handle this problem, they face several limitations, such as text-based identification which requires an additional effort to convert text-based cases to visual information and visualization-based identification in which a safety manager checks every visualized area in a computerized system. A RTLS can be an alternative to solve these problems. By tracking and analyzing workers' location logs, a system can identify hazardous areas where workers recognize them as a risky area. Before developing a system, several methods are analyzed to improve location sensing accuracy which is directly related to the system performance.

Through the preliminary study, two approaches for identifying two types of hazard are suggested based on the consideration of current practices and each hazard characteristic. The following sections will present a process how two approaches are developed.

Chapter 3. Activity Oriented Hazard Identification

This section will discuss how to establish a past accident case retrieval system. First, requirements for improving an accident case retrieval system are analyzed. Second, the indices for generating queries and calculating similarity scores are extracted, classified, weighted, and integrated to retrieve and rank cases. Third, this section covers how to obtain and establish a data source for system development. Finally, a system framework is suggested based on these two preceding processes.

3.1 Requirements for Overcoming Limitations of Current Accident Case Retrieval Systems

As forementioned, the current accident case retrieval systems have limitations such as the user and usability, query generation and providing results, and the retrieval method. This section will discuss how these limitations can be handled.

3.1.1 User and Usability

The progress of information technology has helped address various challenges presented by the construction projects (Paulson 1995), and

information technology has been introduced to safety management. Many researchers use safety information by relating construction activities or adapting accident cases to safety planning (Ko et al. 2005; Ahn 1994; Moon et al. 1997; Ye 1998). Government administrations have also developed safety information systems. Currently, the KOSHA and the OSHA provide a web-based search services to prevent accidents.

These systems focus on improving the practical use of accumulated past accident cases. They provide the information on the related past accident cases when a user inputs queries. Generating a query every time for a user to retrieve information makes the current systems inconvenient. Furthermore, under work pressure, workers do not have enough time to search for safety countermeasures or past accident cases to eliminate risk factors.

Furthermore, there is another problem that must be solved for usability. Since workers are overconfident in their skills and experiences, they usually do not recognize the necessity of using past accident cases. Although changing the workers' attitudes toward safety is of the utmost importance (Lee et al. 2012a), it is also necessary that past accident cases be provided to workers even if they do not voluntarily search for the information. This issue can also be resolved by using a push system.

Push system is one of the subsystems in the knowledge management system and can be defined as automatically providing the right knowledge to

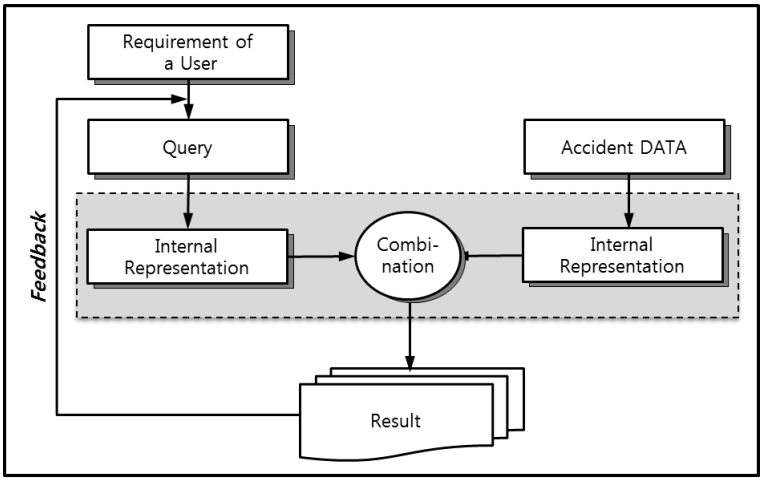
the right person (Meso and Smith 2000; Zhang et al. 2009). To provide similar accident cases to a worker by using a push system, it is required to have the information about where an activity is performed and who is related to the activity. If a past accident case is retrieved based on a construction site's condition, work, and worker, it can be automatically provided to the related works or workers.

The inconvenience and avoidance of voluntary retrieval indicates that the accumulated past accident cases are not fully used. To solve this issue, this study suggests an information retrieval system that automatically searches and pushes similar past accident cases to the related workers. However, most workers are usually reluctant to use such information systems and retrieve information related to them. Therefore, this study suggests that safety managers receive the search results including the related workers' ID from the system. Based on the retrieved information, managers can deliver past accident cases to the related workers and control expected risk.

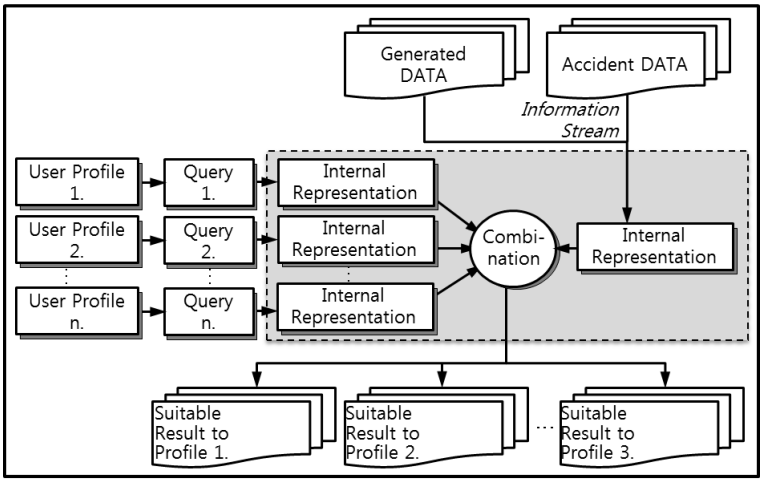
3.1.2 Query Generation and Providing Results

Generally, information retrieval can be achieved by using either Ad hoc systems or filtering systems (Jukna 2012). For Ad hoc systems, a new query by the user is required whenever new information is necessary. Filtering systems store user interests in the user profile and its main objective is to select newly generated information and remove unnecessary information. Furthermore, a routing system—one of the information filtering systems—

automatically pushes search results to end-users. As shown in Fig. 3-1, the difference between these two systems is whether a query is short-term (one-off) or not (semi-permanent).



(a)



(b)

Fig. 3-1. Retrieval Systems: (a) Ad hoc System; and (b) Routing System

If users want to identify countermeasures by retrieving accident cases, current safety retrieval systems using the Ad hoc model should input as many accurate queries as possible with the number of risk factors. However, with the routing model, users do not need to generate queries every time they need to retrieve information. This model retrieves information whenever there is a change in the user profile or the database. Also, it can push results to the related users. Therefore, the routing model is adopted to derive an involuntary user-centric system.

To establish the routing model, the information required by safety managers or workers should be defined and converted into queries. Operating the model is only needed to obtain search results after compiling user profiles, which are automatically converted into queries by the information retrieval system (Jukna 2012). It is also helpful for the user to have a ranked list of results with the information retrieval system (Manning et al. 2008).

3.1.3 Retrieval Method

Most of the current safety information retrieval systems have been developed using the Boolean model to compare queries and the database and extract related cases (Ye 1998). The Boolean model is processed based on whether or not the documents contain the query terms. Although the Boolean model is simple and clear, the results can vary from too many or too few results due to its strict matching rule (Yves and Hammer 2010).

Another problem is that each index is weighted equally (Ye 1998). This leads to a decrease in the effectiveness and usefulness of retrieval results (Clote and Kranakis 2002). For instance, if a user inputs a query, Boolean systems only retrieve exactly matching results and do not provide query-related results. Therefore, a partial match search function that can calculate the similarity between a query and results is necessary. This means that indices of queries or texts are represented as non-binary weights (Manning et al. 2008). Lin and Soibelman (2009) suggested an extended Boolean model that is a hybrid retrieval type of a vector model which can overcome the limitation of the original Boolean model.

Even though the extended Boolean model has been rarely adopted in construction safety management, it may improve the efficiency and quality of hazard identification. To achieve this improvement, this study suggests an information retrieval system that combines the routing model with the extended Boolean model. If these two models are combined, the system can automatically provide retrieval results including their ranks.

3.2 Indices for Information Retrieval

Information retrieval systems require queries to search data. In general, a query means a form of questioning, in a line of inquiry. In the area of computing, a query can be defined as a precise request for information

retrieval with database and information systems. Therefore, it is important for users of an information retrieval system to compose a proper query. A query is a set consisting of a single index or indices. Moreover, a query should be composed in its objective. A query for an accident case retrieval system should include proper indices that can retrieve the most similar past accident case. This section will present three steps for how indices are extracted, classified and weighted. After that, a similarity calculation process that determines the most similar past accident case will be explained.

3.2.1 Extracting Indices

Construction accidents are caused by the influence of various risk factors (Lee et al. 2012a). In the previous studies (Lee et al. 2012a; Choudhry et al. 2009; Hallowell and Gambatese 2009; Lee and Halpin 2003), various risk factors are extracted and evaluated. This study is the follow-up study of Lee et al. (2012a)'s. This study adopted the ideas of hazard identification and risk assessment performed in Lee et al.'s study. Therefore, this study adopts ten factors extracted from the study completed by Lee et al. (2012a). The details of the extraction process are shown in Fig. 3-2. The extraction process of Lee et al. (2012a) consists of two steps.

The first step is to collect various risk influence factors from research studies and disaster investigation according to the code standard (South Korea, United States, Japan and Germany). These studies and investigation standards suggest many influence factors and 27 factors which are repeatedly mentioned

at least three times. The second step is to select factors with more influence than others. A proper number of factors are advantageous to safety management. To identify risk influence factors, surveys were conducted with 42 safety managers, each with more than 10 years of experience in the field. The safety managers suggested that the 10 factors shown in the upper portion of Fig. 3-2 are much influential than others. Those selected 10 factors are used as indices for generating queries in this study. Although all the factors are used as indices for composing queries, not all of them are used in every phase of safety planning.

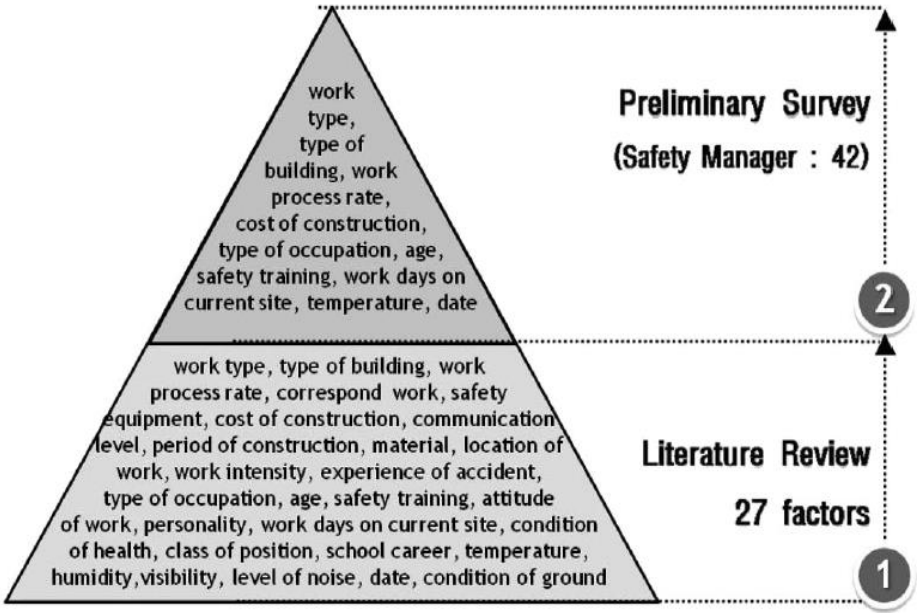


Fig. 3-2. Extracted Factors (adopted from Lee et al. 2012a)

3.2.2 Classifying and Weighting Indices

As shown in Fig. 3-3, safety management plans are usually organized into three phases: preliminary plan, monthly plan, and daily plan. The uncertainty of construction projects makes it necessary to acquire information at each safety planning phase (KOSHA 1999). When safety managers establish a preliminary plan, they set a rough plan because there is not sufficient information for a detail plan. In this phase, safety managers can use limited information. However, available information increases as construction schedule progresses. A daily plan for preventing an accident requires a detail countermeasure. Predictions should be made about activities performed during the day and accidents which should be anticipated, all of which requires more information (indices for generating a query) than previous phases.

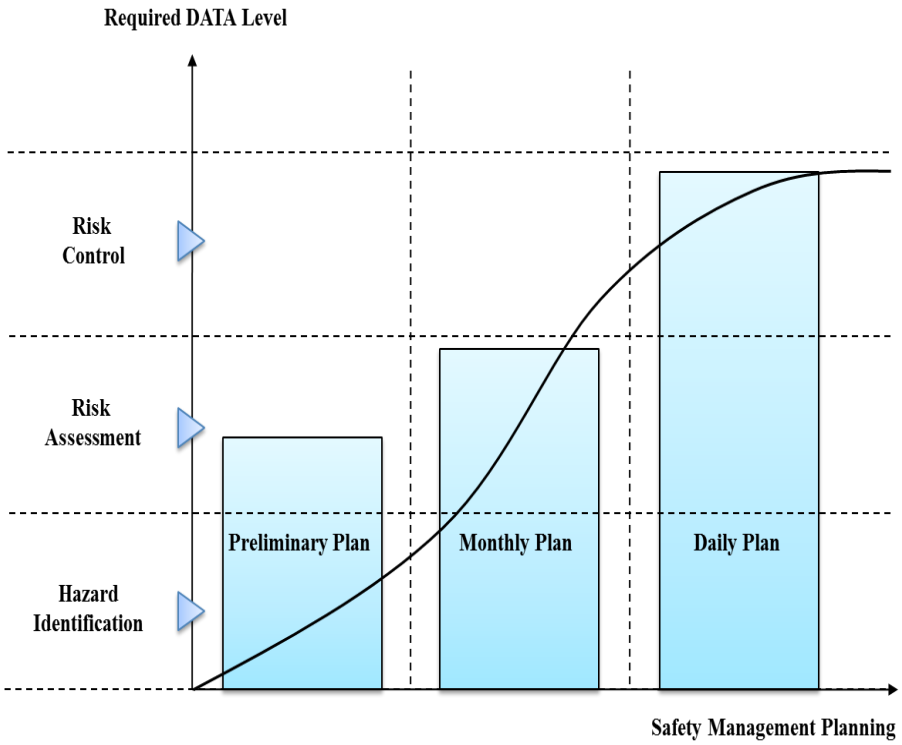


Fig. 3-3. Required Data Level by Safety Management Phase

Therefore, classifying factors is required to make the fittest queries for each safety planning phase. The committee—which is composed of 13 members, each with more than ten years of experience in the safety management field—had presented their opinions to determine the classification of the ten factors. Table 3-1 presents factors that are obtainable and required to retrieve information at each phase.

Table 3-1. Classification of Indices by Safety Planning Phase

	Classification	Preliminary Plan	Monthly Plan	Daily Plan
Influence Factors	Work process rate	○	○	○
	Cost of construction	○	○	○
	Work type	○	○	○
	Building type	○	○	○
	Occupation type	-	○	○
	Date	-	○	○
	Age	-	-	○
	Workdays on current site	-	-	○
	Safety training	-	-	○
	Temperature	-	-	○

Also, determining the weights of indices is necessary to improve the effectiveness of the similarity measurement (Kolodner 1992). To weigh the indices extracted in the former section, the analytic hierarchy process (AHP) developed by Saaty (1980) was used. AHP's capacity to convert qualitative to quantitative values was used to weigh indices.

At first, indices are classified by three safety management phases. Questionnaire includes pair-wise comparisons of between indices of each plan. The pair-wise comparisons are performed at preliminary, monthly, and daily plan. Results with a contingency index lower than 0.1–that confirm the

reliability of the survey and its results–are selected. The results of questionnaires are integrated, and the weights of indices are calculated. Fig. 3-4 shows the AHP application procedure for weighting indices.

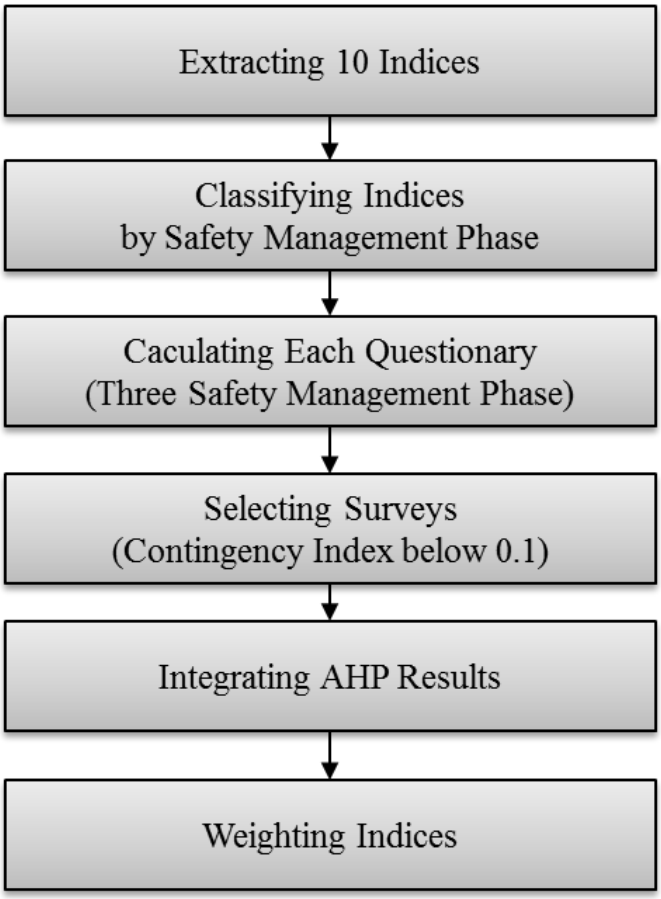


Fig. 3-4. AHP Application Procedure for Weighting Indices

The questionnaire for calculating weights follows that of Lee et al. (2012a)'s. It was sent to 50 experts, each of whom had worked more than ten years as a safety manager. Of the 50 questionnaires distributed, 43 were

collected. The weights of indices were analyzed based on questionnaires with a consistency index below 0.1. The results of the surveys and AHP analysis are shown in Table 3-2.

Table 3-2. Weight of Indices by Safety Planning Phase

	Classification	Preliminary Plan	Monthly Plan	Daily Plan
Influence Factors	Work process rate	0.241	0.167	0.099
	Cost of construction	0.175	0.131	0.077
	Work type	0.334	0.238	0.125
	Building type	0.250	0.173	0.096
	Occupation type	-	0.181	0.100
	Date	-	0.109	0.059
	Age	-	-	0.069
	Workdays on current site	-	-	0.118
	Safety training	-	-	0.166
	Temperature	-	-	0.089
Survey	No. of distributed surveys	50	50	50
	No. of collected Surveys	43	43	43
	No. of collected surveys with a consistency index below 0.1	43	31	23

3.2.3 Similarity Calculation

To judge the degree of congruency between each index, it is important to determine the proper data format (Ye 1998). Depending on the type of data, the similarity index (SI) calculation method varies. The types of data used in this study are string and numeric. If data follows a numeric format, the distance between two values that correspond to the construction site condition value and the accident case is calculated using Equation 3-1:

$$SI = 1 - \frac{|A - B|}{A} \qquad \text{(Eq. 3-1)}$$

where A = condition value of a current site; and B = condition value of a past accident case.

When the type of data follows a string format, SI is 1 if the condition value of a current site and the condition value of a past accident case are identical and 0 if they are completely different. The method of calculating similarity index method and data format are shown in Table 3-3.

Table 3-3. Data Format and Calculating Similarity Index Method

Influence Factors	Definition	Data Form	Calculating Method
Work process rate	Ratio of work process	Numeric	$SI = 1 - \frac{\lvert A - B \rvert}{A}$
Cost of construction	Cost of construction Project	Numeric	$SI = 1 - \frac{\lvert A - B \rvert}{A}$
Work type	Type of trade	String	If A=B, SI=1 If A≠B, SI=0
Building type	Type of building	String	If A=B, SI=1 If A≠B, SI=0
Age	Number of years since born	Numeric	$SI = 1 - \frac{\lvert A - B \rvert}{A}$
Occupation type	Type of occupation	String	If A=B, SI=1 If A≠B, SI=0
Workdays on current site	The days of working in current site	Numeric	$SI = 1 - \frac{\lvert A - B \rvert}{A}$
Safety training	The number of safety trainings in three months	Numeric	$SI = 1 - \frac{\lvert A - B \rvert}{A}$
Date	Correspond date of site	Numeric	$SI = 1 - \frac{\lvert A' - B' \rvert}{A'}$
Temperature	Temperature of site	Numeric	$SI = 1 - \frac{\lvert A' - B' \rvert}{A'}$

Equation 3-1 cannot be directly applied to two indices (e.g. date and temperature). The construction site condition and accident case values (of date

and temperature) need to be converted into modified values to calculate similarity. If site condition (A') is in January and accident case (B') is in December, they have a very high degree of similarity in the continuity perspective. However, the degree of similarity is determined to be zero if Equation 3-1 is applied.

To solve these problems, the weights of each month and temperature, as suggested by Lee et al. (2012a), are adopted to calculate SIs. The number of accidents, nonfatal injuries, fatalities, lost workdays, and worker inputs are taken into account when calculating the weight of the influence factors, and Equation 3-2 shows how the weight of risk influence factor is determined. The values in the risk influence factors are assessed by each factor's frequency and severity.

$$Site\ Condition\ Value = \sqrt{Frequency \times Severity} = \sqrt{\frac{NA_i}{WI_i} \times \frac{LW_i}{NA_i}} = \sqrt{\frac{LW_i}{WI_i}} \quad \text{(Eq. 3-2)}$$

where NA = number of accidents; LW = lost workdays; and WI = worker input (the ratio of entire worker force on the site to the worker force of corresponding work type, and the unit is percentage).

The detailed severity and frequency assement is as follows.

(1) Severity

Lost workdays resulting from injuries and fatalities are used, in

conjunction with insurance fees paid, to assess the severity of risk in a work type. For corporations, insurance fees paid were familiar and easy to tabulate and assess. Most corporations are interested in accident-generated costs. Using costs to assess the severity of accidents has advantages. However, severity calculation based on cost can vary by country, occupation, position, and wage. In this paper, therefore, loss of workdays provides the basis for the severity of risk assessment. Lost workdays includes days away from work due to both nonfatal injuries and fatalities. Korea's disaster investigation standards (2000) designate lost workdays as a range of days, rather than a confirmed value. The severity of risk is, therefore, determined by dividing the number of lost workdays by the number of accident cases in a given work type.

Korea Ministry of Labor (2000) suggested 7,500 lost workdays for every single fatality accident. In Korea, the average age of laborers is 44.6~46.8 years old, and their average workdays may be 150 days in a year (Korea Ministry of Labor 2007). In this situation, 7,500 lost workdays for every single fatality accident can be excessive. Considering the average life span (75 years) in Korea and average workdays in a year, the average lost workdays for fatality accident can be calculated as 4,500 days((75years-45years)*150days). Therefore, the authors regarded 7,500 days excessive.

Moreover, Korea Ministry of Labor has uniformly used 7,500 days to calculate total lost workdays. Each of lost workdays of thirty and fifty is same as 7,500 days in this calculation method. The number of lost workdays per one fatality accident, therefore, is calculated by multiplying 150 days per year by the difference between 75 and the age at death of each case.

In the context of this study, severity of risk for a given work type was calculated by:

$$Severity_j = (NN_j \times MDN_j + NF_j \times MDF_j) / NA_j \quad (2)$$

where NN = number of nonfatal injuries for a given work type; MDN = mean days away from work due to nonfatal injuries for a given work type; NF = number of fatalities for a given work type; MDF = mean days away from work due to fatalities for a given work type, and NA = number of accidents for a given work type.

(2) Frequency

Repetitive occurrence is a noticeable characteristic of construction accidents (Abudayyeh et al. 2003). The frequency of accidents in a work type is derived from statistical analysis. Frequency of accidents is determined by exposure to potential hazards. Exposure is difficult

to assess because there is a lack of detailed information about the degree of worker exposure to hazards. Therefore, a statistical method based on accident cases is appropriate in this situation.

The frequency of accidents can be defined by dividing the number of injuries by the number of work-hours. This is a classical and established method, though calculating the number of work-hours remains difficult (Navon and Goldschmidt 2003). We can find how much risk is involved in a specific work, but it does not consider the number of workers employed for a given work type. To solve this problem, Baradan and Usmen (2006) adopted the number of workers employed for a given work type. Considering the difficulty of calculating the number of work-hours, this study chose to adopt Baradan and Usmen's method.

The proposed frequency of accidents per work type is calculated by dividing the number of accidents for a work type by labor input per work type:

$$\text{Frequency}_j = \frac{\text{NA}_j}{\text{WI}_j} \qquad (3)$$

where NA = number of accidents for a given work type, and WI = Worker input.

Based on the site condition value calculation, the conversion method is as

follows. First, the converted values that conform to the site condition value and case condition value are extracted from Fig. 3-5. Then, they are applied to Equation 3-3:

$$SI = 1 - \frac{|A' - B'|}{A'} \quad \text{(Eq. 3-3)}$$

where A’ = converted condition value of a current site; and B’ = converted condition value of a past accident case

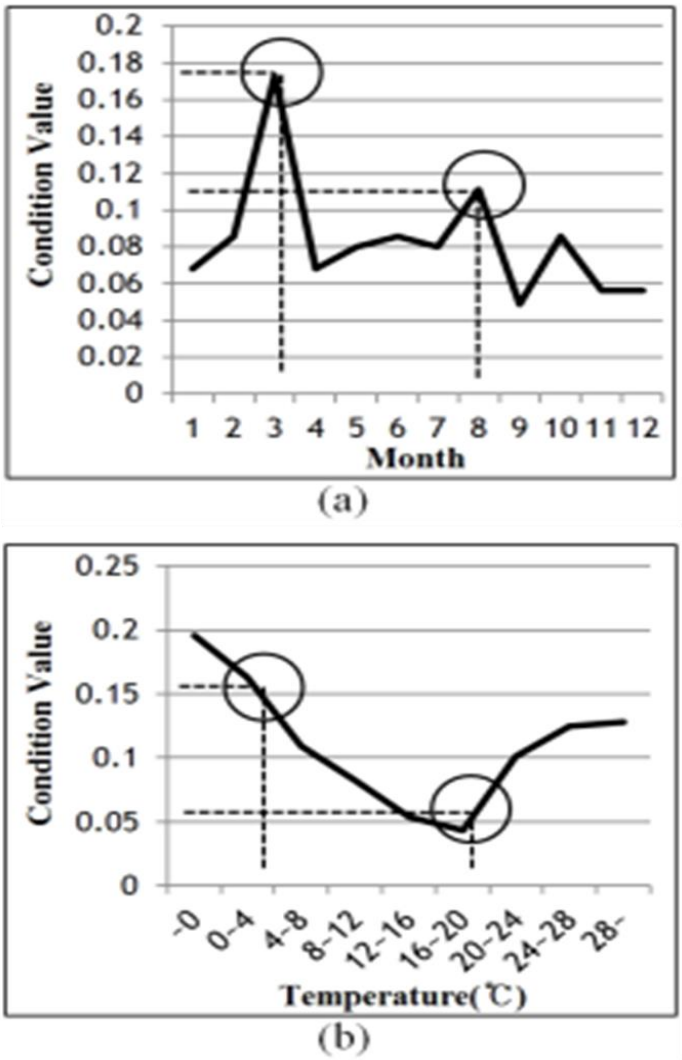


Fig. 3-5. Weight Graph: (a) Date; and (b) Temperature

Site condition values presented in Table 3-4 represents risk level (combination of severity and frequency) under a specific condition. The relative potential of an accident is defined as a value. Unlike the original use of site condition value in Lee et al. (2012a), site condition value of “date” and

“temperature” in this study are only used to determine the closeness of accident cases. For example, if the current site condition value is in March and a past case condition value is in August, the weight of each case becomes 0.173 (A’) and 0.111 (B’) respectively. In this case, the SI of the date is calculated as 0.642 by using Equation 3-2.

Table 3-4. Site Condition Values of Date and Temperature

Date		Temperature	
Classification (month)	Site condition Value	Classification (℃)	Site condition value
1	0.068	~0	0.196
2	0.086	0~4	0.163
3	0.173	4~8	0.109
4	0.068	8~12	0.082
5	0.080	12~16	0.054
6	0.086	16~20	0.043
7	0.080	20~24	0.101
8	0.111	24~28	0.125
9	0.049	28~	0.128
10	0.086	-	-
11	0.056	-	-
12	0.056	-	-

After calculating the similarity index of each influence factor, the similarity score (SS) was determined. This value can be expressed as the sum of the multiplication of the similarity index (SI) and weight (M), as seen in Equation 3-4:

$$ss = \sum_{i=1}^{n} M_i SI_j \qquad \text{(Eq. 3-4)}$$

3.3 Data Source for System Development

As shown in Fig. 3-6., there are two data sources used to identify activity oriented hazards. One is project management information system (PMIS) that has abundant information to manage construction projects (Chitkara 1998). Information from PMIS is used to generate queries. Information (e.g., work type, work time, and workers, etc.) is converted to indices and then combined to form a query. The generated query is entered into the retrieval system. After retrieval system searches through the accident case database, the result (related past accident case) is given to the user. However, PMIS lacks information about extracted-factors-related coordination and objects. If retrieval is performed by a query generated based on combination of indices from PMIS, the result does not include information where an accident occurs. In this case, safety managers cannot use the retrieved result effectively.

This problem can be solved by using BIM programs. Commercial BIM

programs are designed to be able to add properties of an object. Two of the most widely used programs are Revit (Autodesk) and ArchiCAD (Graphisoft) (Khemlani 2007, Hamil 2013). Geometric properties—such as coordination, volume, and length—can be easily extracted. However, the developing tools provided in Revit or ArchiCAD, such as geometric description language (GDL), are necessary to program objects that contain various types of information suggested in this study. Nevertheless, these developing tools are not commonly used in the design process. Although these limitations have been overcome, data processing capability is overloaded when abundant information enters the objects' property value.

Therefore, the structure of the data source in this study incorporates the benefits of PMIS and commercial BIM programs. Lee et al. (2012b) demonstrated the possibility of combining and linking information between PMIS and influence factors. Navon and Kolton (2006) and Lee et al. (2012b) suggested a model combining schedule and safety information based on AutoCAD or BIM. Based on these methods, geometric values are extracted (using the extraction function of BIM) and connected with the information from PMIS. Through this type of system composition, users can save, handle, and use information at their discretion. Also, property values can be fully used without decreasing the efficiency of commercial BIM programs.

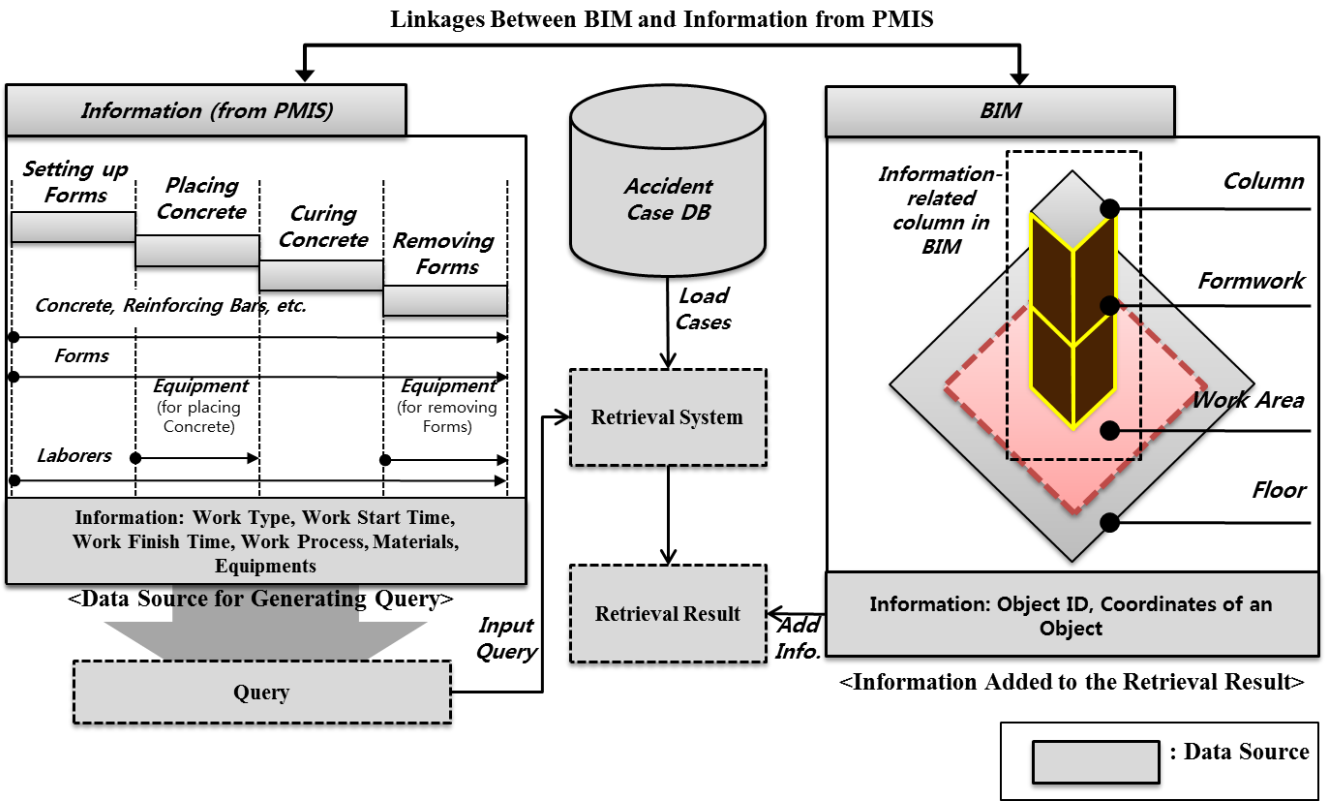


Fig. 3-6. Data Source for Information Retrieval

3.4 Information Retrieval System Framework

The retrieval system framework developed by this study is shown in Fig. 3-7. The system consists of input, data processing, and output. The input has information about factors extracted from indices for the information retrieval section, weight of each factor, and accident case database. The number of influence factors differs from safety planning phases and construction site conditions. The weight of each factor's database has weight values acquired through AHP analysis. These values are used as indices of weight in an information retrieval module. The accident case database is utilized to search similar accident cases by comparing queries based on construction site conditions.

Data processing comprises the BIM module, extraction module, and information retrieval module. The BIM module forms 3D oriented parametric models. It has the geometric property values of objects, and provides objects that include additional property values. The extraction module extracts objects from BIM. The export function in commercial BIM programs is used as an extraction method. Finally, extracted objects and information about influence factors are combined. Information processed in the extraction module is sent to the information retrieval module and used to find similar accident cases.

The most relevant case for each worker can be found by this type of query generation. PMIS including information related to workers is combined with

BIM objects. This combination makes a query set. In this process, property values of a worker category generate a suitable query for each worker. For example, the crews that work on the same activity have the same values of work and work condition. However, an individual worker's property values—such as age, number of safety trainings, and work days on current site—vary. These property values are incorporated into the generating query process. The generated queries include not only characteristics of work and work condition but also characteristics of individual worker.

The information retrieval module determines the type of retrieval method by the number and degree of BIM property values. After that, AHP result values are loaded by the type of retrieval method. Also, queries are generated and converted to internal representations. Based on query sets by the safety planning phase, the system performs retrieval of similar past accident cases. Similarity indices are calculated by comparing accident cases with internal representations based on construction site values.

The output represents similar past accident cases retrieved by query sets of each safety planning phase. The similar accident cases classified by the number of inputted factors can be found. Each case provides the coordination and remaining period of risk factors by combining objects extracted from BIM. Retrieved past accident cases are extracted by the property values connected with a BIM object. The object connected with retrieved past accident cases includes information consisting of worker work type, work

start time, work finish time, coordinates of work area, and etc. as property values. The suggested system searches for the workers whose work type or work conditions correspond to the retrieved past accident cases. Then, these cases will be provided to related workers. The details of the information push process are described as follows. The BIM object including geometric information and additional properties has an identifier. Thus, the worker ID in additional properties can be linked with an object's identifier. The identifier is used to search a worker / workers who has/have relevance to the object. If there is a related worker, the system provides retrieval results to him / her.

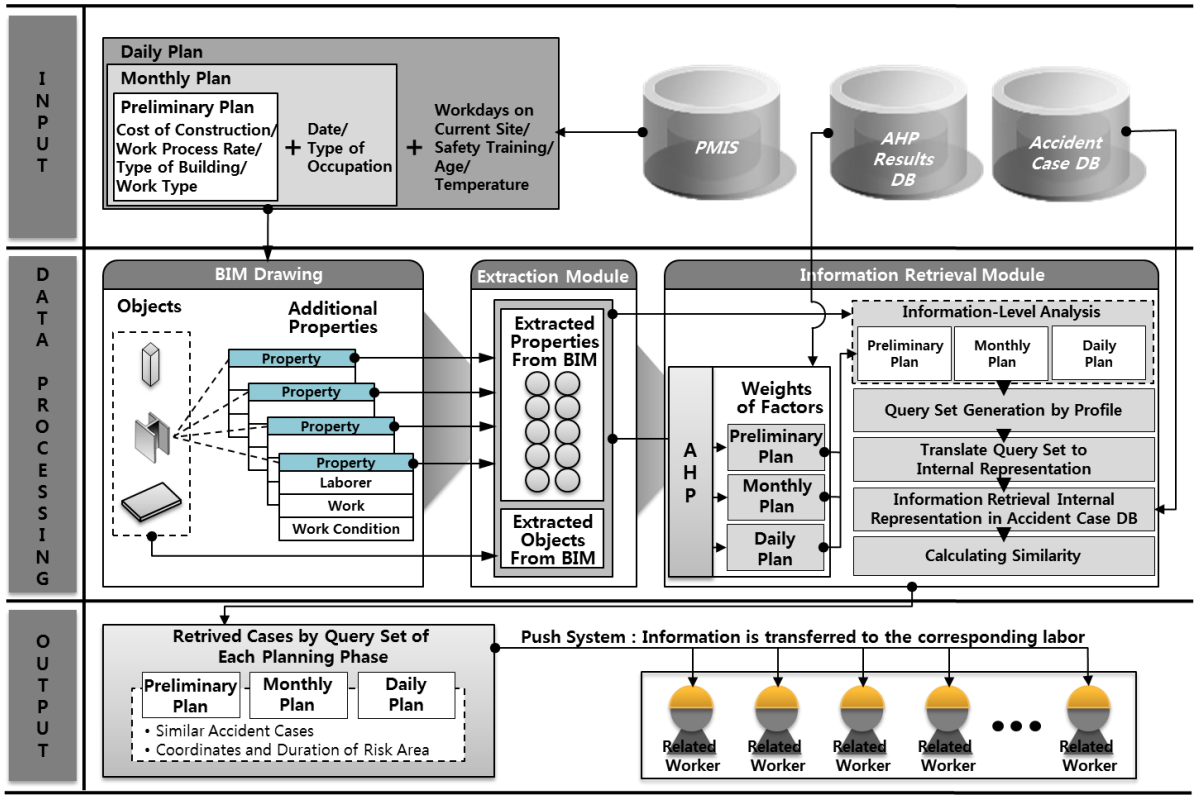


Fig. 3-7. Retrieval System Framework

The suggested retrieval system framework is developed in a computer program written in Visual Basic for Application (VBA), and BIM software (ArchiCAD). The algorithms that perform information retrieval and deliver retrieved past accident cases to related workers are written by VBA in MS-Excel. Linkages between PMIS and BIM objects are also programmed by VBA. ArchiCAD is used in model building, and extracting data about the building elements' properties. These data are used to generate queries after combination with PMIS data. The system is operated in MS-Excel, and the results are shown in the "Hazard Identification System Validation using Case Study" chapter.

3.5 Summary

Learning from past events provides a lesson for someone working in similar circumstances. Among past events, failures and accidents are the strongest stimuli, so many researchers have studied how to apply accumulated past accident cases. Several research studies have been performed by retrieving accident cases based on the repetitive occurrence of accidents. These studies can improve safety management by extracting accident cases. However, they do not include how to meet the demands of end-users (workers), provide retrieved cases to workers, reflect safety-planning steps, and identify the coordination of risk factors.

Therefore, this section develops an automated accident case retrieval system, which can identify activity oriented hazards on construction sites. To apply this system to establish safety management plans (including preliminary plan, monthly plan, and daily plan), query generation process considers the characteristics of each safety management phase. Based on this, each query is composed by different number of indices, and this composition can make a query reflect information level of each safety management phase.

After generating queries, the suggested system searches the most similar past accident case. Queries that represent various situations are automatically inputted to the retrieval module. The result includes time and geometric information, as well. Also, they are automatically provided to the workers via the push system.

To develop the suggested information retrieval system, BIM, PMIS, AHP result database, and the accident case database were structured. Then, information retrieval algorithms were defined. Finally, the push system was established to provide retrieved accident cases to the related workers.

Chapter 4. Non-Activity Oriented Hazard Identification

Although hazards on activity areas can be identified by the suggested activity oriented hazard identification system, hazards on workers' movement paths still remain. Non-activity oriented hazard identification has been usually performed by human observers. The dynamic changes within construction sites also make it hard for safety managers to identify the generation and extinction processes of hazards on the movement paths. Current hazardous area identification methods are generally performed by safety managers, and require additional efforts (e.g., more managers, time, and cost). Also, these methods as performed by human observers have difficulty reflecting the dynamic changes of construction sites because human observation for identifying a hazardous area should be performed whenever a project's condition or progress changes.

Therefore, this dissertation proposes a hazardous area identification model based on the ADC, which facilitates the collection and analysis of raw data on construction sites. It identifies potentially hazardous areas (PHA) on workers' path by using the RTLS of workers and a building information model (BIM). The suggested system identifies PHA by using the deviation between the optimal route (the shortest path), which is determined by extracting nodes from BIM objects, and the real movement path of a worker, which is tracked

by radio frequency identification. After this process, PHA can be divided into uncontrolled PHA and already controlled PHA. The information about uncontrolled PHA is provided to safety managers so they can establish proper safety countermeasures and manage the area.

This chapter will provide the detail process how to establish the non-activity oriented hazard identification system based on utilizing the concept of the ADC.

4.1 Conceptual Model

Before developing a hazardous area identification model based on the usage of RFID, we suggest two hypotheses about workers' moving characteristics. They are as follows:

(1) When workers move to where they want to go, they have a tendency to move along an optimal route to minimize physical work (Dorigo and Gambardella 1997) or time (Mitropoulos et al. 2009). In this hypothesis, "Optimal Route" means the shortest path between two places—where workers are coming from, and where they are going (Rilett and Park 2001).

(2) When obstacles (activities, material, hazards, etc.) exist on an optimal route, workers deviate from the route.

The conceptual model suggested in this study is shown in Fig. 4-1. When

workers move to work spaces, they select the shortest path (Dorigo and Gambardella 1997). If their actual path deviates from the optimal route, there may be an obstacle hindering their movement along the path. These obstacles can be piled-up materials, working spaces, or hazardous areas. If the deviation between workers' actual paths and the optimal route is calculated by using a real-time location system (RTLS) and ADC, safety managers can easily identify hazardous areas.

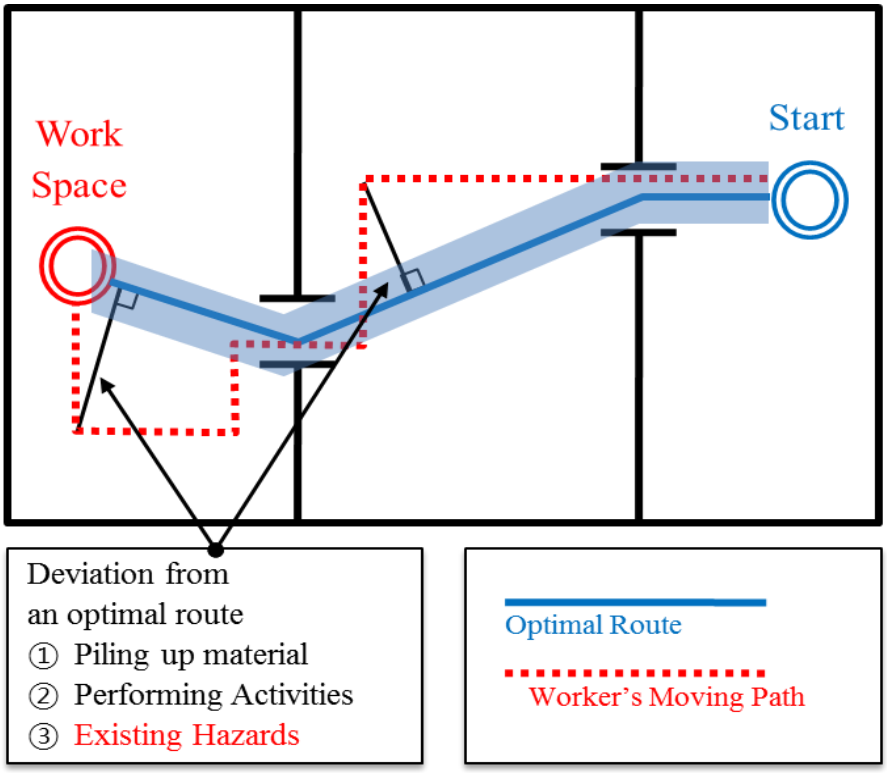


Fig. 4-1. Conceptual Model for Hazardous Area Identification

4.2 Requirements for Developing Non-Activity Oriented Hazard Identification

To develop the suggested conceptual model as a system, various requirements should be demanded. This section starts with the verification of assumptions, and then required databases that are needed to realize the suggested system are explained.

4.2.1 Verification of Assumptions

To validate the hypothesis and the suggested conceptual model, two experiments are performed: 1) workers move following the optimal route (the shortest path); and 2) when an obstacle is on the optimal route, workers make a detour to avoid the obstacle. The test site is an underground parking lot where are ongoing construction activities. There are columns, equipment and other piled materials.

By observing workers' movement, workers do not recognize columns as an obstacle. When a worker wants to move from a point to other point, workers usually follow the shortest path between two points. Fig. 4-2 shows the experiment site and result. In the non-obstacle situation, 94.1% (64 persons) of the 68 workers moved along the optimal route. And when an obstacle was on the optimal route, 85.7% (42 persons) of the 49 workers took a detour to avoid it.

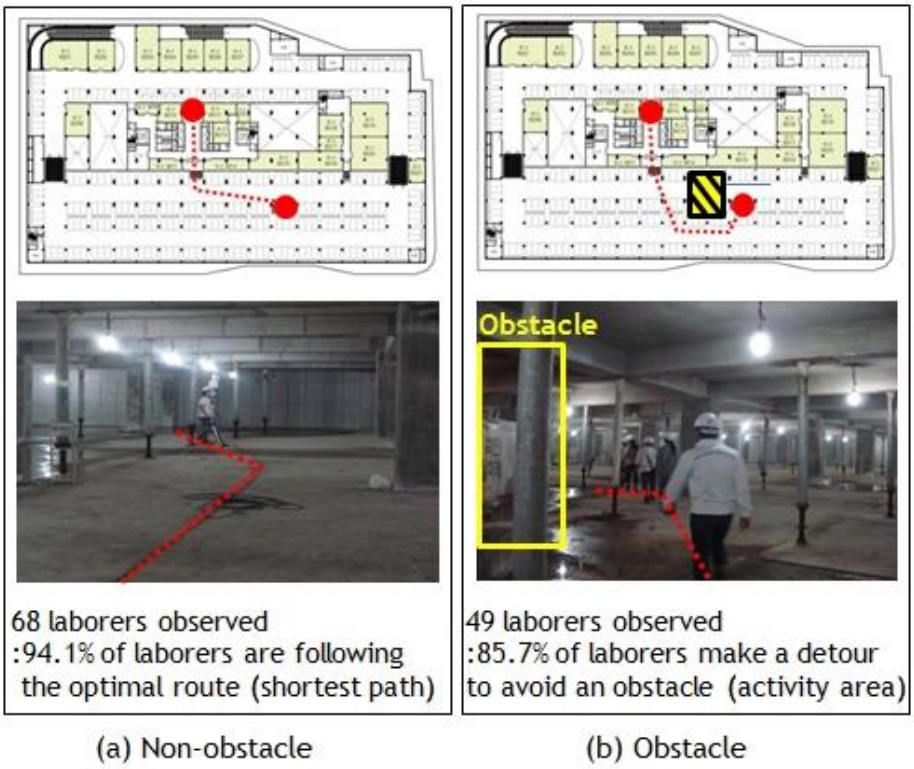


Fig. 4-2. Movement of Worker in Non-Obstacle and Obstacle Conditions

By performing this experiment, workers go through the shortest path if there is not an obstacle. Workers' movement characteristics about an obstacle can be incorporated into the suggested conceptual model to develop a non-activity oriented hazard identification system.

4.2.2 Required Databases

To develop the suggested system, algorithms which represent the suggested concept are needed. However, diverse type of data is needed to operate algorithms. In the non-activity oriented hazard identification system,

six databases (e.g., BIM drawings, workers' moving characteristics, optimal route, workers' location log, activity area, and identified hazardous area) are needed. The detail contents of each database are explained below.

(1) BIM Drawings

The role of BIM drawings is to provide basic information to the system. These drawings extract objects to check the interference in the process of determining the optimal route. It includes the order of activities and object properties. The order can represent the dynamic changes on a construction site. This database provides the shape of each floor to the suggested system and also represents the coordination of other databases.

As shown in Fig. 4-3, "BIM Drawings" DB takes a role of providing 4D-based information. The kinds of information are floor plan (2-dimensional), geometric information of an object (object type, coordinates of x-, y-, and z-axis), and schedule information of an object (activity start time and activity end time). Object type is extracted from object property and used to identify a node element that is needed to determine optimal routes. Geometric and schedule information of an object are used to determine activity areas. An activity area is determined by geometric coordinates of an object. An area within a specific range from an object is considered as a work area. The floor plan information is used to present identified hazardous areas in the monitoring phase. The information provided by

BIM used in this study is almost numeric or text type (e.g., time, coordinates, object type) except floor plan outlines.

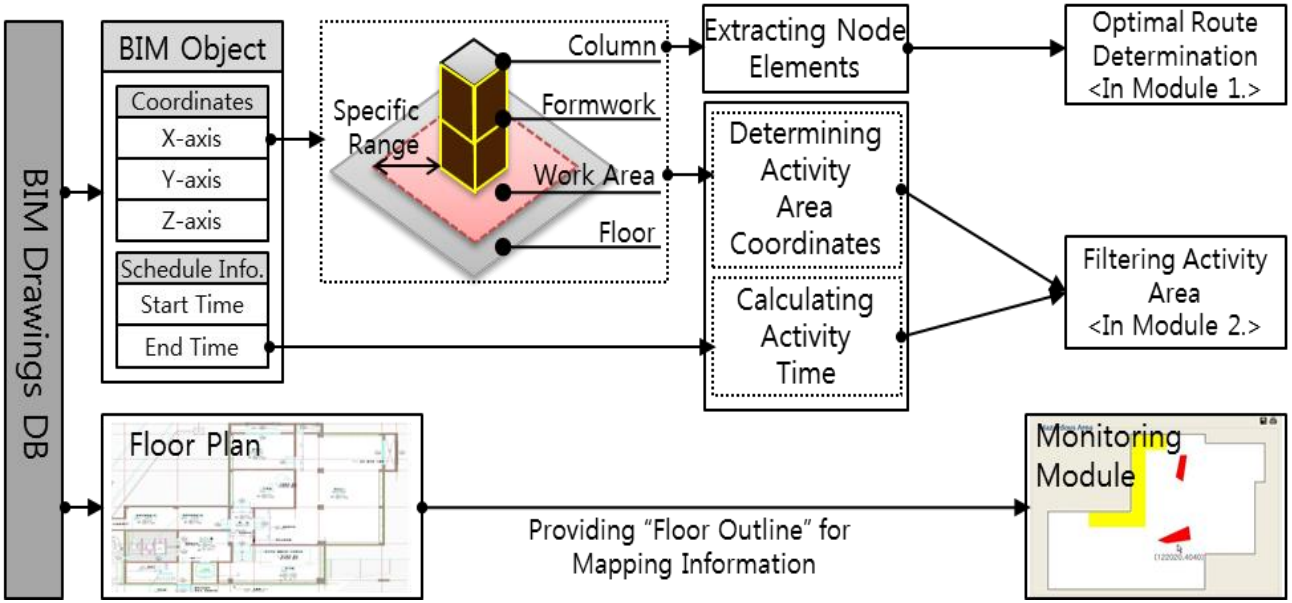


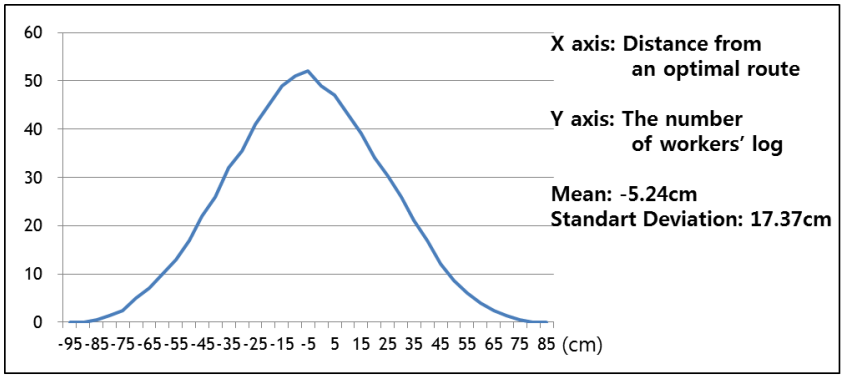
Fig. 4-3. The role of BIM Drawings in Identifying a Hazard on Path

(2) Workers' moving characteristics

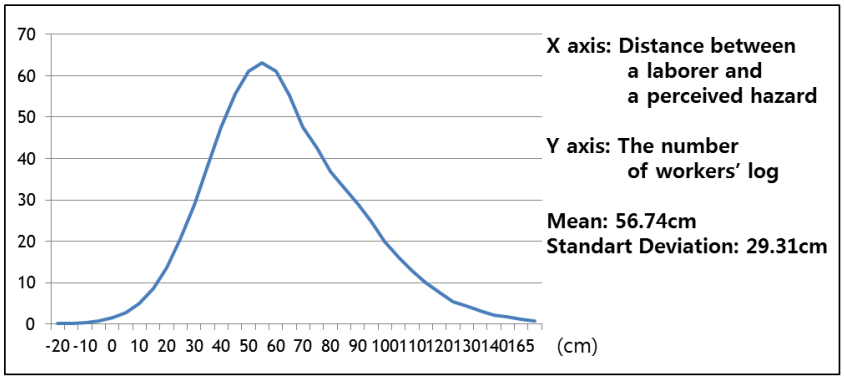
Workers' moving characteristics about an obstacle are analyzed. However, a worker cannot move like a vehicle. Although a worker recognizes an optimal route, it is natural to say that he/she follows the optimal route in a specific range. Because an optimal route is a very narrow line, a worker can get slightly out of the narrow line. If this slight deviation is determined as an error, there must be enormous errors that can decrease the suggested system performance. Therefore, this dissertation determines that this slight deviation is not the proof for workers' deviation from the optimal route. As Fig. 4-4 indicates, there are two kinds of data: the first data ($\boldsymbol{d_a}$) represents the distance from the optimal route when workers move along the route, and the second data ($\boldsymbol{d_b}$) demonstrates the distance from the hazard when workers recognize it.

To obtain the results, two experiments were examined. Thirty-seven workers with helmet-mounted RTLS equipment were examined to calculate the first data. In this experiment, more than 95% of the locations logged were within the range of -54.28cm to +50.66cm (- means left side and + means right side). To apply these results to the system, the left and right widths are determined to be ±50cm. The second experiment proceeds like the first one. The mean of the results, 56.74cm, is used in the unidentified hazardous area identification module. To summarize these experiments, the workers are usually in

the range of 50cm from optimal routes. If they recognize a hazard, they would be out of that range.



(a) Distance from an optimal route



(b) Distance between a worker and a perceived hazard

Fig. 4-4. Workers' Moving Characteristics

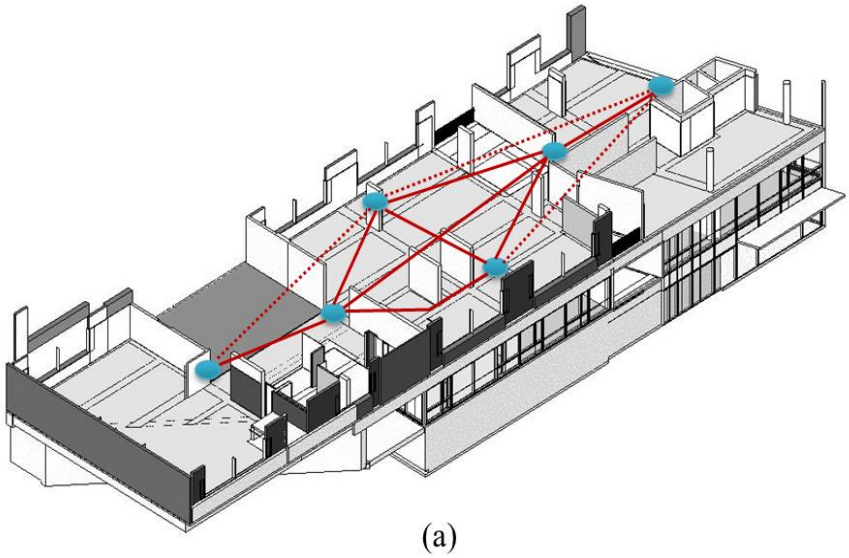
(3) Optimal route

The definition and determination of an optimal route are explained above (the first part of the unidentified hazardous area identification algorithm). An optimal route includes the coordinates of nodes and a set of coordinates between two nodes. An optimal route is stored not

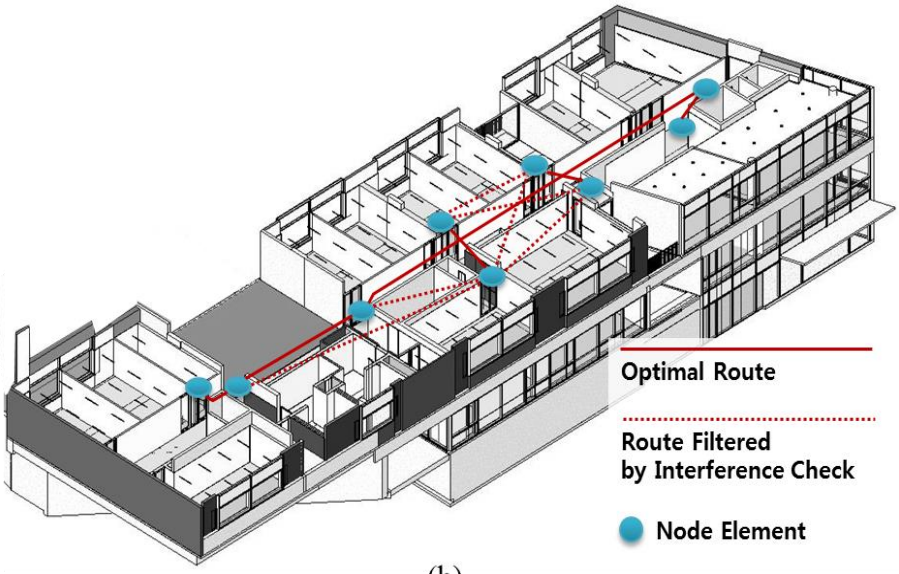
as a graphical result incorporated into BIM, but as a set of coordinates. This set is used to search workers' location logs straying from an optimal route. The tested site (see Fig. 4-5.) has a simple floor plan. When the suggested system is applied in a more complex environment, the system can be used same way as a simple floor plan. However, there are more objects and activities than a simple environment, which can make an environment change more dynamically. The dynamic changes that take place on a construction site constitute a definite problem for determining the shortest path. Because the condition of a construction site changes over time, the shortest path also changes due to newly installed objects.

This problem can be dealt with by two methods. First, an activity schedule (4-Dimension BIM) including the installation order of each object and its activity information is stored in a BIM model. Based on this information, the shortest path can be determined that reflects the dynamic changes. Second, each activity has its own work area, and a specific area around an object that is needed as a work area when the object is installed is applied for the path finding. By defining a specific area, contemporary equipment or many temporary construction objects that are difficult to represent in a BIM model can be considered in the suggested system. Fig. 4-5 illustrates a graphical example of how to find paths reflecting the dynamic changes of a construction site. Fig. 4-5(a) shows a situation immediately after the

completion of a structural activity and when there are only several bearing walls. In this case, the shortest path is determined by nodes that consist of an area where an activity is performed and an entrance to a stairwell. Fig. 4-5(b) presents a situation during interior finishing activities. Because interior walls and columns are installed, nodes mainly consist of doors to rooms.



(a)



(b)

Fig. 4-5. Example of Optimal Routes

(4) Workers' location log

The location tracking method used in this study is RFID-based RTLS. It is a type of local positioning system that allows a user to track the location of objects in real time. RFID-based RTLS uses radio-frequency waves to track objects. As shown in Fig. 4-6, the RTLS reader receives signals from a tag attached to the worker–in this dissertation, all the tags are attached to helmets. These signals are sent to base-stations on each floor. The signal is measured by the type of time of arrival measurement that uses 2.4 GHz communication. All of the information generated in this process is stored for later use in the unidentified and filtering hazardous area modules.

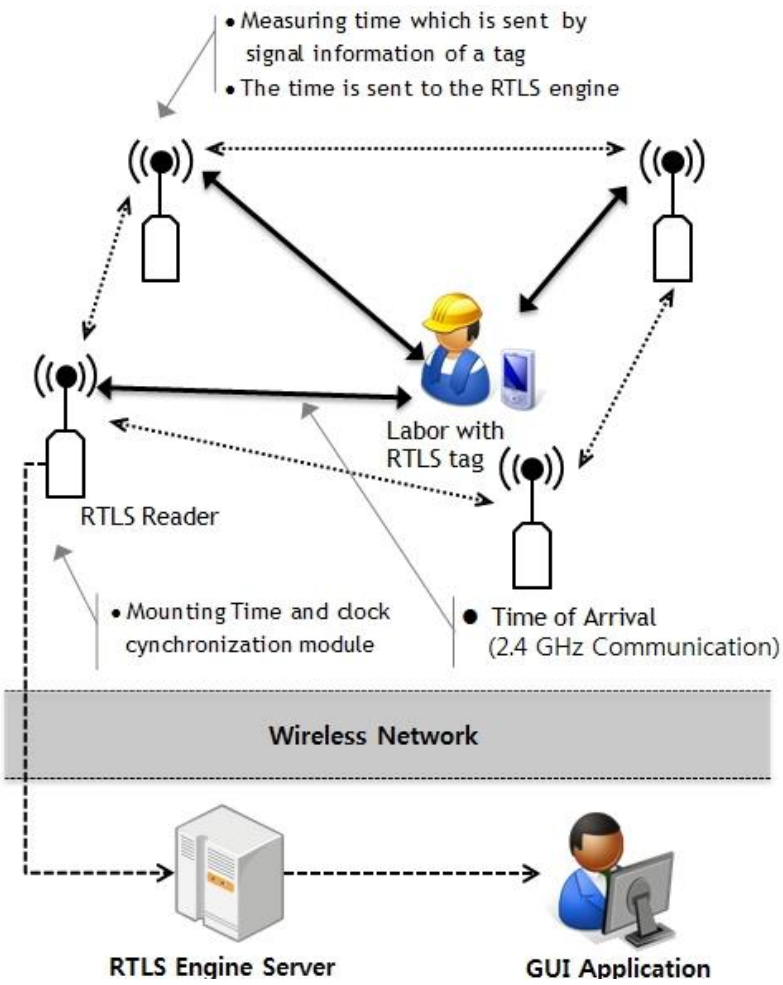


Fig. 4-6. Overview of Real-time Locating System

(5) Activity area

To perform an activity, a set of work spaces is needed. In a construction site, spaces—where activities are performed—change over time. Akinci et al. (2002) proposed a method for generating the activity area of a BIM object. They investigated areas for performing

an activity, and integrated the areas and BIM objects. Choi et al. (2004) suggested a method that can identify a certain work area based on consideration of work schedule (time).

This study adopts the method suggested by Choi et al. (2014) to identify the activity area of each object because this method can provide the more specific coordinates of an activity area. The coordinates of an identified activity area are used as a node for determing optimal routes. It is also used to filter potential hazardous areas. The activity area DB stores areas where work is accomplished over time. This data and the BIM drawings are combined, and are used in the filtering hazardous area algorithm.

(6) Identified hazardous area

This includes three types of data—hazardous areas searched by safety managers, hazardous areas generated by the unidentified and filtering hazardous area algorithms, and hazardous areas eliminated over time.

Considering the characteristics of each database and its required data, location logs are mainly used to utilize the suggested system. Therefore, it is considered that the accuracy of location sensing is the key for successful system development. Next section will introduce an accuracy improvement method for workers' location sensing.

4.3 Accuracy Improvement Method for Workers' Location Sensing

This section presents the development of a radio frequency identification (RFID)-based real-time locating sensing system for improving accuracy of workers' location tracking. In this dissertation, identifying hazards on movement is mainly performed based on a location tracking technology which also greatly affects the accuracy of the non-activity oriented hazard identification system. However, the accuracy of RTLS remains poor in a building construction project where many activities are conducted indoors. Construction sites are very difficult to maintain signal availability because there are lots of obstacles. To improve the accuracy of RTLS in building construction projects, this section suggests a location tracking error mitigation algorithm and presents the concept of using assistant tags.

4.3.1 Location Tracking Error Mitigation Algorithms

In an RFID-based RTLS, a tag repeatedly emits radio frequency signals and a reader receives these signals. The location of the tag is estimated with this signal flight time from the tag to the reader. A tag transmits omnidirectional signals, and can make both direct and reflected paths. Accordingly, the flight times of signals emitted from a tag at a certain moment can vary depending on the type of signal path. For example, as shown in Fig. 4-7, distance of the direct path (D_1) is shorter than that of the reflected path (D_2). In this case, a reader receives two different signals, and the difference can cause an error. If the reader receives a signal conveyed through the

reflected path, the reader estimates the location of the tag based on the distance of the reflected path (D_2) that is longer than the actual distance (D_1). Thus, if the tracking signal through the direct path is not successfully delivered, a reflected path can cause an estimation error as much as the distance difference between the direct and reflected paths (D_2-D_1).

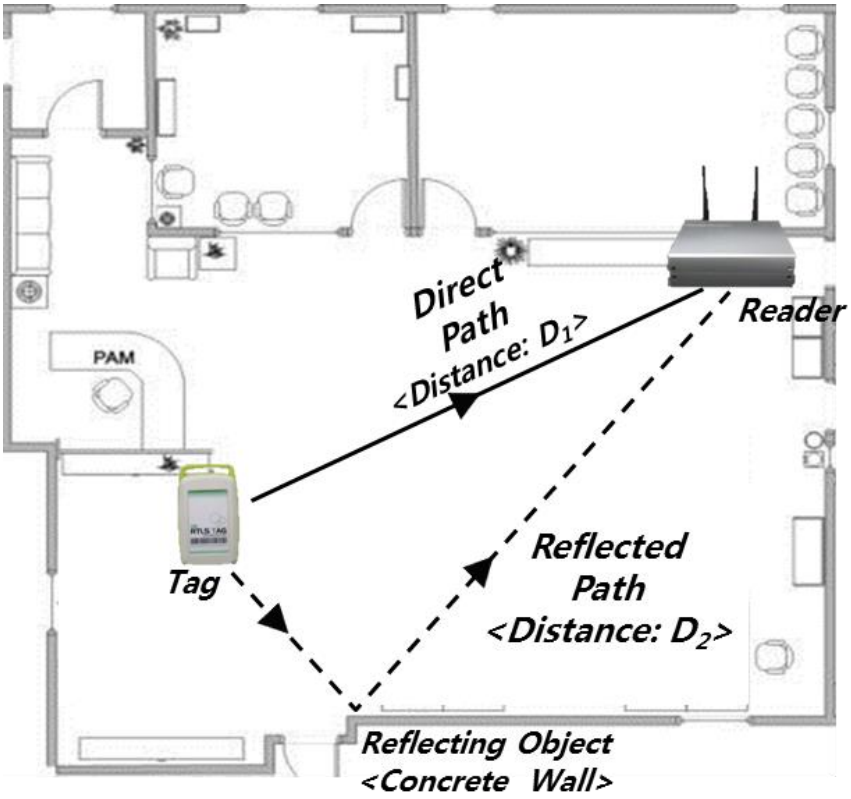


Fig. 4-7. Example of Location Tracking Error Caused by Multi-path

Generally, moving objects have their own movement characteristics in terms of direction, range and velocity. Effective use of these characteristics of

a target object can afford valuable information for filtering out location estimation errors, and thereby improve tracking accuracy (Mullen 2005; Son et al. 2013). Consequently, some notable research has applied the movement characteristics of a target object for location-sensing (Lei et al. 2013; Son et al. 2013). For example, Son et al. (2013) introduced an enhanced asset locating system for vehicle pooling in a port terminal by using the information on the range and speed of a transfer crane.

Since construction equipment and workers also have their own movement characteristics (e.g., operating range, velocity, and path), the moving characteristics of a target object in construction projects are expected to improve the accuracy of real-time location tracking. Use of possible moving range information is the simplest way to detect location estimation errors. If the arrow line is a worker's actual path and the area between the two dotted lines is a possible range where the worker can move (see Fig. 4-8), application of the information of possible moving range can easily detect the four points (P1, P2, P3, and P4) as location estimation errors since they are located outside the possible range.

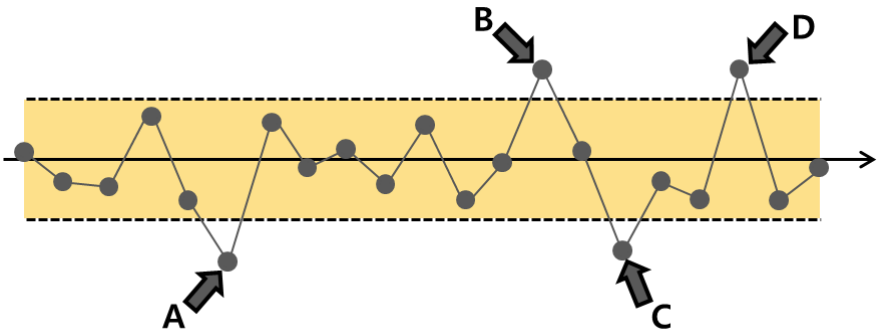


Fig. 4-8. Elimination of Tracking Errors Using Possible Range Information

As such, the possible moving range information is helpful to improve the accuracy of a location-sensing system. However, this kind of information is not very effective for detecting localization errors within the range. Furthermore, it is tedious and time consuming to define and update the information of all possible moving ranges, particularly in a construction project where possible moving range can vary significantly as the project progresses.

Acknowledging these challenges, this paper proposes the concept of reducing location estimation errors by calculating distances between two consecutive logs (i.e., signals). As aforementioned, a target object moving in a construction site has its own moving speed range. For example, workers can move at a speed of approximately up to 70cm/sec. Fig. 4-8 illustrates the concept how to determine an error. A worker cannot move more than their each maximum distance calculated based on worker's maximum velocity and

time (time difference between two consecutive logs). In Fig. 4-9, $n+1_{th}$ location log is within the range and can be determined as a proper log. However, $n+2_{th}$ location log is located on outside of the range. The $n+2_{th}$ location log means that a worker moves beyond his/her maximum speed, so that this log can be determined as an error.

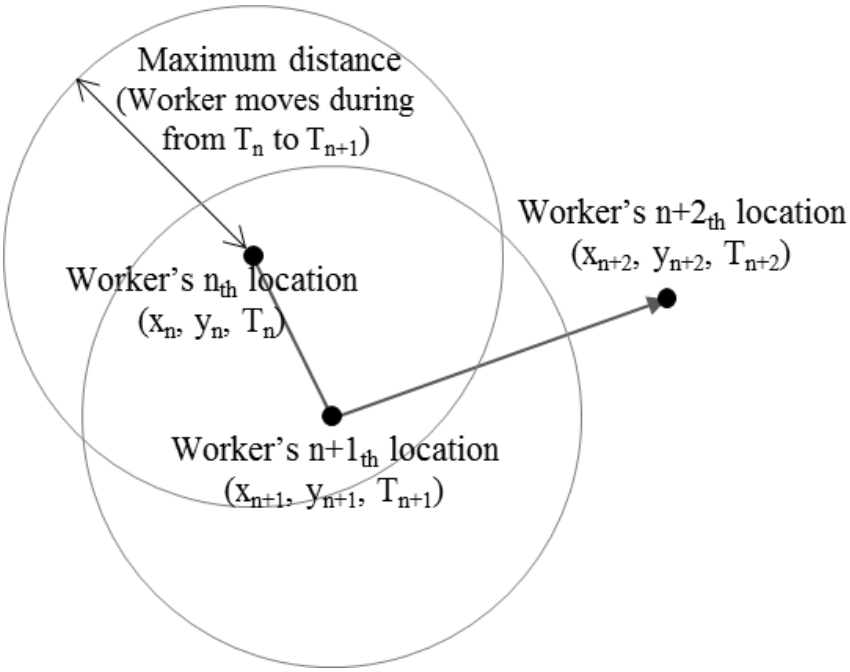


Fig. 4-9. Concept of Error Log Determination Method

Fig. 4-10 presents a procedure how the suggested RTLS calculates every log. At first, the distance and time difference between two consecutive logs (n_{th} and $n+1_{th}$ log) are calculated. Based on this information, the moving velocity between log_n (defined by x_n, y_n, z_n, t_n) and log_{n+1} (defined by x_{n+1}, y_{n+1}, z_{n+1}, t_{n+1}) is obtained using Eq. 4-1.

Moving Velocity

$$= \frac{\sqrt{(x_{n+1}-x_n)^2+(y_{n+1}-y_n)^2+(z_{n+1}-z_n)^2}}{t_{n+1}-t_n} \quad \text{(Eq. 4-1)}$$

Then, if the moving velocity between a log (log_n) and the next log (log_{n+1}) exceeds the maximum velocity of a target object, the next log is regarded as a localization error and is eliminated in the locating process.

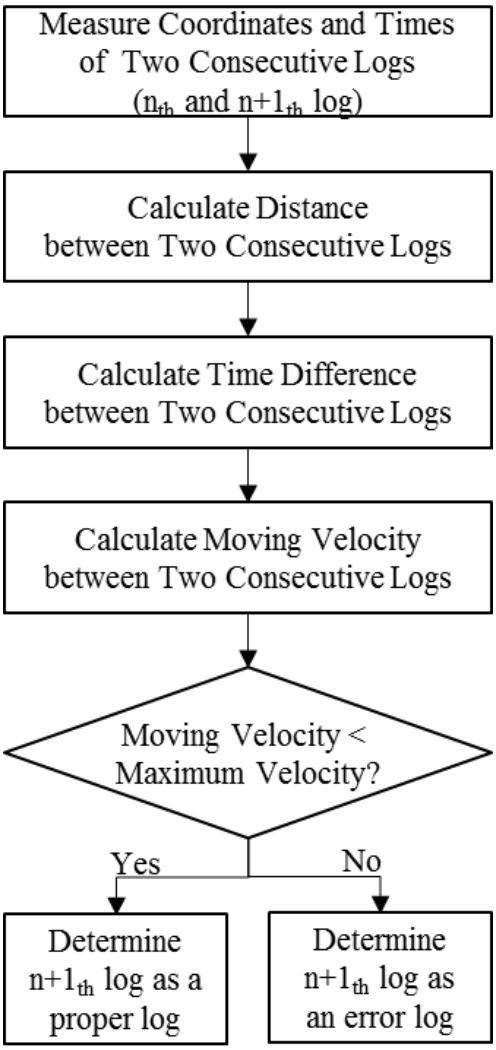


Fig. 4-10. Procedure of Error Determination Method

Although an error can be identified by the aforementioned method, there is still a problem which is caused by serial errors. More than two consecutive error logs cannot be dealt with the suggested method. Location logs can include many errors because construction site conditions can intensify signal

attenuation and multi-path. Therefore, error logs are estimated sequentially and serial errors should be eliminated. As shown in Fig.4-11, log_{n+1} is eliminated because this log is a certain distance away from log_n (exceeding the maximum velocity). Then, the locating system calculates the moving velocity between logn and log_{n+2}. By repeating this filtering out process, logs that are measured as moving faster than the maximum velocity are eliminated. The suggested process shown in Fig. 4-11 may provide too few valid logs if many logs are filtered out as errors. However, more than ten thousand signals are transmitted per second, which provides enough logs for locating a moving object even when many logs are identified as errors.

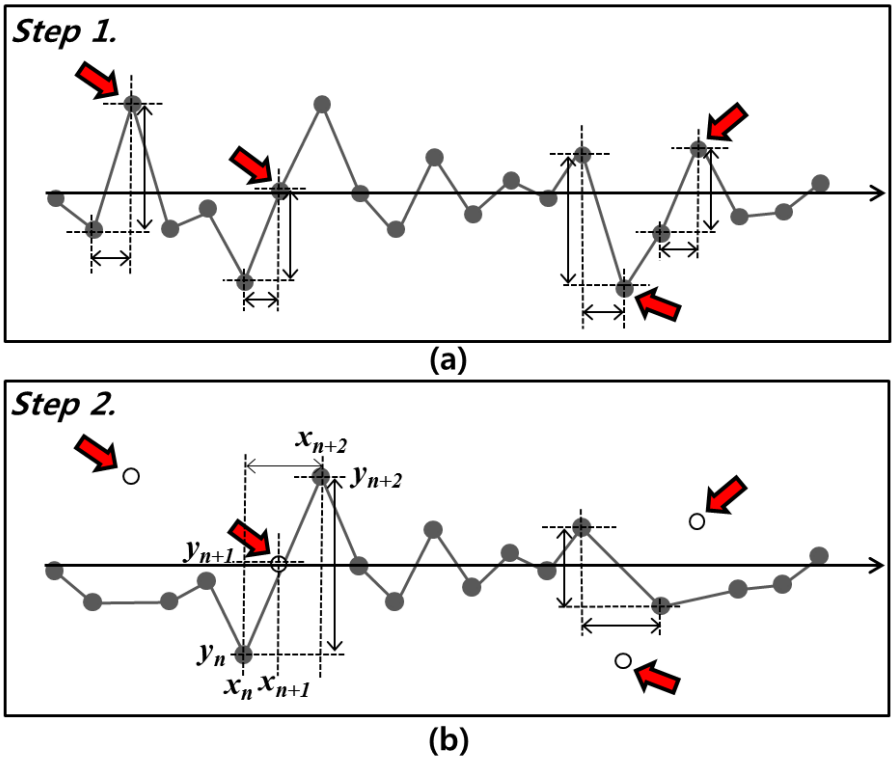


Fig. 4-11. Elimination of Tracking Errors Using Maximum Distance Information

Time synchronization between tags and readers is important when applying this approach for filtering out location estimation errors. Since the suggested approach determines the location estimation errors based on the moving velocity of a target object, a slight time difference between a tag and a reader may result in significant location estimation errors. In fact, an RFID-based wireless network is sensitive to this type of error since the network transmits a huge number of signals for location tracking. To address this issue,

several time synchronization protocols that have been suggested for wireless sensor network include reference broadcast synchronization (RBS) (Elson et al. 2002), timing-sync protocol for sensor network (TPSN) (Ganeriwal et al. 2003), and flooding time synchronization protocol (FTSP) (Maróti et al. 2004). TPSN was selected as the time synchronization protocol for our study because of its effectiveness in addressing the propagation delay time that can often occur in a construction site.

The most important issue of the suggested method is how to determine whether or not an initial log is valid. If the suggested method is directly applied when a log is measured as an error, the second log has high potential to be measured as an error, as well. To solve this problem, a method to identify a valid initial log is required.

Although several sequential logs are determined as error logs, location of a log after these logs can be determined as a valid log. Considering construction site conditions, both valid logs and invalid logs coexist since there are attenuation and reflection of propagation. In this paper, 13,240 logs are analyzed to determine a specific log as a valid initial log.

If all six sequential logs after a specific log are not determined as error logs—all six logs are determined as error logs, the specific log can be considered as valid log that is within a permissible error range. In the other word, if all six sequential logs after a specific log are determined as error logs,

the specific log can be considered as an invalid log. In this case, the next log after the specific log is examined to identify an initial log. Fig. 4-12 illustrates the suggested algorithm for identifying a valid initial log.

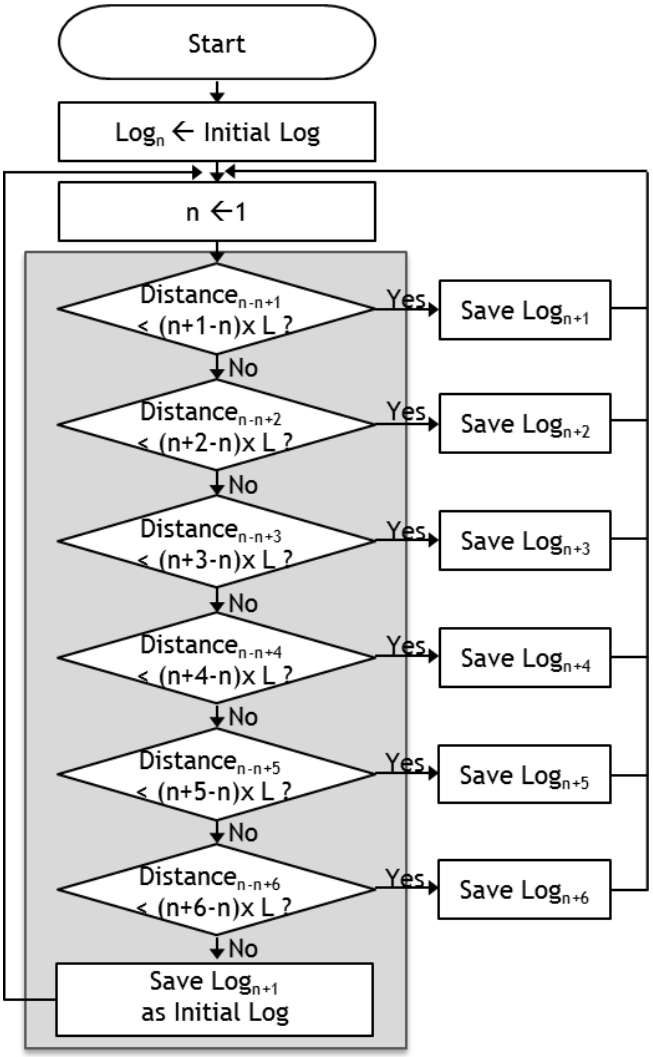


Fig. 4-12. Initial Log Determination

4.3.2 Application of Assistant Tags

In order to locate a target object using an RFID tag, at least three readers should receive signals from the tag. Since an RFID tag sends signals omni-directionally, it is difficult to accurately locate the tag with less than three readers. Thus, successful tracking of a moving object requires careful determination of the readers' installation location so that at least three readers can receive the signals from a tag wherever it is in the tracking environment.

Obstacles between a tag and readers may cause signal attenuation and multi-path, thus preventing successful communication between the tag and the readers. Multi-path is a propagation phenomenon in which two or more paths of radio signals reach a reader due to reflection from walls or floors (Rappaport 1996). Signal attenuation is a gradual loss in radio signal intensity as it propagates through space due to many effects including free-space loss, diffraction, and absorption (Rappaport 1996).

Suppose that four readers are installed in each corner of the room but only two readers (readers C and D) receive signals from the tag and the other two (readers A and B) do not due to an obstacle (Fig.4-13). Thus, while more than three readers are installed in this case, it is still difficult to accurately locate the tag since only two readers receive the signals from the tag.

To address this problem, Lee et al. (2012) suggest an idea of assistant tags

that are fundamentally similar to reference tags but differ in terms of their installation location. Lee et al. (2012) suggest attaching assistant tags to workers' safety helmets, instead of on floors. Hence, tags are not installed on some pre-fixed places on floors but attached to moving workers, which significantly increases the accuracy and applicability of RTLS. In the example shown in Fig.4-13, an assistant tag works as a virtual reader. That is, three readers (readers A, C and D) receive the signals from the assistant tag and the moving location of the assistant tag is identified. Then, the assistant tag and the two readers (readers C and D) can communicate with the tag in the shadow area. The location of the tag in the shadow area can thus be identified by using an assistant tag.

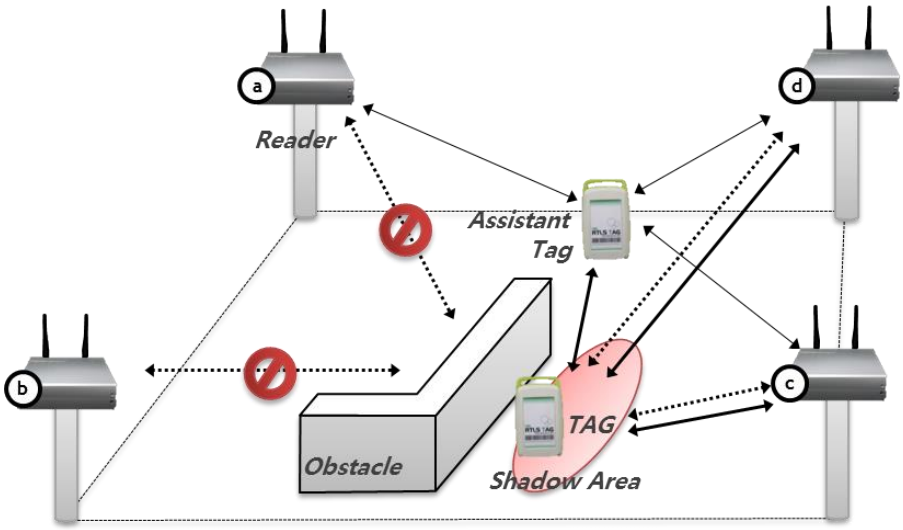


Fig. 4-13. Assistant Tag for Tracking a Target Object in a Shadow Area

However, their method has two main limitations. First, they did not suggest method to determine whether the environment is LOS or NLOS. Thus, a tag that is not in a shadow area continuously communicates with an assistant tag. If there is not any environment determination method, the RTLS engine can be influenced by multi-paths from assistant tags and other readers. As a result, the accuracy of the location sensing decreases. Moreover, a reader tries to search for signals from a target tag, which can burden the system. This searching can also excessively consume the batteries of readers and tags.

To address this, the suggested system searches blink signals from a tag and counts the number of signals. More than three signals means that the tag can be located without the help of an assistant tag and the suggested system determines the location of the target tag based on the three readers. On the other hand, when less than three signals are detected, the suggested system searches the nearest assistant tag to the target tag and estimates its location based on the two readers and the nearest assistant tag.

Second, although they suggested the assistant tag concept for converting NLOS to LOS environment, they did not how to select a proper assistant tag among various neighboring tags. Thus, an inappropriate assistant tag which can makes a large possible position area can be selected, which leads to decrease the accuracy of location-sensing.

In this dissertation, to select a specific assistant tag, geometric dilution of precision (GDOP) is used. GDOP is to evaluate the precision of a locating system in terms of the arrangement of satellites (readers). This method is firstly used to select proper satellites for accurate positioning. The arrangement of multiple readers according to the relative position of the readers can determine the level of precision in estimating a target location. Fig. 4-14 presents examples of good (low) and poor (high) GDOP. The measurement has error ranges, and the true location of the target will lie anywhere in the red area. In Fig. 4-14(b), the error range is the same with Fig. 4-14(a), but the possible position area (red area) has considerably grown due to the arrangement of two readers. These results mean that the close arrangement of readers can increase the uncertainty of location sensing.

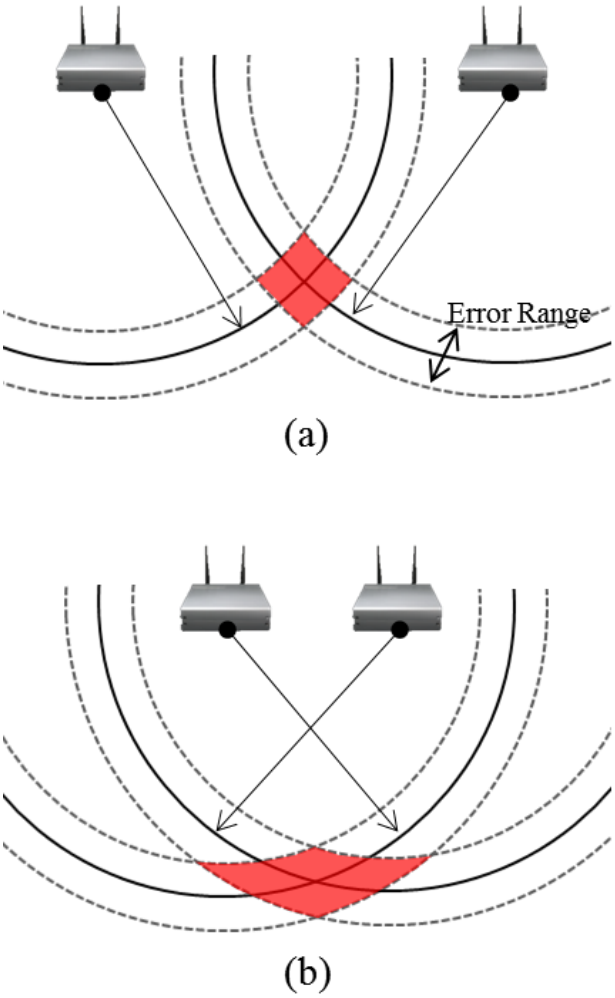


Fig. 4-14. Examples GDOP: (a) good (low) GDOP; (b) poor (high) GDOP

At first, the most recently updated tags among tags with connectivity are extracted. Each selected tag and readers receive a blink message from a target tag are used to calculate GDOP rate. Before calculating GDOP, dilution of precision (DOP) value should be obtained.

Langly (1999) introduced the calculation procedure of GDOP. As a first step in calculating DOP, consider the unit vectors from the target tag (receiver) to reader (satellite) i = $\left(\frac{(x_i - x)}{R_i}, \frac{(y_i - y)}{R_i}, \frac{(z_i - z)}{R_i}\right)$ where $R_i = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}$ and where x, y and z denote the position of reader i.

Formulate the matrix, A as

$$A = \begin{bmatrix} \frac{(x_1 - x)}{R_1} & \frac{(y_1 - y)}{R_1} & \frac{(z_1 - z)}{R_1} & -1 \\ \frac{(x_2 - x)}{R_2} & \frac{(y_2 - y)}{R_2} & \frac{(z_2 - z)}{R_2} & -1 \\ \frac{(x_3 - x)}{R_3} & \frac{(y_3 - y)}{R_3} & \frac{(z_3 - z)}{R_3} & -1 \\ \frac{(x_4 - x)}{R_4} & \frac{(y_4 - y)}{R_4} & \frac{(z_4 - z)}{R_4} & -1 \end{bmatrix} \quad \text{(Eq. 4-2)}$$

The first three elements of each row of A (Eq. 4-2) are the components of a unit vector from the target tag to the indicated reader.

Formulate the matrix, Q, as

$$Q = (A^T A)^{-1} \text{(Eq. 4-3)}$$

Then, the elements of Q are designated as

$$Q = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{xz} & \sigma_{xt} \\ \sigma_{xy} & \sigma_y^2 & \sigma_{yz} & \sigma_{yt} \\ \sigma_{xz} & \sigma_{yz} & \sigma_z^2 & \sigma_{zt} \\ \sigma_{xt} & \sigma_{yt} & \sigma_{xt} & \sigma_t^2 \end{bmatrix} \text{(Eq. 4-4)}$$

By using Eq. 4-4, GDOP is given by

$$GDOP = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2 + \sigma_t^2} \quad \text{(Eq. 4-5)}$$

After calculating GDOP rate of each connected assistant tag, an assistant tag with the best GDOP rate (lowest value) is selected and used to estimate a target location. By selecting an appropriate assistant tag, NLOS environments can be converted to LOS environments and the accuracy of location sensing can be also improved.

4.3.3 Location Tracking System Development

(1) Locating Devices

Tags and readers are the two main devices required to establish RFID-based RTLS. A tag attached to a target object (in this paper, a tag attached to the left side of a safety helmet) sends a regular interval of blink signal, which holds data on the number of times an identification (ID) is sent, and can be

considered a signal from a tag to search neighboring readers. A tag consists of a microcontroller, a wireless communication module, and a sensor module. In the paper, the MSP430F5xxx series of Texas Instruments is selected as a microcontroller, which gives satisfactory performance and low power consumption. A radio transceiver is based on IEEE 802.15.4a, which uses a chirp spread spectrum (CSS) on the 2.4 GHz industrial science medical (ISM) band. In addition, the tag uses acceleration sensors to calculate the velocity of a target object, so that a candidate location range can be determined.

A reader consists of a core process and wireless communication modules. It receives blinks transferred from the target tag and retransferred back to the target tag to estimate the distance between the target tag and the reader. There are two types of readers: stationary and portable. In general situations such as office buildings, the stationary type is more appropriate because there is usually a power supply. Otherwise, the portable type is used with a battery. The battery life of the reader selected in this paper is more than three years. This means that if the portable type is installed once on a site, it will endure until the project ends.

(2) Location Engine

In a multi-step locating system, each device executes a periodic search and measuring phase to collect distance for location estimation. The suggested protocol reduces the number of packets included in the distance information to a blink message. Fig. 4-15 illustrates the suggested protocol for location

estimation. The protocol consists of a search phase (LOS and NLOS environment) and a measuring phase.

In the search phase, tags send a blink message to search for adjacent readers. The blink message includes information about the device and a distance list with neighboring readers. The message also contains how many readers are connected to a tag. When the blink message is received by the readers, they send the distance list to the location engine and transmit acknowledge (ACK) messages to the tags or assistant tags. Transmitting ACK messages to assistant tags means that the location engine determines an environment as an NLOS condition.

In the measuring phase, tags proceed by ranging with neighboring readers sequentially. The Symmetric Double-Sided Two-Way Ranging (SDS-TWR) method is used to measure the distance. The information on the distances is then forwarded to the locating engine.

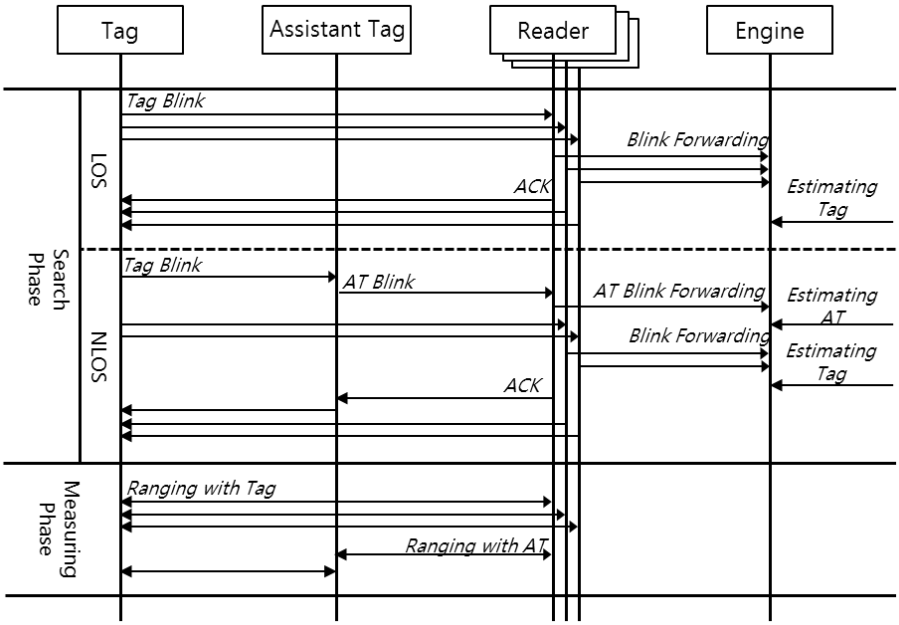


Fig.4-15. Protocol of the Suggested System

The location engine estimates the location of tags and assistant tags using the distances between the readers and tags (and assistant tags in an NLOS environment). In a construction site, the location engine accurately calculates a location using a pre-filter, estimator, and post-filter module to estimate the location of a target tag. Fig. 4-16 presents a block diagram of the location engine.

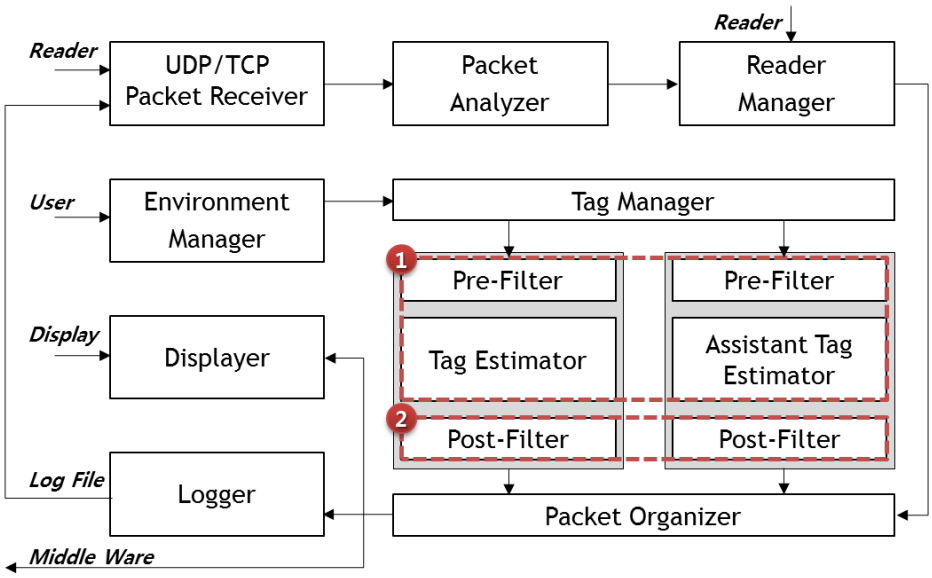


Fig. 4-16. Block Diagram of the Location Engine

The pre-filter (① in Fig. 4-16) discards failure values and mitigates errors caused by hardware. In addition, this module calculates the distance between each log and identifies an estimated log located a specific distance away from an immediately previous log, so that the estimator can receive candidates of invalid logs and remove other consecutive invalid logs.

The tag estimator and post-filter (② in Fig. 4-16) operate sequentially. The tag estimator calculates the distances neighboring logs located around invalid logs, and determines valid and invalid logs by using the suggested algorithm in section Error Mitigation." At the same time, the post-filter operates to persistently identify an initial log. Eventually, the tag estimator and post-filter is completed and the set of logs satisfying the suggested algorithm is selected as the final location.

4.4 Non-Activity Oriented Hazard Identification System Framework

As shown in Fig. 4-17, the framework suggested in this study consists of six databases and three modules. The role of each module is to have data input from related modules and databases, to process the data, and to send it to the next step of modules or databases.

The unidentified hazardous area identification module generates and stores an optimal route based on BIM objects extracted from a BIM drawing. Then, this module searches for workers' location logs that stray from this optimal route. A PHA is determined by a set of strayed logs and the workers' moving characteristics DB.

Searched PHAs can be classified into real hazardous areas, hazardous areas eliminated over time, work areas, and obstacles. Based on the workers' location log DB, activity area DB, object coordinates extracted from a BIM model and identified hazardous area DB, the filtering hazardous area module selects only the real hazardous area and eliminates the other areas. The result of this module presents hazardous areas that are on the construction site. The results will be represented in the monitoring and output generation module. All of these results are stored to reflect the generation and elimination processes of hazards over time.

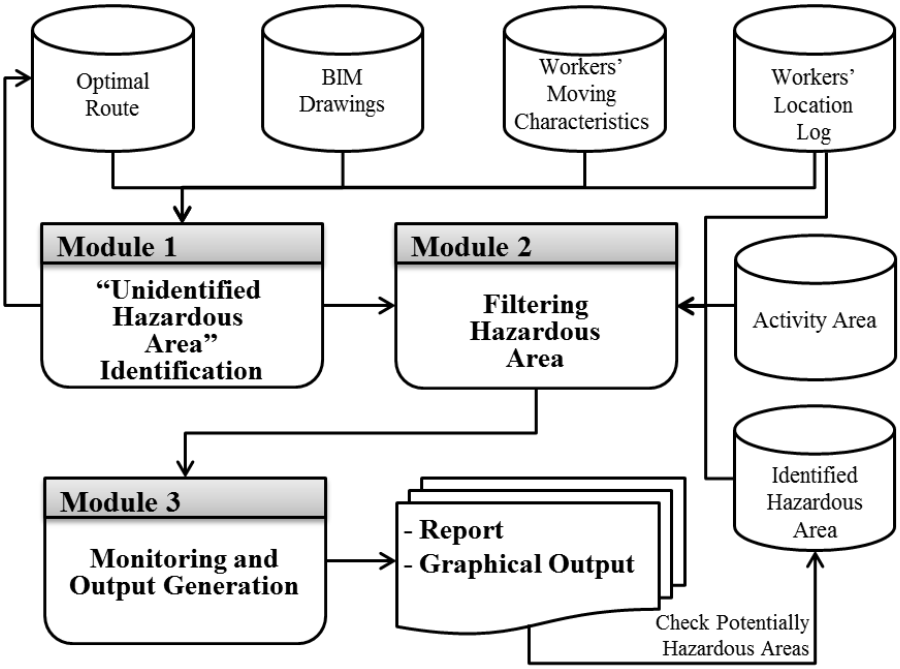


Fig. 4-17. Framework of Hazardous Area Identification

4.4.1 Module Ⅰ: Unidentified Hazardous Area Identification

The algorithm that identifies PHAs is shown in Fig. 4-18. This algorithm consists of three parts: extracting node elements and determining optimal routes, comparing workers' logs and optimal routes, and identifying potentially hazardous areas.

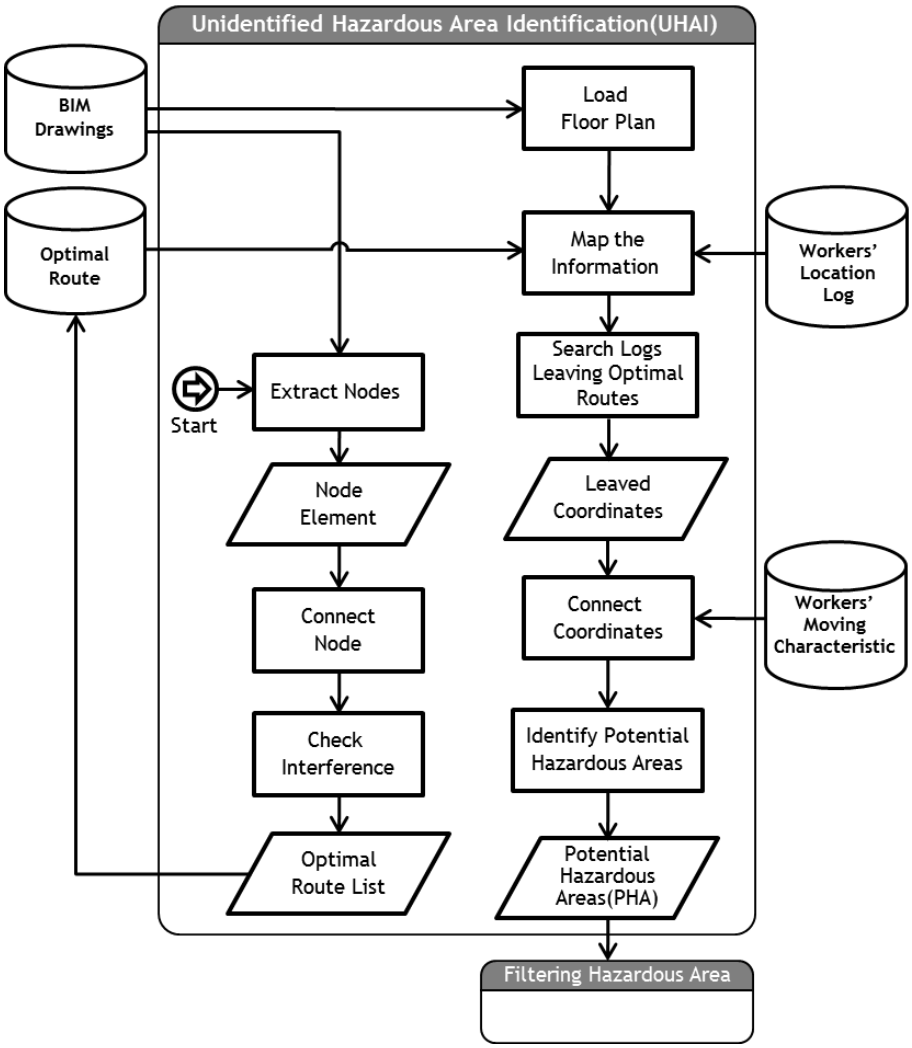


Fig 4-18. Unidentified Hazardous Area Identification Algorithm

The first part of the algorithm extracts node elements and determines optimal routes. An optimal route—defined as the shortest path when a worker moves from one point to another—is needed to search for PHAs by comparing workers' logs. The theoretically most efficient algorithm that we

know of for the shortest path problem is Dijkstra's algorithm (Ahuja et al. 1990). This is a graph search algorithm that solves the single-source shortest path problem for a graph with edge path costs, producing a shortest path tree (Dijkstra 1959). Despite the advantages of this algorithm, it does not fare well when directly applied to construction sites because selecting which object can be used as a node is problematic.

To solve this problem, this study applies the visibility graphs (for polygonal obstacles) introduced by Pocchiola and Vegter (1995). These researchers suggested visibility graphs for planning collision-free paths among polyhedral obstacles. Their algorithm calculated that the shortest Euclidean path between two points runs via the edges of the tangent visibility graph of the collection of obstacles.

Based on an algorithm from Pocchiola and Vegter (1995), the coordinates (the x-, y-, and z-axes, as well as the dimension) of BIM objects corresponding to the two points that can serve as a node (e.g. doors, stairs, hoists, elevators, the corners of walls, etc.) are extracted. Commercial BIM programs provide an extraction function that lists all of the objects with their coordinates, dimension, type, added properties, etc. This study uses this function to extract the type and coordinates of node elements and other objects. Each node is connected to all of the other nodes. After connection, coordinates composing each path are extracted. If there is no interference between the coordinates of the paths and all of the objects, these kinds of lines

and their coordinates remain. Workers can move on the set of remaining lines. Based on the set, the optimal route is identified by using Dijkstra's algorithm. The optimal routes are stored in an optimal route DB. Fig. 4-19 presents the modified Dijkstra's algorithm.

```
#include <iostream>
#include <set>
#include <vector>
using namespace std;

typedef vector<int> vi;
typedef pair<int,int> ii;
typedef vector<ii> vii;
typedef vector<vii> vvii;

const int MAX = 1001;
const int MAXINT = 1000000000;

int n;
vvii G(MAX);
vi D(MAX, MAXINT);

void Dijkstra(int s)
{
	set<ii> Q;
	D[s] = 0;
	Q.insert(ii(0,s));
```

```
    while(!Q.empty())
    {
        ii top = *Q.begin();
        Q.erase(Q.begin());
        int v = top.second;
        int d = top.first;

        for (vii::const_iterator it = G[v].begin(); it != G[v].end(); it++)
        {
            int v2 = it->first;
            int cost = it->second;
            if (D[v2] > D[v] + cost)
            {
                if (D[v2] != 1000000000)
                {
                    Q.erase(Q.find(ii(D[v2], v2)));
                }
                D[v2] = D[v] + cost;
                Q.insert(ii(D[v2], v2));
            }
        }
    }
}

int main()
{
    int m, s, t = 0;
    scanf("%d %d %d %d", &n, &m, &s, &t);

    for (int i = 0; i < m; i++)
```

```
    {
        int a, b, w = 0;
        scanf("%d %d %d", &a, &b, &w);
        G[a - 1].push_back(ii(b - 1, w));
        G[b - 1].push_back(ii(a - 1, w));
    }

    Dijkstra(s - 1);

    printf("%d\n", D[t - 1]);

    return 0;
}
```

Fig. 4-19. Dijkstra's Shortest Path Algorithm (modified from Dijkstra 1959)

The second part of the algorithm (Fig. 4-18) compares workers' logs and optimal routes. All of the optimal routes are mapped onto the floor plans and compared to the workers' logs. Workers' logs are tracked by RFID-based RTLS. This module searches logs that stray from an optimal route. The logs that are located farther from the optimal route than a certain distance (50cm) are stored. This distance is determined by experiments; details will be provided at the end of this section.

In the last part, the algorithm (Fig. 4-18) identifies potentially hazardous areas where the deviation between workers' logs and optimal routes occurs.

The deviation is larger than the distance from the optimal route when workers move along the route.

Fig. 4-20 represents how to determine hazardous areas. When workers' location logs are presented as n_0, n_1, n_2… n_{13} in the same floor, a worker's movement path can be obtained by connecting those points. Moreover, if a range of a worker's optimal route is defined as an area between an optimal route and the distance ($\boldsymbol{d_a}$) from it, those points of n_1—n_{12} can be considered as they stray from the optimal route. A potentially hazardous area is assessed based on several steps. First, the strayed logs are connected in chronological order. Second, n_1 (the first point straying from an optimal route) and n_{12} (the last point straying from an optimal route) are also connected in a straight line. Finally, considering $\boldsymbol{d_b}$—the distance between a worker and a hazard when the worker recognizes the hazard—a potentially hazardous area can be extracted. The method used to assess $\boldsymbol{d_a}$ and $\boldsymbol{d_b}$ is explained at the end of this section.

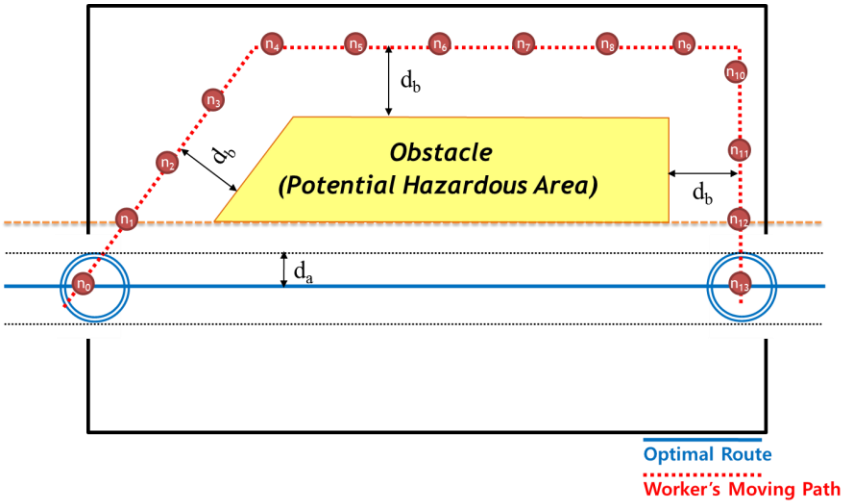


Fig. 4-20. Example of How to Determine a Hazardous Area

4.4.2 Module Ⅱ: Filtering Hazardous Area

The potentially hazardous areas obtained by module 1 can be considered as the following sets: real hazardous areas, areas where hazards are eliminated over time, work areas, and areas with piled materials. To identify hazardous areas, every area except the hazardous areas should be eliminated. The filtering hazardous area module performs this elimination process. This module includes two steps. The first step automatically filters hazards that are eliminated over time. The second step filters potentially hazardous areas.

First of all, the algorithm compares the coordination of identified hazardous areas and workers' logs. If a worker's location log is on the identified hazardous area on a path, the area can be considered managed or controlled. In this case, the algorithm eliminates the area from the identified

hazardous area database. PHAs are extracted in the unidentified hazardous area identification module. They can be defined as the union of unidentified and identified hazardous areas and work spaces.

To provide proper information to safety managers, identified hazardous areas and work spaces are filtered from the potentially hazardous areas. Work spaces imclude dynamic obstacles such as temporary equipment. If an activity using temporary equipment is also filtered as same process of identified hazardous areas by using workers' location logs. By doing this, the algorithm results in a set of twice-filtered potentially hazardous areas. Information about the potentially hazardous area is sent to the monitoring and output generation module. The process mentioned above is represented in Fig. 4-21.

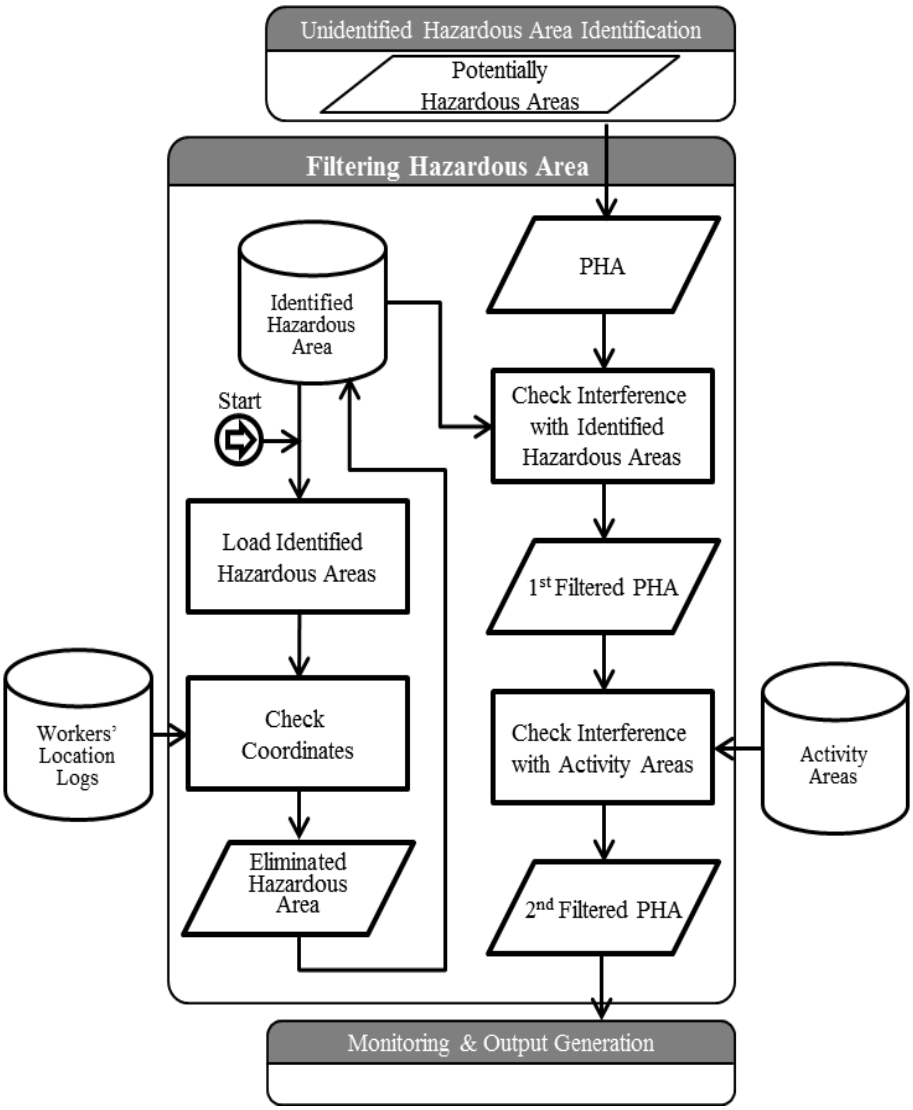


Fig. 4-21. Filtering Hazardous Area Algorithm

4.4.3 Module Ⅲ: Monitoring and Output Generation

The detailed algorithm of the monitoring and output generation module is shown in Fig. 4-22. This module starts by receiving the twice-filtered

potentially hazardous areas. These areas are mapped onto a BIM drawing and the coordinates of the areas are extracted. Based on the coordinates, the algorithm generates reports including graphical outputs, coordinates of hazardous areas, workers coming close to hazardous areas, and related activities.

From these results, safety managers can identify the location of potentially hazardous areas. They can then check the areas with a high potential of hazard risk, and judge whether or not the areas are risky. If some areas are identified as risky areas, they are stored in the identified hazardous area database. The database updates the generation and elimination process of hazardous areas over time. The filtering hazardous area module removes the eliminated hazardous areas, and the monitoring and output generation module produces the hazardous areas. This means that these algorithms reflect the construction characteristics of changing hazardous areas over time.

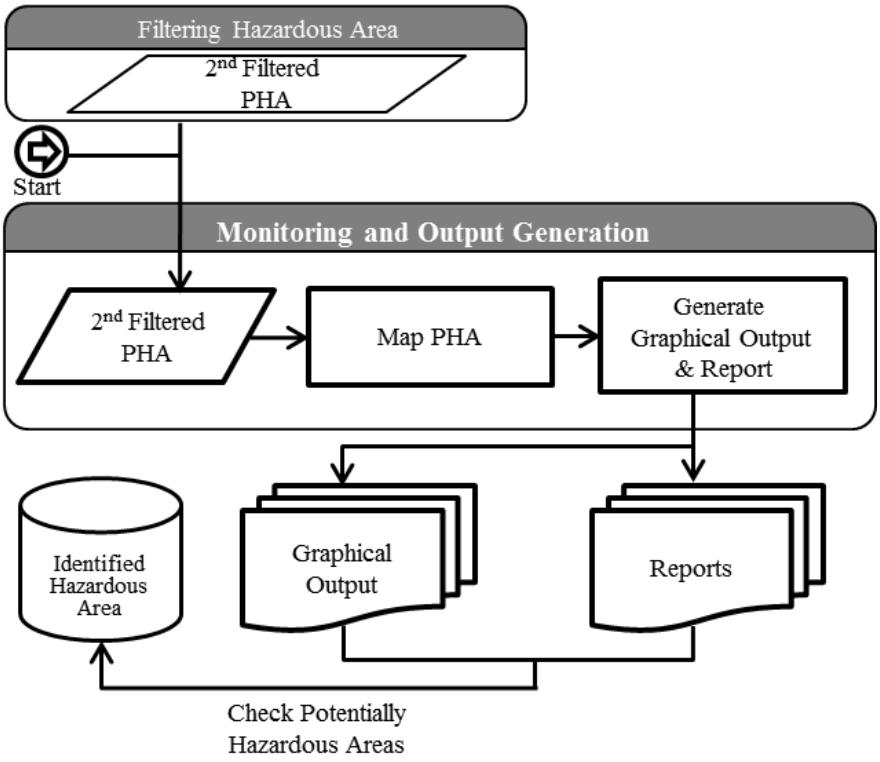


Fig. 4-22. Monitoring and Output Generation

4.4.4 Non-Activity Oriented Hazard Identification System Development

The purpose of this section is to develop a system by using a library developer kit, an API developer kit, and an RTLS device. Before system development, the protocol—a data form to transmit data between each RTLS device—should be defined. The detailed contents of these protocols and devices are shown in Tables 4-1 and 4-2.

Table 4-1. Definitions of System Protocol

Data	Project ID	Tag ID	Coordination		Floor	Time
			X	Y		
Byte	2	4	4	4	1	7

Table 4-2. Specification of Reader and Tag

	Reader	Tag
Function	• Determining distance to tags • Sending data to base-station	• Tracking location
Specifications	• Cortex-M3 of ARM corporation • ISA Supported: ARMv7-M • CPU Structure: RISC • Semiconductor Technology: CMOS • 72MHz • 128KB flash, 20KB RAM • Battery-based	• Processor : TI MSP430 • RF Transceiver : NA5TR1 • Transmission distance : 450m • 0.1 μA RAM retention • 0.4 μA Standby mode (VLO) • 0.7 μA real-time clock mode • 220 μA / MIPS active • Ultra-Fast Wake-Up From Standby Mode in <1 μs • Device parameters • Flash options: 0.5–16 KB • RAM options: 128–512 B • GPIO options: 10, 16, 24 pins • ADC options: Slope, 10-bit SAR • Other integrated peripherals: Capacitive Touch I/O, up to 2 16-bit timers, watchdog timer, brown-out reset, USI module (I^2C, SPI), USCI module, Comparator_A+, Temp sensor

A two-step approach is taken to develop the system. First, BIM property values are extracted using a developer's kit. In this study, ArchiCAD 13—an easy tool to use for developing applications—is used, and C++ is selected for programming. Secondly, the protocols of extracted BIM property values and workers' location logs are defined. Then, data is connected to the detailed algorithms. These algorithms are developed using Visual Studio 6.0 and C#.

The system user interface (UI) is presented in Fig. 4-23. The monitoring system part (1) collects workers' location logs and transmits this information to the hazard identification part (2). Based on transmitted logs, the hazard identification part finds hazardous areas using the suggested algorithms. This will result in a list of hazardous areas including ID, coordinates, and a safety manager's check. The reason why a BIM model is not fully used is a system burden. When the suggested system initially developed, the authors had tried to use 3D model of the project instead of the CAD-type floor plans.

Under this condition, the suggested system should be incorporated as a BIM-API. However, incorporating workers' location logs into a BIM model can burden a BIM system. Workers' location logs are also recognized as moving objects in a BIM model. A worker with an RFID tag transmits a large number of signals per second, thus the total number of all workers' location logs can decrease a system's data processing capacity. To overcome these difficulties, we select a method that extracts the coordinates and schedule

information of objects (numerical data form) and exterior wall boundary from a 3D model. Extracted information about objects is used to find the shortest path, and the exterior wall boundary is also used to represent PHAs. Using this approach, excessive data processing is not required and a large number of workers' location logs can be easily handled.

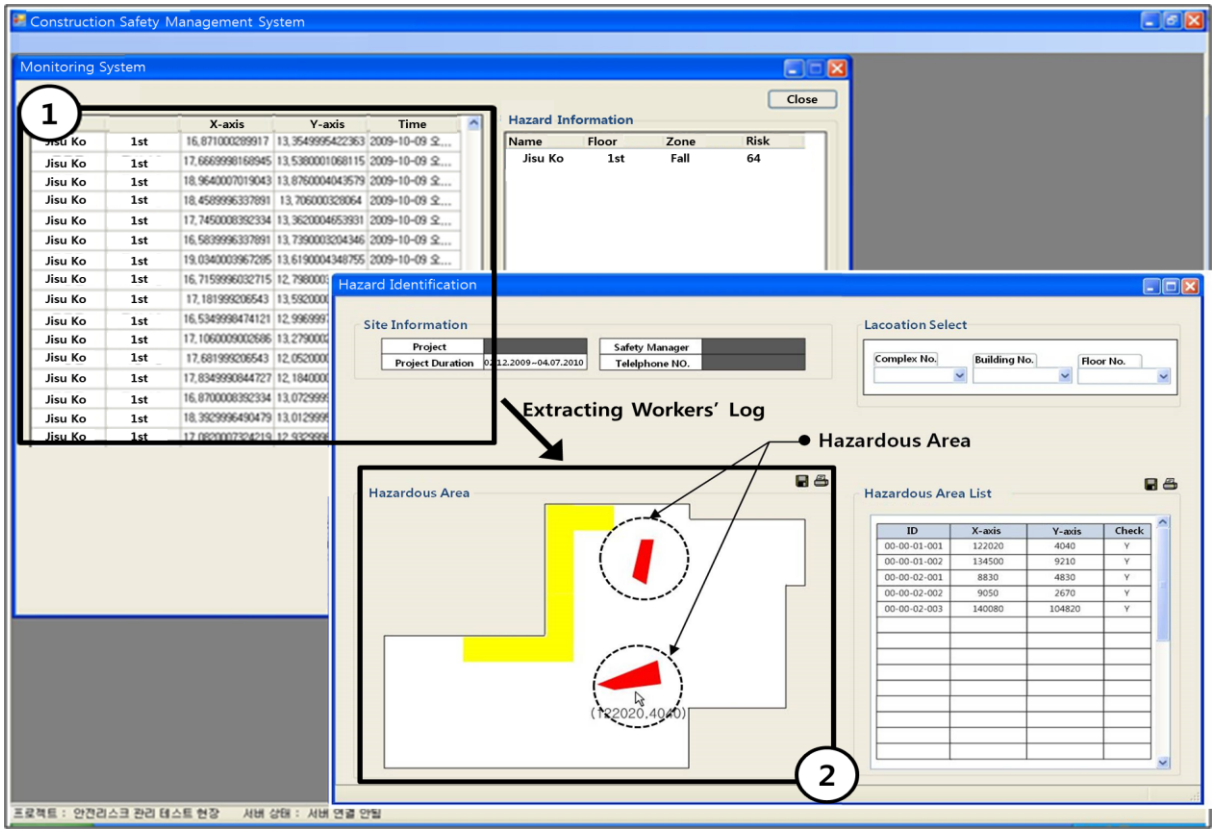


Fig. 4-23. System User Interface

4.5 Summary

Currently, hazard identification on movement paths is performed by human observers. This approach requires additional efforts and presents difficulties reflecting the dynamic changes of construction sites. Therefore, a systematic method is needed to improve the efficiency of non-activity oriented hazard identification.

This chapter suggests an automated hazard identification method using workers' location logs obtained from RFID-based RTLS. This method starts from a workers' movement characteristic, namely that workers in movement tend to want to follow the shortest possible path. By analyzing automatically tracked workers' location logs, hazards on movement paths can be identified. Six databases and three modules are developed and all components are integrated.

Through the three steps of hazard identification procedure, initial PHAs include not only hazardous areas but also other areas such as material piled areas, activity areas and previously identified PHAs. The second algorithm filters out the other areas from initial PHAs, and 2nd filtered PHAs are consequently remaindered.

This chapter also presents a location sensing accuracy improvement method. Because aforementioned three algorithms are operated based on

workers' location logs, the location sensing accuracy is one of the most important factors to the performance of the suggested system. Therefore, two methods are suggested to mitigate locating errors under the LOS and NLOS environments. Considering workers' movement characteristics and construction site environments, the error mitigation algorithm (for LOS environments) and the modified assistant tag method (for NLOS environments) are suggested.

Chapter 5. Hazard Identification System Validation using Case Study

This section will present validation process and its results of both activity-oriented hazards and non-activity oriented hazard identification methods. The validation strategy of each method is suggested. The validation procedure of each system includes case study and comparison between the suggested systems and current practices. Through this procedure, the accuracy and effectiveness of the suggested systems will be introduced.

5.1 Validation Strategy

It is difficult to validate a system. Especially, if a system is related to safety, the validation becomes more difficult work (Lee et al. 2012a). In this dissertation, two systems are suggested to identify both activity-oriented hazard and non-activity-oriented hazards. For this reason, analysis in this chapter mainly takes the form of case study.

Three case studies are performed in different construction sites. The reasons why different sites are selected are as follows: 1) the times when the suggested systems and the sub-module are fully developed are different; 2) because the characteristics of the suggested systems and the sub-module, they

are suitable for different sites (e.g. construction phase, progress of construction, work type, etc.); and 3) some sub modules needs various conditions for proving their effectiveness and accuracy.

Although the case studies present how they work and their effectiveness, the suggested systems should introduce their contributions to current practices. For doing this, several comparisons between the suggested systems and current practices are performed.

5.2 Activity Oriented Hazard Identification Validation

To validate the suggested activity oriented hazard identification system, a case study and comparison with a current accident case retrieval system are performed. The case study tries to present how the suggested system works. The sample case site is an apartment project and the suggested system had been applied for two month.

Moreover, the suggested system performance is validated in two ways. First, the retrieval performance of the system was evaluated through comparison with the established accident case retrieval system of KOSHA, because KOSHA's system is the most frequently used system for searching past accident cases in Korea. Second, usability was validated by performing a user assessment.

5.2.1 Case Study

The sample case is an apartment building construction project located in Seoul, Korea. This construction project duration is from July 2009 to April 2012. The total gross area is 25,660m^2 and the project consists of four buildings with 24 stories and 340 households. The suggested system had been used in this project for two months (February to March 2011).

Fig. 5-1 presents examples of retrieval results during each phase. Query was generated by the combination of influence factors. Based on the query set, the retrieval process was performed and the results presented were coordinates of the related BIM object, date of activity to be performed and relevant workers (only provided in daily plan phase) as well as similar past accident cases. Similarity score for each phase varies mainly because the number of indices is smaller than others. It is generally shown that preliminary phase has higher similarity score than those of other phases.

Based on this information, safety managers can easily define where (coordinates) and when (date) safety countermeasures are needed. Moreover, managers can provide past accident cases for the relevant workers with ease. The lower corner of Fig. 5-1 (number 2) presents retrieved similar accident cases. There are four examples such as ‘injury caused while loading rebar on slab form’, ‘death caused by a dropping rebar attached to gangform’, ‘death from a fall while assembling rebar’, and ‘injury during slab rebar

transportation'. For example, 'death from a fall while assembling rebar (A0172)' has 0.641 of SS and related worker's information is shown in the lower right corner of Fig. 5-1 (number 3). A0172 is a past accident case with circumstances that is similar to those of the current site. By using PMIS and BIM, the coordinates of related object (25934, 20583, 13200) and information of related worker (age, occupational type, workdays on current site, and safety training) are provided to a user. When a user clicks the title, detail information (including details of past accident and prevention methods) pops up on the system and is given to the related worker (RL113).

Query Set

Query_Set		Activity Time		Influence Factor									
Element ID	Unique ID	start time	end time	Work Process Rate	Cost of construction	Work type	Building type	Date (Month)	Occupation type	Age	workdays on current site	Safety training	Temperature
Structural	7)												
	EF4B9-F0D9-48AD-AE19-B81460B69A12	2011-03-01	2011-03-06	23	1195	Concrete	Resident	3	Common	45	15	3	12
									Common	35	14	2	
									Rebar	44	22	3	
	A2C88004-E939-494E-AC0A-5E89221D1B1E	2011-03-01	2011-03-06	23	1195	Concrete	Resident	3	Rebar	52	22	4	12
SW - 009													
	0470BA86-D070-4BE7-B173-7AC31499F553	2011-03-01	2011-03-01	23	1195	Concrete	Resident	3	Rebar	32	4	1	12
									Rebar	50	40	1	
	108AD6A7-0F6C-4C17-866A-6BFEE54552A4	2011-03-01	2011-03-01	23	1195	Concrete	Resident	3	Rebar	24	4	1	12
									Common	38	9	1	
	6CB38C44-BF6B-4A3F-AF88-A6BE6EF09B80	2011-03-03	2011-03-06	23	1195	Concrete	Resident	3	Form	50	38	6	12

Retrieval Result

Retrieval Result			Similar Accident Case						Related Labor ID			
Unique ID	Activity Time		No. Dis(1)	No. Dis(2)	No. Dis(3)	No. Dis(4)	No. Dis(5)	No. Dis(6)	RL ID(1)	RL ID(2)	RL ID(3)	RL ID(4)
	start time	end time	SS	SS	SS	SS	SS	SS	SS	SS	SS	SS
88CEF4B9-F0D9-48AD-AE19-B81460B69A12	2011-03-01	2011-03-06	A0037	A0217	A0501	A0052	A0321	A0039	RL022	RL134	RL019	
			0.690	0.606	0.584	0.547	0.521	0.486	0.690	0.628	0.582	
20EAE4EA-1B8A-427A-8E4B-AF6B7B021E17	2011-03-01	2011-03-06	A0583	A0163	A0224	A0486	A0015	A0551	RL089			
			0.883	0.803	0.739	0.680	0.677	0.642	0.883			
E3AFBEBE-B245-4601-AB9B-7BCA1FACD3A1	2011-03-01	2011-03-01	A0172	A0022	A0386	A0301	A0286	A0383	RL113	RL082		
			0.641	0.554	0.499	0.482	0.449	0.415	0.641	0.596		
108AD6A7-0F6C-4C17-866A-6BFEE54552A4	2011-03-01	2011-03-01	A0036	A0531	A0099	A0130	A0402	A0061	RL036	RL004		
			0.795	0.649	0.571	0.526	0.486	0.447	0.795	0.635		
B7317299-235D-4B79-8172-4BC541A35BA7	2011-03-03	2011-03-06	A0384	A0314	A0145	A0037	A0449	A0109	RL029	RL041	RL083	RL086
			0.597	0.527	0.484	0.447	0.405	0.376	0.597	0.557	0.512	0.485
AFE767AB-7845-43DB-87BC-9253310E1ABE	2011-03-03	2011-03-06	A0202	A0076	A0324	A348	A420	A0156	RL152	RL078	RL003	
			0.522	0.498	0.428	0.360	0.342	0.335	0.522	0.469	0.442	

Fig 5-1. Example of the Information Retrieval Results

5.2.2 Comparison with Accident Case Retrieval System

Due to the size and dynamic nature of document collections and users, evaluating the retrieval performance of search engines is difficult (Manning et al. 2008). Despite these difficulties, there are various methods to evaluate information retrieval systems. Many of the developed methods focus on how to measure the effectiveness of ad hoc information retrieval. Precision and recall are the two most frequent and basic measurement indices for information retrieval effectiveness (Manning et al. 2008). The definitions of precision and recall are as follows:

Precision (P) is the fraction of retrieved documents that are relevant (see Equation 5-1).

$$\text{Precision} = \frac{number\ of\ (relevant\ items\ retrieved)}{number\ of\ (retrieved\ items)} = P(relevant / retrieved) \quad \text{(Eq. 5-1)}$$

Recall (R) is the fraction of relevant documents that are retrieved (see Equation 5-2).

$$\text{Recall} = \frac{number\ (of\ relevant\ items)}{number\ (of\ relevant\ it)} = P(retrieved / relevant) \quad \text{(Eq. 5-2)}$$

Before calculating precision and recall, experiment collections should be prepared. Generally, text retrieval conference (TREC) collections have been used to evaluate the performance of an information retrieval system (Zobel et al. 1996; Jansen and Pooch 2001). However, it is difficult for the suggested system to directly apply the text collections of TREC because the database is structured based on past accident cases. Therefore, the precision and recall of this study were evaluated through the comparison of KOSHA's past fatal accident case retrieval system, which is the most commonly used accident retrieval system in Korea.

As shown in Fig. 5-2, KOSHA builds accident case database which embraces the entire field of the industry and provides a system that can search for cases. Especially, the construction accident case category includes 3,785 fatal accident cases and each case consists of a title and contents (including figures). The search algorithm of KOSHA's accident case retrieval system is very similar to that of OSHA and HSE. When a user enters a query (typically by using keywords related to accidents) into a search box (the red line box in Fig. 5-2), the KOSHA system examines its database (fatal accident cases) and provides listing of best-matching cases. The search engine of the KOSHA system supports the use of the Boolean operator AND, OR, NOT to further specify the search query. The engine looks for words or phrases exactly as entered. It does not provide an advanced feature called proximity search, which allows users to define the distance (or relation) between keywords. It means that the KOSHA system does not provide partially matched results but

exactly matched results. The effectiveness of information retrieval system depends on both the relevance of the result set and the amount of data stored. The KOSHA system has sufficient cases but do not provide an effective method for searching and ranking relevant results. The process of the KOSHA system is familiar to the general and untrained users.

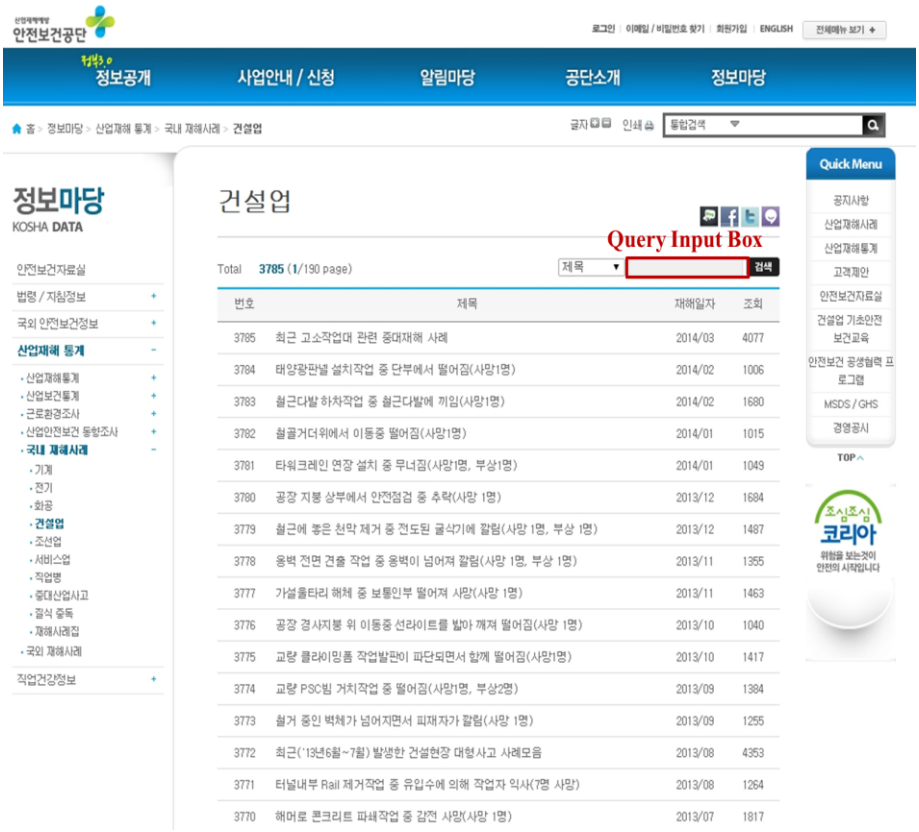


Fig. 5-2. Example of KOSHA's Accident Case Retrieval System (http://www.kosha.or.kr/board.do?menuId=544)

However, this process requires repetitive tasks when a user wants to

search vast quantities of information. KOSHA system requires repetitive retrieval work and does not provide the weight of each index. Hence, it retrieves results that do not necessarily reflect the characteristics of each safety planning phase. For example, on a construction site where there are approximately three hundred workers, three hundred or more searches should be performed to find similar past accident cases with consideration of trades, work conditions and characteristics of workers.

Although these searches are performed, repetitive search is needed when work, work condition, and workers are altered. The suggested system extracts information from PMIS and BIM and automatically generates a query. Moreover, the system can update a query and search past accident cases whenever the conditions of a site are changed. This automated procedure can reduce the burden of safety managers' work and time due to repetitive searches. One hundred past accident cases were extracted from KOSHA's fatal construction accident database by using the random extraction function. The calculations of precision and recall were performed by reflecting safety steps: preliminary plan, monthly plan, and daily plan. The number of indices for each safety plan is different. Precision and recall were estimated by using Formulas 4 and 5, and the results are presented as Table 5-1.

Table 5-1. Retrieval Effectiveness Comparison Results

Safety planning phase	Preliminary plan		Monthly plan		Daily plan	
Number of indices	4		6		10	
Search system	P	R	P	R	P	R
KOSHA system	0.72	0.61	0.63	0.41	0.84	0.34
Suggested system	0.68	0.82	0.79	0.75	0.92	0.61

5.2.3 Usability Validation

Although the retrieval performance meets the demand of users, without usability, the retrieval system cannot be a fine system (O'Keefe and O'Laery 1993). For this reason, user assessment was performed in order to validate the usability of this study. Total 50 surveys were distributed and 34 workers had participated in the user assessment process and were provided with retrieved past accident cases for three weeks.

Questionnaires were administrated to the workers in order to collect data to support the validation. The survey assessed both the suggested system and the KOSHA system in five aspects. Table 5-2 presents the main concerns addressed by the questionnaire designed for the validation of the suggested system.

Table 5-2. Sample of Questionnaire for Usability Validation

Aspect	1	2	3	4	5
(a) the reflection of construction works and site conditions					
(b) the suitability of the selected indices composing queries					
(c) the ease of use of the suggested system					
(d) the promptness of the system					
(e) the usefulness of the retrieved cases					

Note: Please evaluate the suggested system on the following six aspects (Check the appropriate response category using the five-point scale, with 1 representing poor and 5 representing excellent.)

To confirm the two systems have different user assessments, pair Student's t-test was performed in each individual question, and the result is shown in Table 5-3. For all the questions, null hypothesis ($H_0 : \mu_1 - \mu_2 = 0$) is rejected. It means that the users do not consider that the two systems are equivalent. Considering the average values of the questions for each system, the suggested system presents higher average than that of the KOSHA's.

Table 5-3. Analysis of Survey Results

Question	(a)		(b)		(c)		(d)		(e)	
	The suggested system	KOSHA system	The suggested system	KOSHA system	The suggested system	KOSHA system	The suggested system	KOSHA system	The suggested system	KOSHA system
Average	4.353	3.206	4.029	2.706	4.765	3.412	4.618	3.118	4.500	2.882
Variance	0.417	0.471	0.575	0.941	0.185	0.856	0.243	0.410	0.621	0.713
Degree of freedom (*df*)	66		66		66		66		66	
t statistic	7.095		6.268		7.732		10.821		8.166	
Critical value of t	1.997		1.997		1.997		1.997		1.997	
P-value	1.10E-09		3.17E-08		3.99E-11		1.44E-16		6.68E-12	
H0: μ_1-μ_2=0	rejected		rejected		rejected		rejected		Rejected	

Except two participants who rated the KOSHA higher than the suggested system, 32 of the workers participated in the assessment reported that the suggested system has the potential to be an effective accident prevention tool due to following aspects: (a) the reflection of construction works and site conditions; (b) the suitability of the selected indices composing queries; (c) the ease of use; (d) the promptness of the system; and (e) the usefulness of the retrieved cases.

The suggested method has a strength compared to current safety training methods. Safety managers provide guide lines to workers for preventing an accident. This act usually takes place in a tool-box meeting. Although the provided information is useful, information is general and rarely considers workers' characteristics or detail acitivities.

On the other hand, the suggested system retrieves similar past accident cases based on workers' characteristics, work conditions, and works. The end users (workers) can receive the most related similar past accident case to them. Based on this 32 of workers also expressed that similar accident cases related to their activities are useful information. In detail, safety managers and workers obtain similar past accident cases related to their works automatically. They satisfied that the automated retrieval process decreases the effort for searching similar accident cases and reflecting the construction progress.

5.3 Non-Activity Oriented Hazard Identification Validation

Accuracy and promptness can be considered two of the most important criteria of a safety management system. Identifying hazardous areas on worker's paths encourages safety managers to respond promptly to these areas. It can also reduce the time in which a hazardous area is left without any countermeasure, so that the degree of workers' exposure to a hazard decreases.

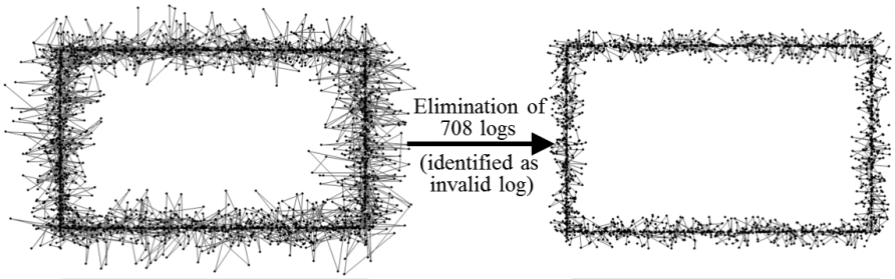
The high accuracy of hazard identification provides the precise coordinates of a hazard. Information about a hazardous area provided to a safety manager can lighten the burden of identifying hazards, and it also enables a proactive response to a hazard. Therefore, a case study was executed to test the system promptness and accuracy of hazard identification. The test site was an academic building with 5 stories and a 5000m^2 gross floor area. The test was performed from August to September 2011.

Before executing the case study, base-stations, readers, and tags were installed at their own proper location. Base-stations have their z-axis coordinates. Because the RTLS readers and tags used in this study do not have an identifier to represent z-axis coordinates, base-stations should have z-axis coordinates. This means that at least one base-station should be placed on every floor. RTLS readers are also set in the corners of each floor to increase the effective propagation area. After installing the base-stations and RTLS

readers, their coordinates are mapped in the engine. Taking the height of readers into account, RTLS tags are mounted on a helmet.

5.3.1 Accuracy of Location Sensing

Tests for the accuracy of the suggested system were conducted in two different tracking environments. The first test was performed in a basement parking lot of an ongoing apartment project. The test area had no obstacles except some structural columns. The test analyzed tracking logs of a worker moving around a 7.5 m × 5.5m rectangle. The moving velocity of the worker was set in the range of 0cm/s (stop state) to 70cm/s. Fig. 5-3(a) shows the tracking logs when the location tracking error mitigation algorithms suggested in this paper were not applied. In this case, 2,648 logs were tracked and the average error was around 38 cm. On the other hand, Fig. 5-3(b) shows the tracking logs after eliminating the tracking errors identified by the location tracking error mitigation algorithms suggested in this paper. In comparison with the results shown in Fig. 5-3(a), the average tracking error was reduced to around 21 cm: a 44% improvement in location tracking accuracy. This improvement was obtained by eliminating the logs that exceeded the maximum distance that a target object can move in a specific time interval. In this test, 708 out of 2,648 logs were filtered out as tracking errors by the suggested algorithms. These results demonstrate the strong potential of the suggested location tracking error mitigation algorithms to improve the accuracy of location-tracking in construction projects.



Total No. of Logs	Avg. Error (cm)	Standard Deviation
2,648	38	31.77

(a)

Total No. of Logs	Avg. Error (cm)	Standard Deviation
1,940	21	14.78

(b)

Fig. 5-3. Improvement of Tracking Accuracy with Error Mitigation Algorithms

The suggested system was further applied in eight different construction environments in order to test its applicability and effectiveness under various situations, as shown in Fig. 5-4.

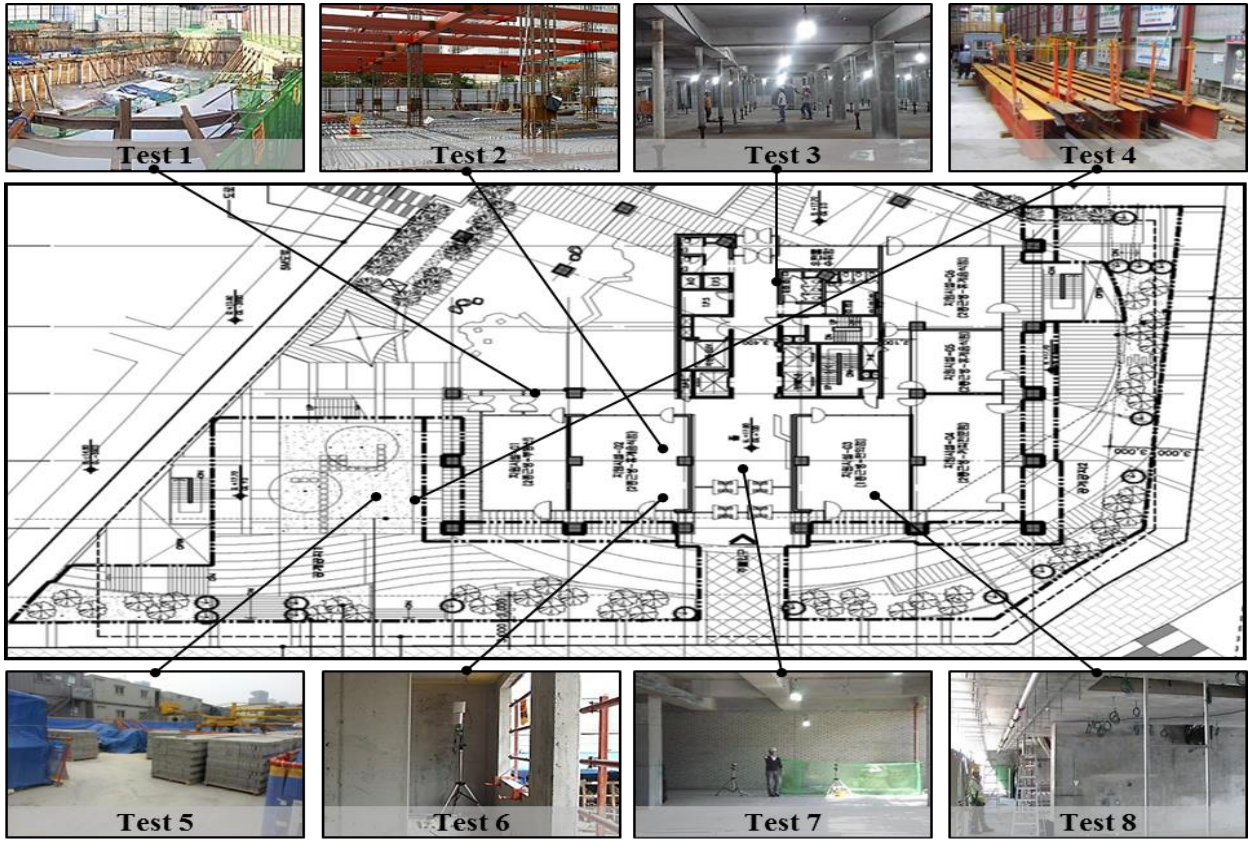


Fig. 5-4. Location Tracking Accuracy Tests

In each testing environment, four different experiments (A to D) were conducted in order to measure the effect of the location tracking error mitigation algorithms and the application of the assistant tag, as shown in Table 5-4. The four different experiments are as follows.

- Accuracy experiment A: location tracking applying neither the error mitigation algorithms nor the assistant tags;
- Accuracy experiment B: location tracking applying the error mitigation algorithms only
- Accuracy experiment C: location tracking applying the assistant tags only
- Accuracy experiment D: location tracking applying both the error mitigation algorithms and the assistant tags.

Table 5-4 shows the tracking results of the four different experiments under eight different construction environments. As expected, the average tracking error (256 cm) was greatest in experiment A. In contrast, experiment D showed the lowest tracking errors (58 cm), resulting in a 77.3% improvement in location accuracy in comparison with experiment A.

Particularly, experiment A showed a significant tracking error (791 cm) in stock yard environments where H-beams were piled (i.e., experiment A under test 4). This is because the steel of the H-beams easily causes heavy multi-path and signal attenuation. In this case, applying both the algorithms and the assistant tags (experiment D) can reduce the location tracking error to 96.5 cm, 87.9% of the improvement in tracking accuracy from experiment A. The test results showed that the 65% average improvement in experiment C was slightly higher than that in experiment B (50.3%), which suggests that assistant tags are more effective than the algorithms in tracking environments where there are many obstacles between readers and tags.

Table 5-4. Location Tracking Accuracy Test Results

Tests	Tracking Environments		Location Tracking Accuracy			
			Accuracy Experiment A	Accuracy Experiment B	Accuracy Experiment C	Accuracy Experiment D
Test 1	Earth Work Area (Outdoor)	Tracking Error	138 cm	61 cm	47 cm	38 cm
		Improvement	-	55.8%	65.9%	72.5%
Test 2	Steel Work Area (Outdoor)	Tracking Error	158 cm	120 cm	96 cm	70 cm
		Improvement	-	24.1%	39.2%	55.7%
Test 3	Underground Parking Lot	Tracking Error	116 cm	73 cm	76 cm	51 cm
		Improvement	-	37.1%	34.5%	56.0%
Test 4	Stock Yard (H-Beam piled)	Tracking Error	791 cm	321 cm	148 cm	96 cm
		Improvement	-	59.4%	81.3%	87.9%
Test 5	Stock Yard (Brick piled)	Tracking Error	549 cm	243 cm	152 cm	88 cm
		Improvement	-	55.7%	72.3%	84.0%
Test 6	Frame Structure (Indoor)	Tracking Error	127 cm	80 cm	78 cm	53 cm
		Improvement	-	37.0%	38.6%	58.3%
Test 7	Wall Structure (Indoor)	Tracking Error	78 cm	71 cm	57 cm	35 cm
		Improvement	-	9.0%	26.9%	55.1%

Tests	Tracking Environments		Location Tracking Accuracy			
			Accuracy Experiment A	Accuracy Experiment B	Accuracy Experiment C	Accuracy Experiment D
Test 8	Finishing Work Area	Tracking Error	86 cm	46 cm	61 cm	33 cm
		Improvement	-	46.5%	29.1%	61.6%
Average		Tracking Error	255 cm	127 cm	89 cm	58 cm
		Improvement	-	50.3%	65.0%	77.3%

Location tracking estimation time is an important factor in evaluating performance of an RTLS. Even though RTLS is very accurate, if locating a moving object is too slow, the RTLS is meaningless. The two methods suggested herein to improve the accuracy of RTLS (i.e., location tracking error mitigation algorithms and assistant tags) can significantly increase the location tracking estimation time. To address this issue, this paper suggests a floor classification method that does not measure the z-coordinate but uses the information of floor level in order to reduce the location tracking estimation time while maintaining the accuracy of location tracking. That is, two-dimensional coordinates (x- and y-), rather than three-dimensional coordinates (x-, y- and z-), are used for tracking the location of the target object by arbitrarily assigning the information of the floor number to the z-coordinate.

In order to measure the impact of the floor classification method in reducing location tracking estimation time, the test was conducted under the following three different settings.

- Time experiment A: measuring the estimation time when applying neither location tracking error mitigation algorithms nor assistant tags (i.e., the same to the accuracy experiment A in Table 1).
- Time experiment B: measuring the estimation time when applying both location tracking error mitigation algorithms and assistant tags (i.e., the same to the accuracy experiment D in Table 1).

- Time experiment C: measuring the estimation time when applying the floor classification method in addition to the error mitigation algorithms and assistant.

These three experiments were conducted in three different settings: one floor with four readers, three floors with twelve readers (four readers on each floor), and five floors with twenty readers (four readers on each floor). Furthermore, each experiment was conducted to measure the estimation time in tracking a different number of target tags (i.e., 10 to 50 tags), as shown in Table 5-5.

Table 5-5. Location Tracking Estimation Time Test Results

Experimental Settings			Location Tracking Estimation Time (ms)			Time Reduction ((B–C)/B)
Number of Floors	Number of Readers	Number of Tags	Time Experiment A	Time Experiment B	Time Experiment C	
1	4	10	157	227	173	23.8%
		20	282	348	309	11.2%
		30	418	525	445	15.2%
3	12	10	284	377	307	18.6%
		30	376	497	424	14.7%
		50	483	687	519	24.5%
5	20	10	254	409	298	27.1%
		30	448	564	420	25.5%
		50	756	863	628	27.2%
Average			384.2	499.7	391.4	21.7%

As expected, the time experiment A required the shortest time for location estimation and the time experiment B required the longest time. This was mainly attributed to the increased level of computational time and effort to apply the location tracking error mitigation algorithms and assistant tags. This result reveals a trade-off between location tracking accuracy and estimation time. That is, application of the error mitigation algorithms and assistant tags can increase the tracking accuracy by 77.3% (see Table 5-5) at the cost of 115.5 ms extra computational time. Although seemingly negligible, 115.5 ms may be significant, particularly in RTLS.

In comparison with the time experiment B, the time experiment C required a similar time (391.4 ms) to that of the time experiment A while maintaining the same level of accuracy of the experiment B. This result confirmed the effectiveness of the suggested floor classification algorithms in reducing location tracking estimation time, thus expanding the applicability of the suggested system for real-time location tracking purposes.

5.3.2 Comparison with Inspection of Safety Manager: Promptness

Hazardous area identification of four floors in a construction site is performed separately by a human-based approach and the suggested system. ‘Human-based approach’ means a traditional method in which human observers identify hazardous areas by checking each area. Through the test, three criteria—such as time, cost, and accuracy—are to be checked.

The time of a human-based approach, which is the duration of identifying hazardous areas, is longer than that of the suggested method. The suggested system analyzes workers’ location logs and identifies hazardous areas immediately. Although the suggested method needs equipment installation time, in terms of time, it can be considered that the suggested method needs less time for identifying hazardous areas than the current method. In this case study, the time difference between two methods may seem small because the floor plan of the case study building is simple. However, if a floor is complex, the time for identifying hazardous areas can be much longer because a human observer needs much more time and because of the moving distance for identification on a complex floor.

Another advantage of the suggested system that can be identified by this case study is that the duration of identifying a hazardous area on a workers’ path without a proper countermeasure decreases. Generally, a safety manager strays from identifying hazards. Therefore, there is a time gap between when a

specific place is checked once and when the place is checked again. Even if a hazard is generated on a once-checked place before the next check schedule, the hazard does not get registered in the management process. For example, when a safety manager checks the n_{th} floor and then he/she goes to the n_{th+1} floor to identify other hazards, it is difficult for that safety manager to identify a hazard that is generated on the n_{th} floor while checking the n_{th+1} floor. However, because the suggested system identifies coordinates of a hazardous area in real-time through the algorithms whenever a worker approaches a hazardous area after it is generated, the duration while a hazard leaves without any countermeasure is shorter than the current method.

5.3.3 Validation of Accuracy of Non-Activity Oriented Hazard Identification

To validate the accuracy of the suggested system, the ratio of the hazard identification of the suggested methods is analyzed. The results of the test are presented in Table 5-6. The system found 35 hazardous areas across the 4 floors (basement, ground, first, and second floor). Among potentially hazardous areas found by the suggested system, 80% are real hazards. The causes of hazards include: piling hazard materials, deficiency of electric wire protection, and deficiency of fall protection, among others. Piling hazard materials is the most frequent cause. Piled materials are understood to more easily attract workers' attention than other causes. On the other hand, hazards such as unprotected electric wires are difficult to see. Though hazards are on

the optimal routes, their visibility (or lack thereof) affects identification.

However, some of the results are defined as system errors. Those errors occur around a particular area. This area is near the piled rebar and forms. Iron materials are widely known to decrease the accuracy of wireless location systems. This decrease in accuracy may be caused by wave propagation attenuation. Without wave propagation attenuation, potentially hazardous areas on the optimal route are accurately obtained. Other errors are caused by radio shadow areas. As construction progresses, more readers are needed to acquire accurate location sensing results. If there are objects that form an obstacle, the reader's installation location should be modified to improve the accuracy of the suggested system. One error (among seven errors) is caused by a worker's abnormal behavior—that is, the worker not following the optimal route. Occasionally, workers stray and their location logs can make an error.

Table 5-6. Site Test Results of Non-activity Oriented Hazard Identification Method

Cause of a hazard	Number of cases
Piling hazard materials	9(26%)
Deficiency of fall protection (opening)	7(20%)
Deficiency of electric wire protection	4(11%)
Deficiency of falling and flying object protection	2(6%)
Misc	6(17%)
System Error	7(20%)

Example of Hazard Identification Result (Date 23-25 Oct 2010)

Hazardous area ID	X-axis (mm)	Y-axis (mm)	Hazard	Description
00-00-01-001	122020	4040	○	Piling hazard materials
00-00-01-002	134500	9210	○	Deficiency of electric wire protection
00-00-02-001	8830	4830	○	Piling hazard materials
00-00-02-002	9050	2670	×	-
00-00-02-003	140080	104820	○	Deficiency of fall protection (opening)

Although there are some errors in the results, the suggested system can contribute to safety management. In the hazardous areas identified by the suggested system, safety managers choose proper countermeasures to prevent accidents. Information regarding hazardous areas is also provided to workers such that they can improve their cognitive level and so be fully aware of any

hazards. The cost benefit (economic value) of the suggested system is difficult to measure.

The devices (tags and readers) used in this study are developed as a prototype. The production cost of a tag is approximately $6. A reader costs approximately $20. When the suggested system is commercialized, the production cost of a tag is likely to come down because of the great number of tags that will be produced if the suggested system is applied to various construction sites.

The cost benefit can be considered as another aspect such as the existence time of a hazard. The suggested system can minimize the lifespan of a potential hazardous area without any countermeasures. It can also decrease the time that workers are exposed to hazards and may help prevent accidents. Regulations on construction accident prevention are being strengthened. Consequently, an accident may cause the temporary interruption of a project or may require huge compensation. Considering these situations, the cost of the suggested system would not put a big financial burden on a construction project.

One remarkable thing drawn from the site test is the timing of the process cycles. The suggested method can operate in real time. It identifies PHAs by analyzing schedule information and workers' location logs. Identified PHAs are updated in real time, and safety managers can recognize updated PHAs.

The update takes place in two main situations. Even if there is no variation in the activity process, workers are constantly moving. The suggested system analyzes this movement and identifies PHAs. In addition, every time an object is installed, the shortest path changes. Based on changes, the new shortest path is determined and PHAs can be identified.

How often safety managers check PHAs depends on the safety management policy of each construction project. When a floor plan is complex and various activities are performed, PHAs may be more likely to exist than in the case of a simple floor plan. Under this circumstance, safety managers should inspect the updated PHAs. Although the suggested system identifies PHAs automatically, changed information about the schedule and activities should be entered into a BIM model to find the shortest path. If the update of changed information is not conducted, the system identifies PHAs based on the wrong shortest path and thus it can cause an error.

There are also other factors affecting hazard identification. Workers' location logs—which identify PHAs—are affected by workers' psychological states and training levels. Therefore, the suggested system can apply these factors. A worker with low safety perception or consciousness (caused by his/her psychological state and training level) has a tendency to not avoid hazards (O'Toole 2002). In this case, managers who use this system can modify the values of workers' moving characteristics to identify hazardous areas. For example, when the tendency of hazard avoidance is low, safety

managers decrease the value of da and db. This can lead the suggested system to identify more PHAs. Furthermore, this increases applicability of the suggested system to various construction sites.

5.4 Summary

This chapter deals with the validation of hazard identification system. This validation consists of case studies and comparison between the suggested systems and current practice.

By the case study of the activity-oriented hazard identification system, it is shown that the system can automatically generate queries which reflect construction site conditions. The retrieval results are also provided to corresponding workers. Automated retrieval and provision of similar accident cases can reduce the burden of safety managers' work. It can also help workers recognize expected hazards, which can in turn prevent an accident.

The retrieval results' accuracy is one of the most important concerns. To validate the accuracy of the retrieval results, a comparison between the suggested activity-oriented hazard identification system and KOSHA's accident case retrieval system is performed. Precision and recall are estimated for a quantitative comparison. The comparison results show the efficacy of the suggested system. With the exception of the precision value in the preliminary

planning phase, the suggested system obtains higher precision and recall values than KOSHA's system.

The validation procedure of the non-activity-oriented hazard identification system is more complicated than that of the activity-oriented hazard identification system since the former involves the location-sensing module (which requires proof of accuracy). The suggested algorithms for improving workers' location sensing accuracy show more accurate than a normal RTLS without error mitigation algorithms. After verifying the accuracy improvement of location sensing, the improved location sensing method is applied to the non-activity-oriented hazard identification system. The suggested system can automatically identify hazards with high accuracy. Approximately 80% accuracy is shown in the experiments. Given these results, it can be concluded that the suggested system is capable of identifying hazards on movement paths and that the results are reliable in terms of accuracy.

Chapter 6. Conclusions

There are two types of hazards classified by the causes. This study develops a hazard identification system that consists of two parts. Each part of the suggested system is developed based on consideration of each hazard's characteristics. This final chapter summarizes the findings and results of this dissertation. It discusses the contribution of this study to the current hazard identification and safety management.

6.1 Results and Contributions

Accurate and effective hazard identification is the prerequisite of successful safety management. Pre-existing hazard identification systems require iterative process to reflect the dynamic changes of a construction project and additional safety managers and effort. Moreover, current methods for identifying hazards focus on a specific hazard type (e.g., activity oriented or non-activity oriented hazard).

To address these problems, this study develops a hazard identification system using both information retrieval technique and RTLS-based data acquisition and analysis. By using the suggested system, safety managers may identify hazards much easier than current approaches. Further, the suggested system is computerized in order to facilitate it in the real construction sites. The results and contributions of this study can be summarized as follows.

(1) Current systems focus on improving the practical use of accumulated past accident cases. They provide information on related past accident cases when a user inputs queries. Generating a query every time a user retrieves information makes the current systems inconvenient. This process is not effective when a user wants to search for vast quantities of information. A user needs to perform repetitive retrievals (e.g., generating queries, inputting queries, selecting a result, providing a result to a relevant worker, etc.) in order to search for various past accident cases that can be used to prevent accidents in the current project. Therefore, this study proposes an accident case retrieval system that can automatically generate queries based on construction site conditions. The suggested activity-oriented hazard identification introduced an information retrieval system that automatically searches and pushes similar past accident cases to the related workers. By doing this, iterative process for information retrieval is eliminated and usability of the system is improved.

(2) In the retrieving process of activity-oriented hazard identification system, indices are classified by each construction safety planning phase. Each phase has different amount of information. Therefore, although the difference should be considered, current practices do not reflect it. In this study, obtainable information is analyzed and it

is classified by each phase. Based on the classified result, different number of indices is selected by safety planning phase. Considering each phase's characteristics, queries are more suitable to each phase. Consequently, the retrieved results can be more accurate.

(3) The retrieved results–the most similar past accident case–include information about time, coordinates and the most related worker. Through the results, safety managers can perceive related workers, coordinates and time of expected hazards. They can establish countermeasures by using provided the most similar past accident cases. In addition, workers perceive the most similar past accident cases to themselves. By this perception, workers can be careful to the expected hazards to themselves.

(4) To identify hazards on the movement path, this study uses workers' location logs based on an RFID-RTLS. Before developing the non-activity-oriented hazard identification system, workers' movement characteristics are analyzed. The main results reveal the following: 1) if there is not any obstacle on the shortest path, construction workers usually move along the path; and 2) if construction workers recognize an obstacle, they move around the obstacle. The workers are usually in the range of 50cm from the paths. If they recognize a hazard, they would be out of that range. The non-activity-oriented hazard system identifies potentially hazardous areas (PHA) on

workers' path by using the RTLS of workers and a building information model (BIM). The suggested system identifies PHA by using the deviation between the optimal route (the shortest path), which is determined by extracting nodes from BIM objects, and the real movement path of a worker, which is tracked by radio frequency identification. After this process, PHA can be divided into uncontrolled PHA and already controlled PHA. The information about uncontrolled PHA is provided to safety managers so they can establish proper safety countermeasures and manage the area. Morevoer, these movement characteristics (obstacle avoidance) can be used in other construction management areas such as material arrangement, evacuation planning, etc.

(5) Current location-sensing techniques require additional efforts (e.g. installing reference tags and more readers, and pre-mapping of wave, etc.) for improving accuracy. This study suggested a method for improving accuracy workers' location sensing based on the maximum velocity filtering method. This method focuses only on workers' location sensing. Therefore, the algorithm of the method was developed considering workers' movement characteristics. Moreover, it does not require installation of any additional device, which means that it does not interfere with construction activities.

(6) In safety management, it is important to decrease the time workers

are exposed to a hazard. The suggested system is automatically operated, so that it can identify a hazard quickly. This means that the time a hazard exists from when it is first generation to when it is identified diminishes. Consequently, the suggested system can decrease the exposure time of hazards and help prevent an accident from occurring.

6.2 Limitations and Future Research

Although it must be noted that this study is based on a limited scope, additional research and tests must be conducted to further validate the suggested systems and generalize them by overcoming their limitations. Regarding this, this study suggests the future research areas as follows.

(1) To operate the activity-oriented hazard identification system, project information data is required as well as accident cases. If there is no abundant data source such as PMIS, enormous efforts are required to input data. Therefore, future research should take measures to obtain data source easily. Moreover, the activity-oriented hazard identification system does not provide visualized information system. Although it provides coordinates of identified hazards and gives similar past accident cases to the most related workers, there are additional efforts to compare coordinates of those and drawings.

Interworking between 3D models and the coordination of the retrieval result is needed to visualize the risk area or factors. If visualized information is provided, safety managers easily recognize what and where is risky and workers will be also able to better deal with the expected risk.

(2) Non-activity-oriented hazard identification system focuses on identifying hazards on movement path. It can find hazards only on the optimal routes. However, all the hazards are not on the optimal route. Although hazards on the optimal routes have higher possibility of accident occurrence, hazards on other places can also cause an accident. Thus, hazard identification should be performed in not only on optimal routes but also other places. Although the non-activity-oriented hazard identification system proved its applicability and effectiveness for real-time location tracking in indoor environments, the performance of the system needs to be further validated in many different construction environments.

(3) Because of wave propagation attenuation—the gradual loss in intensity of radio-frequency waves by obstacles or conditions—workers' real locations do not accurately correspond to the result of RTLS. When the indoor location system is fully developed, the accuracy of potentially hazardous areas is expected to increase. In further studies analyzing workers' movements and behavior patterns,

the scope of hazardous area identification will extend to hazards throughout a whole construction site.

Bibliography

Abudayyeh, O., Federricks. T., Palmquist, M., and Torres, H. (2003) “Analysis of Occupational Injuries and Fatalities in Electrical Contracting.” *J. Constr. Eng. Manage*., 129(2), 152-158.

Akinci B., Fischer, M., and Kunz, J. (2002). “Automated Generation of Work Spaces Required by Construction Activities.” *J. Constr. Eng. Manage*., 128(4), 306-315.

Ahn, H.(1994). “A Knowledge Model for Enhancing Construction Hazard Communication.” Ph.D. dissertation, Seoul National Univ., Seoul, Korea.

Ahuja, R., Mehlhorn, K., Orlin, J., and Tarjan, R. (1990). “Faster Algorithms for the Shortest Path Problem.” *Journal of the Association for Computing Machinery*, 37(2), 213-223.

An, S. (2007). "Integrated safety risk management for construction projects." Ph.D. dissertation, Seoul National Univ., Seoul, Korea.

Bahl, P., Padmanabhan V.N., and Balachandran, A. (2000). *Enhancements to the RADAR user location and tracking system*, Microsoft Research Technical Report, http://research.microsoft.com/～bahl/papers/pdf/msr-tr-2000-12.pdf

Bahr, N. J. (1997). *System Safety Engineering and Risk Assessment: A Practical Approach*, CRC Press, New York, USA.

Bai, Y. and Amirkhanian, S. (1994). ”Knowledge-Based Expert System for Concrete Mix Design.” *J. Constr. Eng. Manage*., 120(2), 357–373.

Baradan, S., and Usmen, M.A. (2006). "Comparative Injury and Fatality Analysis of Building Trades.", *J. Constr. Eng. Manage*., 132(5), 533~539

Blakey, L. (1990). "Bar Codes: prescription, performance and productivity.", *J. Constr. Eng. Manage*., 116(3), pp. 468-479.

Blackmon, R. and Gramopadhye, A. (1995). ”Improving Construction Safety by Providing Positive Feedback on Backup Alarms.” *J. Constr. Eng. Manage*., 121(2), 166–171.

British Standard Institute (BSI). (1996). *Guide to Occupational Health and Safety Management Systems*., BS 8800, London

Carter, G. and Smith., S. D. (2006). "Safety hazard identification on construction projects." *J. Constr. Eng. Manage*., 132(2), 197-205.

Catino, C. A. and Ungar, L. H. (1995) “Model based approach to automated

hazard identification of chemicals." *AiChe Journal*, 41(1), 97-109.

Chae, S. and Yoshida, T. (2010). "Application of RFID technology to prevention of collision accident with heavy equipment." *Automation in Construction*, 19(3), 368–374.

Cheng, T. and Teizer, J. (2010). "Real-Time Data Collection and Visualization Technology in Construction." *Proc., Construction Research Congress*, ASCE, Reston, VA, 339–348.

Chin, S., Yoon, S., Choi, C., and Cho, C. (2008). "RFID+4D CAD for Progress Management of Structural Steel Works in High-Rise Buildings." *J. Comput. Civ. Eng.*, 22(2), 74–89.

Chitkara, K. K. (1998). *Construction project management: planning, scheduling and controlling*, Tata McGraw-Hill, New Delhi, India.

Cho, H., Jung, Y., Choi, H., Jang, H., Son, S., & Baek, Y. (2008, September). "Precise location tracking system based on time difference of arrival over LR-WPAN.", *In Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*, ACM, 67–72.

Choudhry, R. M., Fang, D., Lingard, H. (2009). "Measuring Safety Climate of

a Construction Company." *J. Constr. Eng. Manage*., 135(9), 890-899.

Chua, D. K. H. and Goh, T. M. (2002). "Application of case based reasoning in construction safety planning." *Proceedings of International Workshop on IT in Civil Engineering 2002*, 298-307.

Chua, D. K. H. and Loh, P. (2006). "CB-Contract: Case-Based Reasoning Approach to Construction Contract Strategy Formulation." *J. Comput. Civ. Eng*., 20(5), 339–350.

Chung, P. W. H., and Jefferson, M. (1998). "The integration of accident databases with computer tools in the chemical industry." *Computers & Chemical Engineering*, 22(1), S729–S732.

Clote, P. and Kranakis, E. (2002). *Boolean Functions and Computation Models*, Springer, New York, USA.

Davidson I. N. and Skibniewski, J. (1995). "Simulation of automated data collection in buildings." *J. Comp. Civ. Eng*., 9(1), pp. 9-20.

Demian, P. and Balatsoukas, P. (2012). "Information Retrieval from Civil Engineering Repositories: Importance of Context and Granularity." *J. Comput. Civ. Eng*., 26(6), 727–740.

Diekmann, J. and Gjertsen, K. (1992). “Site Event Advisor: Expert System for Contract Claims.” *J. Comput. Civ. Eng*., 6(4), 472–479.

Dijkstra, E. W. (1959). “A note on two problems in connexion with graphs.”, Numrische Mathematik, 1(1), 269-271.

Ding, L. Y., Yu, H. L., Li, H., Zhou, C., Wu, X. G., and Yu, M. H. (2012). “Safety risk identification system for metro construction on the basis of construction drawings.” *Automation in Construction*, 27, 120-137.

Dorigo, A. and Gambardella, L. M. (1997). “Ant Colony System: A Cooperative Learning Approach to the Travelling Salesman Problem.”, *IEEE Transactions on Evolutionary Computation*, 1(1), 53-66.

Dziadak, K., Kumar, B., and Sommerville, J. (2009) "Model for the 3D location of buried assets based on RFID technology." *J. Comp. Civ. Eng*., 23(3), pp. 148-159.

El-Diraby, T., Lima, C., and Feis, B. (2005). ”Domain Taxonomy for Construction Concepts: Toward a Formal Ontology for Construction Knowledge.” *J. Comput. Civ. Eng*., 19(4), 394–406.

Elson, J., Girod, L., and Estrin, D. (2002). “Fine-grained network time synchronization using reference broadcasts.”, *ACM SIGOPS Operating*

Systems Review, 36(SI), 147-163.

Frank, M. V. (2007). *Choosing Safety, Resource for the Future*, Washington, DC, USA.

Fredericks, T., Abudayyeh, O., Choi, S., Wiersma, M., and Charles, M. (2005). ”Occupational Injuries and Fatalities in the Roofing Contracting Industry.” *J. Constr. Eng. Manage*., 131(11), 1233–1240.

Fullerton, C., Allread, B., and Teizer, J. (2009). Pro-Active-Real-Time Personnel Warning System. *Construction Research Congress 2009*,. 31–40.

Ganeriwal, S., Kumar, R., and Srivastava, M. B. (2003). “Timing-sync protocol for sensor networks.” *In Proceedings of the 1st international conference on Embedded networked sensor systems*, ACM, 138-149.

Garmin Corporation, *About GPS*, http://www8.garmin.com/aboutGPS/

Goh, T. M. and Chua, D. K. H. (2010). "Case-based reasoning approach to construction safety hazard identification: adaptation and utilization." *J. Constr. Eng. Manage*., 136(2), 170-178.

Gowen, L. D. and Collofello, J. S. (1994). "Software safety and preliminary

hazard analysis." *Prof. Saf.*, November 1994, pp. 20-25.

Hallowell, M. R. and Gambatese, A. (2009). "Activity-based safety risk quantification for concrete formwork construction." *J. Constr. Eng. Manage*., 135(10), pp. 990-998.

Hallowell, M., Teizer, J., and Blaney, W. (2010). “Application of Sensing Technology to Safety Management.”, *CRC 2010*, 31-40.

Hamil, S. (2013). “BIM Objects from Manufacturers” *Construction Code*, <http://constructioncode.blogspot.co.uk/2013/01/bim-objects-from-manufacturers.html〉(Jan. 24, 2013)

Han, J., Zhao, Y., Cheng, Y. S., Wong, T. L., and Wong, C. H. (2012). “Improving Accuracy for 3D RFID Localization,” *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 865184, 9 pages, 2012. doi:10.1155/2012/865184

Harter, A., Hopper, A., Steggles, P., Ward, A., and Webster, P. (1999). “The anatomy of a context-aware application.” *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom 1999)*, Seattle, WA (ACM Press, August 1999), 59–68.

Health and Safety Commission. (1995) *Designing for health and safety in construction*, HSE Books, London.

Health and Safety Executive (http://www.hse.gov.uk/statistics/industry/construction/index.htm)

Heinrich, H. W. (1941). A Scientific Approach. *Industrial Accident Prevention. A Scientific Approach*., (Second Edition)., McGraw Hill, New York

Jannadi O. A., and Almishari S. (2003). "Risk assessment in construction." *J. Constr. Eng. Manage*., 129(5), pp. 492-499.

Jansen, B. and Pooch, U. (2001). “Web user studies: A review and framework for future work.” *Journal of the American Society of Information Science and Technology*, 52(3), 235-246.

Jaselskis, E. J., El-Misalami, T. (2003). "Implementing radio frequency identification in the construction process." *J. Constr. Eng. Manage*., 129(6), pp. 680-688.

Jin, G. Y., Lu, X. Y., and Park, M. S. (2006, June). "An indoor localization mechanism using active RFID tag," *in IEEE Int. Conf. on Sensor Networks, Ubiquitous, and Trustworthy Computing*, Taichung, Taiwan, June 2006, pp. 1–4.

Jukna, S. (2012). *Boolean Function Complexity: Advances and Frontiers*. Springer, New York, USA.

Karshenas, S., and Tse, J. (2002). “A case-based reasoning approach to construction cost estimating.” *Proc., Int. Workshop on Information Technology in Civil Engineering 2002*, ASCE, Reston, VA, 113-123.

Kartram, N. A. (1997). "Integrating safety and health performance into construction CPM." *J. Constr. Eng. Manage*., 123(2), 121-126.

Khemlani, L. (2007). *Top Criteria for BIM Solutions: AECbytes Survey Results* (online press), <http://www.aecbytes.com/feature/2007/BIMSurveyReport.html〉(Feb. 20, 2013)

Ko, S., Song, H., and Lee, H. (2005). “Development of the Safety Information Management System according to the Risk Index for the Building Construction Work.” *Journal of Architectural Institute of Korea*, 21(6), 113-120.

Kolodner, J. L. (1992). “An introduction to case-based reasoning.” *Artificial Intelligence Review*, 6(1), 3-34.

Koo, C., Hong, T., Hyun, C., and Koo, K. (2010). “A CBR-based hybrid

model for predicting a construction duration and cost based on project characteristics in multi-family housing projects." *Can. J. Civ. Eng*., 37(5), 739-752.

Korea Occupational Safety and Health Agency (1999) *Research Report: A Study on the Danger Index of the Construction Works* (S-RD-I-2000-20-40), Incheon, Korea.

Korea Occupational Safety and Health Agency (2001~2003) *Annual Report: Fatal Accident Cases*, Seoul, Korea

Langley, R. B. (1999). Dilution of precision. GPS world, 10(5), 52-59.

Lee, H., Lee, K., Park, M., Kim, H., and Lee, S. (2009). "A construction safety management system based on building information modeling and real-time locating system." *J. KICEM*., 10(6), pp.135-145.

Lee, H., Kim, H., Park, M., Teo, E. A. L., and Lee, K. (2012a). "Construction risk assessment using site influence factors." *J. Comput. Civ. Eng*., 26(3), 319-330.

Lee, H., Lee, K., Park, M., Baek, Y., and Lee, S. (2012b). "RFID-Based Real-Time Locating System for Construction Safety Management." J. *Comput. Civ. Eng*., 26(3), 366-377.

Lee, S., and Halpin, D. (2003). “Predictive Tool for Estimating Accident Risk.” *J. Constr. Eng. Manage*., 129(4), 431-436.

Lei, Z., Taghaddos, H., Olearczyk, J., Al-Hussein, M., and Hermann, U. (2013). "Automated Method for Checking Crane Paths for Heavy Lifts in Industrial Projects." *J. Constr. Eng. Manage*, 139(10), Online publication date: 1-Oct-2013.

Li, H., Chan, G., and Skitmore, M. (2012). “Visualizing safety assessment by integrating the use of game technology.” *Automation in Constrcution*, 22 (2012) 498-505.

Lin, K. and Soibelman, L. (2009). "Incorporating domain knowledge and information retrieval techniques to develop and architectural/engineering/construction online product search." *J. Comput. Civ. Eng*., 23(4), 201-210.

Lindberg, A., Hansson, S. O., and Rollenhagen, C. (2010). “Learning from accidents – What more do we need to know?” *Safety Science*, 48(6), 714–721.

Lu, M., and Shen, X., Chen, W. (2009). "Automated collection of mixer truck operations data in highly dense urban areas." *J. Constr. Eng. Manage*.,

135(1), pp. 17-23.

MacCollum, D., V. (2007). *Construction Safety Engineering Principles: Designing and Managing Safer Job Sites*., McGraw-Hill, London

Mahfouz, T. S. (2009). “Construction legal support for differing site conditions (DSC) through statistical modeling and machine learning.” Ph.D. dissertation, Iowa State University, US.

Manning, C. D., Raghavan, P. and Schutze, H. (2008). *Introduction to information retrieval*, Cambridge University Press, New York, USA

Maróti, M., Kusy, B., Simon, G., and Lédeczi, Á. (2004). “The flooding time synchronization protocol.”, *In Proceedings of the 2nd international conference on Embedded networked sensor systems*, ACM, 39-49.

McKinney, K. and Fischer, M. (1998). "Generating, evaluating and visualizing construction schedules with CAD tools." *Automation in Construction*, 7(6), pp. 433-447.

Meso, P. and Smith, R. (2000). "A resource-based view of organizational knowledge management systems." *Journal of Knowledge Management*, 4(3), 224-234.

Mitropoulos, P., Cupido, G., and Namboodiri, M. (2009). “Cognitive Approach to Construction Safety: Task Demand-Capability Model.”, *J. Constr. Eng. Manage*., 135(9), 881–889.

Mohamed, S. (2002). "Safety Climate in Construction Site Environments." *J. Constr. Eng. Manage*., 128(5), 375-384.

Moon, M., Kim, E., and Yang, K. (1997). “A Study on the Implementation of the Accident Information Management System in Constructions.” *Journal of Architectural Institute of Korea*, 13(8), 429-440.

Morcous, G., Rivard, H., and Hanna, A. (2002). ”Case-Based Reasoning System for Modeling Infrastructure Deterioration.” *J. Comput. Civ. Eng*., 16(2), 104–114.

Navon, R. and Kolton, O. (2006). "Model for automated monitoring of fall hazards in building construction." *J. Constr. Eng. Manage*., 132(7), pp. 733-740.

Navon, R. and Kolton, O. (2007). "Algorithms for automated monitoring of fall hazards." *J. Comp. Civ. Eng*., 21(1), pp. 21-28.

Ni, L. M., Liu, Y., Lau, Y. C., and Patil, A. P. (2004). “LANDMARC: indoor location sensing using active RFID.” *Wireless networks*, 10(6), 701–710.

O'Keefe, R. M. and O'Laery, D. E. (1993). "Expert system verification and validation: a survey and tutorial." *Artificial Intelligence Review*, 7(1), 3-42.

Oloufa, A., Ikeda, M., and Oda, H. (2003). "Situational awareness of construction equipment using GPS, wireless and web technologies." *Automation in Construction*., 12(6), pp. 737-748.

Ortolano, L., Le Coeur, G., and MacGilchrist, R. (1990). "Expert System for Sewer Network Maintenance: Validation Issues." *J. Comput. Civ. Eng.*, 4(1), 37–54.

O'Toole, M. (2002). "The Relationship between Employees' perceptions of safety and organizational culture.", *Journal of Safety Research*, 33, 231-243.

Park, S. H., Kim, Y. J., Lee, H. C., & Lim, M. T. (2008, August). "Improved adaptive particle filter using adjusted variance and gradient data." *In Multisensor Fusion and Integration for Intelligent Systems, MFI 2008*. IEEE International Conference on (pp. 650–655). IEEE.

Paulson, B. C. (1995). "Computers and construction - Midcareer reflections." *J. Constr. Eng. Manage*., 121(2), 197-200.

Pocchiola, M. and Vegter, G. (1995). "Computing Visibility Graphs via Pseudo-triangulations.", *In Proc. 11th Annu. ACM Sympos. Comput. Geom.*, 1-34.

Priyantha, N.B., Chakraborty, A., and Balakrishnan, H., (2000). “The Cricket location-support system.” *Proceedings of MOBICOM 2000*, Boston, MA, ACM Press, 32–43.

Rappaport, T. S. (1996). *Wireless communications: principles and practice (Vol. 2)*. New Jersey: Prentice Hall PTR.

Rasdorf, W. and Herbert, M. J. (1990). "Bar coding in construction engineering." *J. Constr. Eng. Manage*., 116(2), pp. 261-280.

Rezgui, Y. (2006). ”Ontology-Centered Knowledge Management Using Information Retrieval Techniques.” *J. Comput. Civ. Eng*., 20(4), 261–270.

Rilett, L. R. and Park, D. (2001). "Incorporating uncertainty and multiple objectives in real-time route selection." *J. Transp. Engrg*. 127(6). pp.531-539

Ryu, H., Lee, H., and Park, M. (2007). ”Construction Planning Method Using Case-Based Reasoning (CONPLA-CBR).” *J. Comput. Civ. Eng*., 21(6),

410–422.

Rohrig, C., and Muller, M. (2009). “Indoor location tracking in non-line-of-sight environments using a IEEE 802.15. 4a wireless network.” *In Intelligent Robots and Systems*, 2009. IROS 2009. IEEE/RSJ International Conference on, IEEE, 552-557.

Satty, T. L. (1980). *The analytic hierarchy process*, McGraw-Hill, New York. USA.

Sacks, R., Rozenfeld, O. and Rozenfeld, Y. (2009) "Spatial and temporal exposure to safety hazards in construction." *J. Constr. Eng. Manage*., 135(8), pp. 726-736.

Saidi, K. S., Teizer, J., Franaszek, M., and Lytle, A. M. (2011). “Static and dynamic performance evaluation of a commercially-available ultra wideband tracking system.” *Automation in Construction*, 20(5), 519–530.

Schatterman, D., Herroelen, W., Van de Vonder, S., Boone, A. (2008). "Methodology for integrated risk management and proactive scheduling of construction projects." *J. Constr. Eng. Manage*., 134(11), pp. 885~893.

Shepherd, A. and Ortolano, L. (1996). “Water-Supply System Operations: Critiquing Expert-System Approach.” *J. Water Resour. Plann. Manage*.,

122(5), 348–355.

Skibniewski, M. J., and Jang, W. S. (2007). “Localization Technique for Automated Tracking of Construction Materials Utilizing Combined RF and Ultrasound Sensor Interfaces.” *Proc., Computing in civil Engineering 2007*, ASCE, Reston, Va., 657-664.

Son, S., Park, Y., & Baek, Y. (2013) “Design and Implementation of Enhanced Asset Locating System for Vehicle Pooling in a Port Terminal.” *Proc., 2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA)*, Barcelona, 710–715.

Song, J., Haas, C., and Caldas, C. (2006). “Tracking the Location of Materials on Construction Job Sites.” *J. Constr. Eng. Manage*., 132(9), 911–918.

Suh, J. C., Lee, S., and Yoon, E. S. (1997a). “New strategy for automated hazard analysis of chemical plants. Part 1: knowledge modeling.” *J. Loss Prevention in the Process Industries*., 10(2), 113-126

Suh, J. C., Lee, S., and Yoon, E. S. (1997b). “New strategy for automated hazard analysis of chemical plants. Part 2: reason algorithm and case study.” *J. Loss Prevention in the Process Industries.*, 10(2), 127-134

Tah, J. M., Carr, V., and Howes, R. (1998). “An application of case-based

reasoning to the planning of highway bridge construction." *Eng. Constr. Archit. Manage.*, 5(4), 327-338.

Teizer, J. and Castro-Lacouture, D. (2007). "Combined Ultra-Wideband Positioning and Range Imaging Sensing for Productivity and Safety Monitoring in Building Construction." *Proc., 2007 ASCE Int. Workshop on Computing in Engineering*, ASCE, Reston, VA, 681–688.

Teizer, J., Caldas, C., and Haas, C. (2007). "Real-Time Three-Dimensional Occupancy Grid Modeling for the Detection and Tracking of Construction Resources." *J. Constr. Eng. Manage.*, 133(11), 880–888.

Torrent, D. and Caldas, C. (2009). "Methodology for Automating the Identification and Localization of Construction Components on Industrial Projects." *J. Comput. Civ. Eng.*, 23(1), 3–13.

Woldesenbet, A. and Jeong, H. S. (2012). "Historical Data Driven and Component Based Prediction Models for Predicting Preliminary Engineering Costs of Roadway Projects." *Construction Research Congress 2012*, 417-426.

Wu, W., Yang, H., Chew, D. A. S., Yang, S., Gibb, A. G. F, and Li, Q. (2010). "Towards an autonomous real-time tracking system of near-miss accidents on construction sites." *Automation in Construction*, 19(2),

134–141.

Yau, N. J. and Yang, J. B. (1998). “Case-based reasoning in construction management.” *Comput. Aided Civ. Infrastruct. Eng*., 13(2), 143-150.

Ye, T. (1998). “Reasoning Model of the Case-Based Construction Safety Management System.” Master Course. Dissertation, Seoul National Univ., Seoul, Korea.

Yeh, K., Tsai, M., and Kang, S. (2012). ”On-Site Building Information Retrieval by Using Projection-Based Augmented Reality.” *J. Comput. Civ. Eng*., 26(3), 342–355.

Yelamarthi, K., Haas, D., Nielsen, D., and Mothersell, S. (2010). “RFID and GPS integrated navigation system for the visually impaired” *Proceedings of 2010 53rd IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)*, 1149-1152.

Yves, C. and Hammer, P. L. (2010). *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*, Cambridge University Press, London.

Zhang, L., Wang, Q., and Nie, G. (2009). “Study on an Intelligent Knowledge Push Method for Knowledge Management System.” *Communications in*

Computer and Information Science, 35(5), 202-208.

Zobel, J., Moffat, A, and Ramamohanarao, K. (1996). “Guidelines for presentation and comparison of indexing techniques.” *ACM SIGMOD Record*, 25(3), 10-15.

Appendices

Appendix A: Glossary of Acronyms

Appendix B: Accident Case Sample Data

Appendix C: The Source Code of Non-activity Oriented Hazard Identification System

Appendix A: Glossary of Acronyms

Acronyms	Fullname
NIOSH	National Institute for Occupational Safety and Health
KOSHA	Korea Occupational Safety and Health Agency
IT	Intelligent Technology
BIM	Building Information Model
RFID	Radio Frequency Identification
OSHA	Occupational Safety and Health Administration
HSE	Health and Safety Executive
CBR	Case-Based Reasoning
RTLS	Real-Time Location Tracking System
GPS	Global Positioning System
AOA	Angle of Arrival
RSSI	Received Signal Strength Indication
TOA	Time Of Arrival
TDOA	Time Difference Of Arrival
UWB	Ultra Wide Band
NLOS	Non-Line Of Sight
LOS	Line Of Sight
AHP	Analytic Hierarchy Process
SI	Similarity Index
SS	Similarity Score
PMIS	Project Management Information System
GDL	Geometric Description Language
VBA	Visual Basic for Application
PHA	Potentially Hazardous Area
RBS	Reference Broadcast Synchronization
TPSN	Timing-sync Protocol for Sensor Network

RTSP	Flooding Time Synchronization Protocol
GDOP	Geometric Dilution Of Precision
DOP	Dilution Of Precision
UI	User Interface
TREC	Text REtrieval Conference

Appendix B: Accident Case Sample Data

[Accident Case Database]

The data for assessing risk was collected from 4 general contractors who provided information on accident cases over 5 years (2003~2007). The total number of cases is 596, with 544 non-fatalities and 79 fatalities.

Note: (X1) Work type, (X2) Work process rate (%), (X3) Cost of the construction (hundred million won), (X4) Type of building, (X5) Age, (X6) Type of Occuaption (Worker), (X7) Work days on current site, (X8) Safety Training (number of times), (X9) Date (Month), (X10) Temperature (℃), (SS) Social and sports facilities, (HA) Housing accommodation, (HS) Hospital and school facilities, (Apt) Apartment, (CP) Commercial public facilities

No.	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10
1	R/C	54	7	SS	61	Carpenter	2~3 months	5~6	6	20.9
2	Steel Structure	48	43	Apt	52	Brick	3~4 months	7~8	7	29.8
3	R/C	100	1046	CP	20	R/C	10~20 days	1	3	7.1
4	E/V	27	515	SS	40	Concrete finish	Over 1 years	21~	10	6
5	Window	91	23	Apt	51	Plumber	6~12 months	16~20	8	34.2
6	E/V	30	1	HA	37	Earth	Over 1 years	21~	10	7.8
7	R/C	86	629	HA	44	Plaster	2~3 months	5~6	5	17.3
8	Steel Structure	19	4	CP	64	Common	3~4 months	7~8	7	30
9	Temporary	95	3	CP	30	R/C	~10 days	0	1	-9.1

10	R/C	68	1068	Apt	44	R/C	10~20 days	1	3	15
11	R/C	8	44	SS	24	Paint	1~2 months	3~4	5	15.2
12	R/C	23	527	HS	39	R/C	10~20 days	1	3	14.1
13	R/C	21	35	CP	29	Framing	1~2 months	3~4	4	16.8
14	Etc.	86	48	HA	30	Tile	6~12 months	16~20	10	0.4
15	R/C	67	519	HS	51	Framing	1~2 months	3~4	4	16.2
16	R/C	75	32	CP	31	Framing	1~2 months	3~4	4	10.3
17	Temporary	14	5	HS	47	Scaffold	~10 days	0	1	-7.6
18	Earth	7	29	SS	54	R/C	~10 days	0	2	-1.3
19	Etc.	59	1021	HA	23	Concrete finish	6~12 months	16~20	10	9.7
20	R/C	64	48	HS	39	Framing	1~2 months	3~4	4	11.3
21	Temporary	78	1	Apt	45	Scaffold	~10 days	0	1	-1.1
22	R/C	32	21	HA	49	Roof	20~30 days	2	3	12.5
23	E/V	63	3	HS	25	Concrete finish	Over 1 years	21~	10	9.3
24	Metal	4	20	HS	31	Common	4~5 months	9~10	8	34.7
25	E/V	11	4	SS	44	Concrete finish	Over 1 years	21~	10	2.6
26	R/C	72	30	HS	27	R/C	10~20 days	1	3	11
27	Window	15	5	CP	54	Plumber	6~12 months	16~20	8	36.7
28	Curtain Wall	56	34	HA	66	Earth	Over 1 years	21~	10	4.8
29	R/C	74	4	HA	52	Paint	1~2 months	3~4	5	24.4
30	Steel Structure	73	518	Apt	32	Common	3~4 months	7~8	7	30.4
31	R/C	85	32	HA	26	Finishing	2~3 months	5~6	7	27.8
32	Foundation	80	3	SS	66	R/C	~10 days	0	2	-7.4
33	Temporary	15	54	SS	66	Scaffold	~10 days	0	1	-4.2
34	Foundation	93	686	CP	59	R/C	10~20 days	1	2	-4.1
35	Plastering	55	50	HS	53	Common	5~6 months	11~15	8	33.5

36	Window	94	2	Apt	47	Plumber	6~12 months	16~20	8	36.7
37	Foundation	66	51	SS	54	R/C	~10 days	0	2	0
38	Window	11	53	HA	47	Re-bar	5~6 months	11~15	8	27.1
39	R/C	1	676	Apt	39	Framing	1~2 months	3~4	4	11.3
40	R/C	55	616	HA	54	R/C	10~20 days	1	3	8.5
41	Foundation	75	50	Apt	35	R/C	10~20 days	1	2	1.7
42	R/C	36	9	SS	20	Framing	1~2 months	3~4	4	15.1
43	Plastering	10	41	HS	45	Watertight	5~6 months	11~15	8	31.9
44	R/C	83	35	SS	63	Paint	1~2 months	3~4	5	20.2
45	R/C	37	4	Apt	68	Electronic	2~3 months	5~6	6	24.1
46	Foundation	18	7	Apt	20	R/C	~10 days	0	2	0.5
47	Window	79	173	Apt	41	Re-bar	5~6 months	11~15	8	28.4
48	Earth	77	1181	SS	43	R/C	~10 days	0	2	-7.5
49	R/C	21	9	CP	66	Framing	20~30 days	2	4	16.9
50	R/C	56	1	Apt	35	Plaster	2~3 months	5~6	6	25.4
51	R/C	7	35	SS	60	R/C	10~20 days	1	3	13.9
52	Temporary	34	2	HS	39	Scaffold	~10 days	0	1	-5.7
53	Window	75	4	SS	21	Re-bar	5~6 months	11~15	8	30
54	Etc.	99	2	CP	29	Concrete finish	6~12 months	16~20	10	4.2
55	R/C	49	3	HA	63	Plaster	1~2 months	3~4	5	18.9
56	R/C	67	52	CP	26	Finishing	2~3 months	5~6	7	34
57	Earth	22	1	CP	40	R/C	~10 days	0	2	-5.1
58	R/C	22	42	HS	20	Finishing	3~4 months	7~8	7	31.4
59	R/C	23	152	HS	57	Paint	1~2 months	3~4	5	22
60	Etc.	69	637	HS	68	Tile	6~12 months	16~20	10	8.4
61	Window	96	144	CP	32	Re-bar	5~6 months	11~15	8	34.5

62	Temporary	13	148	HA	32	Scaffold	~10 days	0	1	-5.1
63	Window	26	55	CP	25	Re-bar	5~6 months	11~15	8	29.4
64	Window	57	249	CP	63	Re-bar	5~6 months	11~15	8	37
65	R/C	6	236	HS	32	R/C	20~30 days	2	3	5.3
66	Plastering	88	41	Apt	26	Common	5~6 months	11~15	8	34
67	R/C	77	232	HA	59	Electronic	2~3 months	5~6	6	24
68	Earth	73	43	HA	35	R/C	~10 days	0	2	-6.8
69	R/C	99	4	HA	58	R/C	10~20 days	1	3	5.5
70	Stone	28	104	CP	62	Common	4~5 months	9~10	7	29.8
71	Plastering	82	142	SS	41	Common	5~6 months	11~15	8	35.5
72	Curtain Wall	68	30	SS	69	Earth	Over 1 years	21~	10	0.4
73	R/C	28	3	CP	65	R/C	20~30 days	2	3	6.3
74	R/C	71	4	SS	58	Electronic	2~3 months	5~6	6	23.9
75	Temporary	97	4	SS	26	Scaffold	~10 days	0	1	-6.4
76	Earth	84	189	HA	49	R/C	~10 days	0	2	-3.8
77	R/C	60	49	HS	44	R/C	10~20 days	1	3	8.4
78	Finish	76	553	HA	63	Plumber	6~12 months	16~20	9	11
79	Window	65	34	HS	55	Plumber	6~12 months	16~20	8	36.6
80	Temporary	35	50	HS	53	Scaffold	~10 days	0	1	-3.8
81	Foundation	27	541	HS	67	R/C	~10 days	0	2	-4.2
82	R/C	69	1121	HA	29	Roof	20~30 days	2	3	8.8
83	Stone	39	1081	CP	59	Common	4~5 months	9~10	7	34.2
84	R/C	5	236	Apt	34	R/C	10~20 days	1	3	13.9
85	Foundation	43	238	Apt	54	R/C	10~20 days	1	2	-2
86	Tile	42	52	Apt	52	Common	4~5 months	9~10	8	28.8
87	Etc.	75	1191	Apt	38	Tile	6~12 months	16~20	10	8.1

88	R/C	18	34	CP	47	R/C	10~20 days	1	3	13.3
89	Metal	68	36	CP	24	Common	4~5 months	9~10	8	34.9
90	R/C	46	1120	CP	29	Paint	1~2 months	3~4	5	17.4
91	R/C	40	34	SS	67	Roof	20~30 days	2	3	11.7
92	Steel Structure	75	45	HS	33	Brick	3~4 months	7~8	7	35.4
93	Stone	37	52	HS	51	Common	4~5 months	9~10	7	36.8
94	Plastering	28	38	HS	54	Common	5~6 months	11~15	8	31.8
95	Steel Structure	54	1132	SS	55	Brick	3~4 months	7~8	7	35.6
96	Stone	57	231	HS	41	Common	4~5 months	9~10	7	35.7
97	Steel Structure	13	2	CP	27	Sheet Metal	3~4 months	7~8	7	28.5
98	Steel Structure	81	1056	CP	20	Sheet Metal	3~4 months	7~8	7	28.6
99	R/C	93	1195	Apt	32	Roof	20~30 days	2	3	6.2
100	R/C	83	588	Apt	50	Roof	20~30 days	2	3	6.2
101	R/C	31	19	CP	34	Plaster	1~2 months	3~4	5	21.5
102	Plastering	32	1192	Apt	25	Common	5~6 months	11~15	8	28.3
103	Plastering	73	108	CP	54	Watertight	5~6 months	11~15	8	31.9
104	R/C	64	644	Apt	65	Framing	1~2 months	3~4	4	18.6
105	Plastering	27	11	Apt	67	Watertight	5~6 months	11~15	8	31.3
106	Plastering	93	51	HA	60	Common	5~6 months	11~15	8	34.6
107	Foundation	100	190	SS	26	R/C	~10 days	0	2	-4.6
108	Earth	31	208	SS	32	R/C	~10 days	0	2	-7.8
109	Temporary	29	53	CP	41	Scaffold	~10 days	0	1	-2.3
110	R/C	32	4	HA	57	Electronic	2~3 months	5~6	6	28
111	Window	72	25	CP	65	Re-bar	5~6 months	11~15	8	29.8
112	Etc.	68	2	HA	24	Tile	6~12 months	16~20	10	2.5
113	Plastering	81	3	HS	62	Common	5~6 months	11~15	8	27

114	R/C	64	1	HA	49	Finishing	2~3 months	5~6	6	26.6
115	Steel Structure	55	17	SS	52	Brick	3~4 months	7~8	7	28.9
116	Foundation	85	5	HA	35	R/C	~10 days	0	2	1.9
117	E/V	57	5	HS	36	Earth	Over 1 years	21~	10	6.3
118	R/C	70	40	HS	41	R/C	10~20 days	1	3	10.8
119	Steel Structure	14	34	HA	25	Common	3~4 months	7~8	7	30.4
120	R/C	56	21	SS	23	Plaster	2~3 months	5~6	5	17.7
121	R/C	7	1076	Apt	51	Paint	1~2 months	3~4	5	19
122	R/C	9	2	Apt	63	Plaster	2~3 months	5~6	5	15.6
123	R/C	44	554	HS	23	Framing	1~2 months	3~4	5	21
124	R/C	73	4	CP	54	Brick	3~4 months	7~8	7	31
125	Stone	38	24	HA	24	Common	4~5 months	9~10	7	29.3
126	Steel Structure	74	3	Apt	62	Sheet Metal	3~4 months	7~8	7	30.5
127	Dismantling	4	12	CP	30	Tile	6~12 months	16~20	9	13.2
128	Foundation	85	14	HS	53	R/C	~10 days	0	2	-7
129	R/C	89	1030	HA	56	Paint	1~2 months	3~4	5	19.8
130	Metal	8	1038	CP	45	Common	4~5 months	9~10	8	32.9
131	Wall	65	26	Apt	28	Common	4~5 months	9~10	8	35.3
132	R/C	53	5	CP	29	R/C	10~20 days	1	3	14.5
133	R/C	75	54	SS	41	Finishing	2~3 months	5~6	6	21.9
134	R/C	76	1152	HA	65	Plaster	2~3 months	5~6	6	24.3
135	Window	56	1081	HA	26	Re-bar	5~6 months	11~15	8	31.8
136	Dismantling	15	149	CP	59	Plumber	6~12 months	16~20	9	20
137	Wall	59	36	SS	46	Common	4~5 months	9~10	8	33.3
138	Etc.	30	9	HA	28	Tile	6~12 months	16~20	10	8.3
139	Earth	70	217	Apt	34	R/C	~10 days	0	2	-1.1

140	R/C	58	9	HS	42	Plaster	2~3 months	5~6	5	17
141	Plastering	82	1190	HA	45	Watertight	5~6 months	11~15	8	29.7
142	Wall	67	249	CP	39	Common	4~5 months	9~10	8	33.5
143	R/C	90	1110	Apt	51	Plaster	2~3 months	5~6	5	18.9
144	Temporary	25	8	HS	60	Scaffold	~10 days	0	1	-7.2
145	R/C	59	2	SS	50	Roof	20~30 days	2	3	5.9
146	Foundation	85	11	HS	62	R/C	~10 days	0	2	-6.2
147	Plastering	43	236	Apt	27	Watertight	5~6 months	11~15	8	28.4
148	R/C	65	3	HA	46	Roof	20~30 days	2	3	9.6
149	Temporary	72	42	Apt	23	Scaffold	~10 days	0	1	-8.6
150	R/C	80	1038	SS	21	Roof	20~30 days	2	3	13.2
151	Wood	78	34	SS	66	Common	4~5 months	9~10	8	31.8
152	Earth	80	234	SS	57	R/C	~10 days	0	2	-5
153	Plastering	83	24	HA	34	Common	5~6 months	11~15	8	34.7
154	R/C	25	24	HA	30	Framing	1~2 months	3~4	4	19.2
155	Curtain Wall	56	23	CP	53	Earth	Over 1 years	21~	10	5.1
156	R/C	25	45	Apt	45	R/C	10~20 days	1	3	12.3
157	R/C	51	27	HS	37	R/C	10~20 days	1	3	13
158	R/C	36	51	SS	36	Finishing	2~3 months	5~6	7	34.3
159	R/C	9	42	HA	32	R/C	10~20 days	1	3	6.7
160	Roof	55	623	HS	39	Common	4~5 months	9~10	8	31
161	Plastering	65	1168	SS	36	Watertight	5~6 months	11~15	8	34.4
162	Etc.	56	1067	HA	52	Tile	6~12 months	16~20	10	8.9
163	R/C	44	37	CP	23	R/C	10~20 days	1	3	10.9
164	E/V	65	46	CP	31	Concrete finish	Over 1 years	21~	10	6.9
165	R/C	55	39	CP	26	Framing	1~2 months	3~4	4	11.2

166	Isolation	71	9	CP	63	Plumber	6~12 months	16~20	9	15.8
167	Dismantling	97	247	CP	65	Tile	6~12 months	16~20	9	11.4
168	R/C	57	47	Apt	64	Plaster	2~3 months	5~6	6	20.2
169	Steel Structure	63	49	Apt	48	Brick	3~4 months	7~8	7	30.6
170	Window	46	4	HS	38	Re-bar	5~6 months	11~15	8	27.8
171	Curtain Wall	99	42	HS	21	Earth	Over 1 years	21~	10	8.4
172	Steel Structure	55	45	SS	39	Sheet Metal	3~4 months	7~8	7	34.3
173	Earth	50	31	Apt	32	R/C	~10 days	0	2	-2.4
174	Isolation	42	129	HA	29	Plumber	6~12 months	16~20	9	13.2
175	Steel Structure	99	47	HS	33	Brick	3~4 months	7~8	7	27.4
176	Etc.	58	1055	Apt	69	Tile	6~12 months	16~20	10	9.4
177	R/C	50	121	HA	47	Framing	1~2 months	3~4	4	10.6
178	R/C	77	44	SS	44	Electronic	2~3 months	5~6	6	23.9
179	R/C	43	5	SS	46	Paint	1~2 months	3~4	5	20.1
180	R/C	92	1121	Apt	53	Finishing	3~4 months	7~8	7	28.7
181	R/C	73	548	Apt	51	Roof	20~30 days	2	3	9.3
182	Wall	71	48	SS	40	Common	4~5 months	9~10	8	32.5
183	Earth	86	1148	HS	49	R/C	~10 days	0	2	0.9
184	Etc.	75	33	CP	20	Tile	6~12 months	16~20	10	0.3
185	Foundation	76	48	CP	55	R/C	~10 days	0	2	-1.3
186	Dismantling	57	45	Apt	34	Plumber	6~12 months	16~20	9	15.6
187	R/C	83	254	Apt	26	R/C	10~20 days	1	3	13.6
188	Etc.	30	25	CP	42	Concrete finish	6~12 months	16~20	10	1.8
189	R/C	13	533	SS	37	R/C	10~20 days	1	3	8.9
190	R/C	32	38	SS	20	Roof	20~30 days	2	3	5.4
191	Temporary	96	33	HA	29	Scaffold	~10 days	0	1	-6.6

192	R/C	67	120	SS	20	R/C	10~20 days	1	3	11.4
193	R/C	72	25	SS	40	Plaster	1~2 months	3~4	5	19.1
194	R/C	65	1092	Apt	33	R/C	10~20 days	1	3	5.6
195	R/C	74	250	Apt	38	Framing	1~2 months	3~4	5	22.9
196	Steel Structure	81	44	Apt	48	Brick	3~4 months	7~8	7	28.5
197	Temporary	71	2	HS	33	Scaffold	~10 days	0	1	-7.9
198	Stone	89	45	CP	51	Common	4~5 months	9~10	7	31.8
199	Temporary	62	33	SS	22	Scaffold	~10 days	0	1	-6
200	E/V	70	1014	CP	69	Earth	Over 1 years	21~	10	5.5
201	R/C	56	8	Apt	67	Framing	20~30 days	2	4	12.7
202	R/C	74	200	Apt	45	Framing	1~2 months	3~4	4	15.6
203	R/C	76	31	HA	50	Framing	1~2 months	3~4	5	18.9
204	R/C	41	2	HS	22	R/C	10~20 days	1	3	7.5
205	Temporary	77	629	SS	40	Scaffold	~10 days	0	1	-0.5
206	R/C	77	47	HS	59	Electronic	2~3 months	5~6	6	23.7
207	Tile	79	45	SS	52	Common	4~5 months	9~10	8	34.4
208	Plastering	81	1109	Apt	53	Watertight	5~6 months	11~15	8	28.5
209	Plastering	19	1015	Apt	41	Watertight	5~6 months	11~15	8	34.1
210	E/V	62	46	Apt	32	Concrete finish	Over 1 years	21~	10	4.4
211	Metal	90	2	Apt	22	Common	4~5 months	9~10	8	35.9
212	Stone	81	34	CP	66	Common	4~5 months	9~10	7	36.4
213	E/V	73	9	CP	45	Earth	Over 1 years	21~	10	5.7
214	R/C	50	1007	HS	61	Roof	20~30 days	2	3	11.6
215	Plastering	64	55	CP	28	Watertight	5~6 months	11~15	8	28.8
216	R/C	93	6	SS	38	R/C	10~20 days	1	3	6.7
217	Tile	94	588	Apt	62	Common	4~5 months	9~10	8	36.1

218	Metal	33	2	HS	69	Common	4~5 months	9~10	8	33.1
219	Plastering	58	51	Apt	65	Common	5~6 months	11~15	8	32.8
220	R/C	75	16	SS	46	Paint	1~2 months	3~4	5	20.8
221	Foundation	75	4	Apt	31	R/C	~10 days	0	2	-0.4
222	Dismantling	43	2	HA	34	Plumber	6~12 months	16~20	9	12.6
223	Temporary	41	19	Apt	54	Scaffold	~10 days	0	1	-7.3
224	Window	26	522	Apt	46	Re-bar	5~6 months	11~15	8	31.1
225	R/C	72	116	CP	61	R/C	10~20 days	1	3	5.2
226	Window	35	1106	Apt	34	Re-bar	5~6 months	11~15	8	36.8
227	Plastering	93	181	Apt	67	Common	5~6 months	11~15	8	35.7
228	Plastering	1	8	CP	53	Watertight	5~6 months	11~15	8	28.9
229	R/C	53	1022	CP	25	Roof	20~30 days	2	3	11.1
230	E/V	21	633	HS	24	Concrete finish	Over 1 years	21~	10	5.8
231	R/C	98	17	HA	45	Plaster	1~2 months	3~4	5	17.7
232	Window	93	39	HS	41	Plumber	6~12 months	16~20	8	32.8
233	R/C	73	43	HS	30	Framing	20~30 days	2	4	19
234	R/C	36	1088	Apt	64	R/C	20~30 days	2	3	14.4
235	Tile	62	20	HS	58	Common	4~5 months	9~10	8	28.6
236	R/C	58	1143	HS	40	Framing	1~2 months	3~4	4	12.8
237	Paint	61	22	HA	20	Plumber	6~12 months	16~20	8	30.1
238	R/C	41	4	HS	45	Electronic	2~3 months	5~6	6	21.8
239	Stone	40	7	Apt	37	Common	4~5 months	9~10	7	32.1
240	R/C	29	180	HA	45	Finishing	3~4 months	7~8	7	33.9
241	R/C	52	51	CP	21	Framing	20~30 days	2	4	13.5
242	Wall	46	18	CP	29	Common	4~5 months	9~10	8	30.4
243	Temporary	58	1082	Apt	52	Scaffold	~10 days	0	1	-7.2

244	E/V	65	9	HA	48	Concrete finish	Over 1 years	21~	10	0.2
245	R/C	74	186	HS	32	R/C	10~20 days	1	3	5.8
246	Steel Structure	83	42	CP	27	Brick	3~4 months	7~8	7	27.2
247	Foundation	10	41	Apt	66	R/C	~10 days	0	2	-7.2
248	Window	98	1168	HS	64	Re-bar	5~6 months	11~15	8	32.1
249	Plastering	78	639	Apt	36	Common	5~6 months	11~15	8	32.9
250	Window	36	24	Apt	54	Re-bar	5~6 months	11~15	8	27.8
251	E/V	57	1063	Apt	51	Earth	Over 1 years	21~	10	9.5
252	Steel Structure	88	22	CP	69	Common	3~4 months	7~8	7	29.5
253	Window	44	49	CP	31	Re-bar	5~6 months	11~15	8	30.2
254	R/C	61	132	Apt	39	R/C	10~20 days	1	3	10.4
255	Tile	61	515	SS	34	Common	4~5 months	9~10	8	32.7
256	R/C	54	43	HA	32	Plaster	1~2 months	3~4	5	21.1
257	R/C	44	39	CP	56	Finishing	3~4 months	7~8	7	29.1
258	Etc.	66	21	HA	43	Concrete finish	6~12 months	16~20	10	6.6
259	Window	10	39	SS	36	Plumber	6~12 months	16~20	8	32.6
260	Window	38	150	SS	47	Plumber	6~12 months	16~20	8	32.5
261	Temporary	67	6	CP	56	Scaffold	~10 days	0	1	-2.9
262	Window	26	123	SS	44	Re-bar	5~6 months	11~15	8	34.8
263	Temporary	49	106	CP	49	Scaffold	~10 days	0	1	-3.7
264	Window	47	1099	Apt	66	Re-bar	5~6 months	11~15	8	30.7
265	R/C	36	103	CP	63	Electronic	2~3 months	5~6	6	22.8
266	R/C	38	34	HA	21	Brick	3~4 months	7~8	7	36.7
267	R/C	85	33	HS	20	Roof	20~30 days	2	3	9.4
268	R/C	100	643	Apt	52	Plaster	2~3 months	5~6	6	26.8
269	R/C	63	1130	CP	36	Plaster	2~3 months	5~6	5	23.2

270	R/C	43	135	HS	45	Framing	20~30 days	2	4	11.5
271	R/C	24	653	Apt	37	R/C	20~30 days	2	3	13.1
272	E/V	22	198	HS	59	Concrete finish	Over 1 years	21~	10	1.5
273	R/C	6	676	HA	21	Framing	1~2 months	3~4	4	14.2
274	Temporary	25	213	HA	49	Scaffold	~10 days	0	1	0
275	Etc.	49	185	Apt	33	Tile	6~12 months	16~20	10	3.4
276	Earth	40	1062	Apt	66	R/C	~10 days	0	2	-0.2
277	Window	49	608	CP	59	Plumber	6~12 months	16~20	8	35.9
278	Roof	61	165	Apt	35	Common	4~5 months	9~10	8	35.9
279	R/C	41	6	Apt	64	Framing	1~2 months	3~4	4	15.3
280	R/C	81	156	SS	50	Framing	1~2 months	3~4	4	13.8
281	R/C	33	227	SS	58	Framing	1~2 months	3~4	4	14
282	E/V	34	1115	SS	68	Concrete finish	Over 1 years	21~	10	9.2
283	R/C	84	174	Apt	27	Framing	1~2 months	3~4	4	12.9
284	Foundation	56	191	HA	51	R/C	~10 days	0	2	1.6
285	R/C	79	167	Apt	37	Framing	1~2 months	3~4	4	16.2
286	Earth	33	246	SS	64	R/C	~10 days	0	2	-6.5
287	R/C	81	43	Apt	30	R/C	10~20 days	1	3	13.4
288	R/C	27	6	HA	47	Electronic	2~3 months	5~6	6	21.7
289	R/C	69	222	Apt	31	Plaster	1~2 months	3~4	5	19.4
290	R/C	56	1196	HA	42	R/C	20~30 days	2	3	8.1
291	R/C	6	142	HA	35	Electronic	2~3 months	5~6	6	25
292	R/C	98	181	HA	43	Framing	1~2 months	3~4	4	14.2
293	R/C	84	1180	CP	67	Roof	20~30 days	2	3	11.6
294	R/C	94	1157	Apt	34	Finishing	2~3 months	5~6	7	31.9
295	Tile	100	52	SS	68	Common	4~5 months	9~10	8	32.1

296	E/V	10	163	SS	23	Concrete finish	Over 1 years	21~	10	0
297	R/C	61	564	Apt	45	R/C	10~20 days	1	3	10.7
298	Curtain Wall	50	49	Apt	40	Earth	Over 1 years	21~	10	1.3
299	Window	100	121	Apt	27	Re-bar	5~6 months	11~15	8	27.7
300	Plastering	70	31	CP	48	Common	5~6 months	11~15	8	36.3
301	Steel Structure	66	50	CP	30	Sheet Metal	3~4 months	7~8	7	28.1
302	Metal	40	1089	CP	20	Common	4~5 months	9~10	8	35
303	Foundation	31	123	HS	26	R/C	10~20 days	1	2	0.4
304	R/C	93	1099	CP	64	Plaster	1~2 months	3~4	5	17.4
305	Temporary	59	44	Apt	31	Scaffold	~10 days	0	1	-2
306	Temporary	83	654	HA	43	Scaffold	~10 days	0	1	-9
307	Plastering	82	18	HS	56	Common	5~6 months	11~15	8	29
308	Temporary	50	1070	Apt	47	Scaffold	~10 days	0	1	-3.7
309	R/C	48	55	Apt	69	Carpenter	2~3 months	5~6	6	29.4
310	Stone	42	1011	CP	52	Common	4~5 months	9~10	7	31.4
311	Stone	12	9	Apt	24	Common	4~5 months	9~10	7	33.3
312	Temporary	99	49	CP	64	R/C	~10 days	0	1	-5
313	R/C	27	523	CP	32	Framing	20~30 days	2	4	13
314	R/C	71	528	SS	35	Roof	20~30 days	2	3	9.4
315	R/C	77	690	Apt	55	Roof	20~30 days	2	3	6
316	R/C	37	554	Apt	53	Framing	1~2 months	3~4	4	18.3
317	Curtain Wall	4	1068	HA	23	Earth	Over 1 years	21~	10	6
318	R/C	88	677	HA	35	Roof	20~30 days	2	3	13.2
319	R/C	12	248	SS	41	Paint	1~2 months	3~4	5	22.6
320	Window	32	225	HS	45	Plumber	6~12 months	16~20	8	35.9
321	R/C	72	52	HS	68	Roof	20~30 days	2	3	6.6

322	Etc.	25	514	Apt	60	Concrete finish	6~12 months	16~20	10	8.8
323	Plastering	43	561	Apt	51	Common	5~6 months	11~15	8	32.9
324	Finish	24	522	Apt	47	Plumber	6~12 months	16~20	9	20
325	R/C	91	672	Apt	38	Plaster	2~3 months	5~6	6	23.9
326	Window	38	250	HS	23	Plumber	6~12 months	16~20	8	29.3
327	R/C	82	15	CP	45	Framing	1~2 months	3~4	4	11.9
328	R/C	25	15	SS	49	Framing	20~30 days	2	4	15.3
329	Paint	6	34	SS	67	Plumber	6~12 months	16~20	8	27.1
330	Roof	76	7	CP	66	Common	4~5 months	9~10	8	33
331	Finish	64	630	Apt	52	Plumber	6~12 months	16~20	9	10.5
332	Earth	2	49	HS	46	R/C	~10 days	0	2	-6.8
333	Temporary	62	672	HS	60	Scaffold	~10 days	0	1	-6.9
334	Wall	88	695	HS	39	Common	4~5 months	9~10	8	27.5
335	R/C	11	33	HS	65	Framing	20~30 days	2	4	20
336	Plastering	66	124	SS	45	Watertight	5~6 months	11~15	8	29.6
337	Stone	40	699	HA	55	Common	4~5 months	9~10	7	37
338	R/C	61	47	HS	34	Framing	20~30 days	2	4	14.6
339	R/C	15	196	HS	28	Plaster	2~3 months	5~6	5	21.2
340	Wood	23	1042	Apt	43	Common	4~5 months	9~10	8	31.2
341	Earth	58	1167	HA	60	R/C	~10 days	0	2	1.7
342	Window	66	181	CP	56	Plumber	6~12 months	16~20	8	32.4
343	Window	26	1179	SS	23	Re-bar	5~6 months	11~15	8	31.5
344	Stone	6	168	SS	65	Common	4~5 months	9~10	7	33.9
345	R/C	48	42	CP	61	R/C	20~30 days	2	3	12.4
346	R/C	11	1039	Apt	26	Roof	20~30 days	2	3	6.6
347	Wall	94	47	HS	66	Common	4~5 months	9~10	8	35.4

348	R/C	89	565	SS	56	Framing	1~2 months	3~4	5	18.6
349	Stone	65	1030	CP	47	Common	4~5 months	9~10	7	35.9
350	R/C	35	637	HS	40	R/C	10~20 days	1	3	8.9
351	R/C	7	129	HA	49	Framing	20~30 days	2	4	12
352	R/C	40	1064	HA	58	Framing	1~2 months	3~4	4	19.7
353	Curtain Wall	19	209	HS	25	Earth	Over 1 years	21~	10	9.6
354	Plastering	86	42	SS	30	Watertight	5~6 months	11~15	8	37
355	R/C	79	614	HA	21	Brick	3~4 months	7~8	7	34.6
356	Etc.	6	220	CP	47	Tile	6~12 months	16~20	10	6.3
357	Steel Structure	63	218	HA	60	Sheet Metal	3~4 months	7~8	7	36.8
358	R/C	30	1084	HS	69	Framing	1~2 months	3~4	4	19.4
359	Window	39	208	SS	67	Re-bar	5~6 months	11~15	8	36.6
360	R/C	64	1017	Apt	33	Plaster	2~3 months	5~6	5	22.4
361	Plastering	27	240	HA	57	Common	5~6 months	11~15	8	30.6
362	Earth	5	138	HS	21	R/C	~10 days	0	2	-1.7
363	R/C	36	1084	SS	54	Framing	20~30 days	2	4	13.1
364	Steel Structure	12	1173	Apt	27	Sheet Metal	3~4 months	7~8	7	33.5
365	R/C	91	212	HS	30	Paint	1~2 months	3~4	5	20.8
366	R/C	91	37	CP	65	Roof	20~30 days	2	3	12.3
367	Steel Structure	59	194	SS	64	Brick	3~4 months	7~8	7	30.3
368	Earth	21	1108	Apt	41	R/C	~10 days	0	2	1.6
369	Steel Structure	90	182	HS	69	Common	3~4 months	7~8	7	29.3
370	R/C	91	9	CP	43	Electronic	2~3 months	5~6	6	23.2
371	Earth	19	181	HS	37	R/C	~10 days	0	2	-4.8
372	Earth	11	248	SS	30	R/C	~10 days	0	2	-3.1
373	Roof	64	8	SS	31	Common	4~5 months	9~10	8	32.7

374	R/C	91	176	CP	48	Plaster	2~3 months	5~6	5	17.2
375	R/C	88	167	CP	52	Carpenter	2~3 months	5~6	6	23.9
376	Window	88	1173	Apt	39	Re-bar	5~6 months	11~15	8	33.6
377	Etc.	80	14	HA	56	Tile	6~12 months	16~20	10	6.4
378	R/C	80	16	SS	63	Finishing	2~3 months	5~6	7	32.3
379	Temporary	27	14	HS	57	Scaffold	~10 days	0	1	-5
380	R/C	84	571	CP	65	Framing	20~30 days	2	4	18.2
381	E/V	10	160	SS	25	Concrete finish	Over 1 years	21~	10	1
382	R/C	83	1142	CP	67	Roof	20~30 days	2	3	8.9
383	R/C	79	43	Apt	24	Plaster	2~3 months	5~6	6	25.1
384	R/C	40	14	Apt	21	Framing	20~30 days	2	4	17.8
385	R/C	28	12	HS	63	Framing	20~30 days	2	4	17.9
386	Steel Structure	94	1064	Apt	39	Sheet Metal	3~4 months	7~8	7	27
387	Metal	42	525	HA	31	Common	4~5 months	9~10	8	36.4
388	Metal	74	189	SS	67	Common	4~5 months	9~10	8	30
389	R/C	71	1011	CP	23	R/C	10~20 days	1	3	5.3
390	R/C	31	19	SS	57	Paint	1~2 months	3~4	5	19.3
391	R/C	54	7	Apt	31	Finishing	3~4 months	7~8	7	32.5
392	R/C	62	15	SS	27	Plaster	2~3 months	5~6	6	21.5
393	Roof	3	40	CP	46	Common	4~5 months	9~10	8	35.4
394	Plastering	29	46	HA	30	Watertight	5~6 months	11~15	8	28.7
395	R/C	75	572	Apt	21	Framing	1~2 months	3~4	4	14.1
396	R/C	50	15	HA	49	Finishing	3~4 months	7~8	7	28.6
397	R/C	67	526	SS	37	Framing	1~2 months	3~4	4	14.5
398	R/C	58	231	HS	42	R/C	10~20 days	1	3	11.8
399	R/C	70	1066	HA	57	Plaster	1~2 months	3~4	5	18.6

400	R/C	29	222	CP	37	Finishing	3~4 months	7~8	7	27.1
401	Curtain Wall	15	159	HA	30	Earth	Over 1 years	21~	10	2
402	Isolation	35	12	HS	49	Plumber	6~12 months	16~20	9	11.1
403	Paint	63	17	HA	42	Plumber	6~12 months	16~20	8	28.3
404	Foundation	61	514	CP	37	R/C	~10 days	0	2	-7.9
405	Etc.	60	246	HA	62	Tile	6~12 months	16~20	10	5.6
406	Tile	49	38	Apt	53	Common	4~5 months	9~10	8	33.5
407	R/C	44	689	Apt	50	Framing	1~2 months	3~4	4	17.5
408	R/C	10	198	HS	39	Framing	1~2 months	3~4	4	13.6
409	R/C	56	214	HA	24	R/C	10~20 days	1	3	6.8
410	Curtain Wall	54	52	CP	43	Earth	Over 1 years	21~	10	5.1
411	Finish	22	643	SS	63	Plumber	6~12 months	16~20	9	16
412	R/C	17	668	HS	55	Framing	1~2 months	3~4	4	15.9
413	Temporary	2	183	HS	69	Scaffold	~10 days	0	1	-9.8
414	Curtain Wall	12	560	Apt	45	Earth	Over 1 years	21~	10	1.3
415	R/C	27	112	Apt	48	Plaster	2~3 months	5~6	6	24.4
416	R/C	34	11	SS	55	Electronic	2~3 months	5~6	6	24.1
417	R/C	71	19	SS	65	R/C	10~20 days	1	3	7.8
418	Steel Structure	78	604	Apt	40	Brick	3~4 months	7~8	7	31.4
419	Window	22	1117	Apt	24	Re-bar	5~6 months	11~15	8	35.9
420	Steel Structure	4	174	CP	59	Brick	3~4 months	7~8	7	30.1
421	Temporary	13	224	SS	28	Scaffold	~10 days	0	1	-9.9
422	Temporary	34	565	HA	67	Scaffold	~10 days	0	1	-3.5
423	Stone	37	50	HA	32	Common	4~5 months	9~10	7	31.8
424	R/C	40	190	SS	58	R/C	10~20 days	1	3	9.6
425	Steel Structure	90	50	Apt	56	Sheet Metal	3~4 months	7~8	7	28.4

426	R/C	53	220	HS	40	Finishing	3~4 months	7~8	7	33.5
427	R/C	14	213	HA	51	Finishing	3~4 months	7~8	7	28.1
428	Etc.	83	1106	Apt	54	Tile	6~12 months	16~20	10	7.2
429	R/C	90	9	HS	26	Carpenter	2~3 months	5~6	6	24.3
430	R/C	91	694	CP	63	R/C	10~20 days	1	3	11.2
431	Wall	42	1161	Apt	56	Common	4~5 months	9~10	8	27.6
432	Etc.	7	1135	Apt	38	Concrete finish	6~12 months	16~20	10	4.7
433	Window	45	219	HA	24	Re-bar	6~12 months	16~20	8	34.5
434	Steel Structure	21	615	Apt	55	Common	3~4 months	7~8	7	34.2
435	R/C	72	119	CP	68	R/C	10~20 days	1	3	13.9
436	R/C	86	185	HA	29	Plaster	2~3 months	5~6	6	29.1
437	Etc.	52	15	CP	25	Tile	6~12 months	16~20	10	7.7
438	R/C	28	28	SS	42	Framing	20~30 days	2	4	10.9
439	R/C	3	181	HS	40	Paint	1~2 months	3~4	5	16.7
440	Window	65	1103	HS	41	Re-bar	5~6 months	11~15	8	28.9
441	Window	70	623	HA	30	Re-bar	5~6 months	11~15	8	31.1
442	R/C	56	1166	HA	46	Framing	1~2 months	3~4	4	16.4
443	Wall	75	697	HS	59	Common	4~5 months	9~10	8	29.1
444	Dismantling	97	699	HA	41	Tile	6~12 months	16~20	9	14.3
445	Plastering	79	41	Apt	55	Common	5~6 months	11~15	8	30.8
446	Earth	43	49	HA	64	R/C	~10 days	0	2	-3
447	Wood	72	611	CP	54	Common	4~5 months	9~10	8	27.7
448	Finish	86	45	HS	62	Plumber	6~12 months	16~20	9	11.3
449	R/C	90	562	HA	28	Plaster	2~3 months	5~6	5	17.7
450	R/C	15	11	HA	49	Finishing	3~4 months	7~8	7	29.9
451	Foundation	34	55	HS	31	R/C	~10 days	0	2	-7.9

452	Steel Structure	7	211	Apt	30	Sheet Metal	3~4 months	7~8	7	30.1
453	R/C	68	148	SS	37	Plaster	2~3 months	5~6	6	21.9
454	Wood	92	1180	Apt	46	Common	4~5 months	9~10	8	36.9
455	Wall	69	43	HA	58	Common	4~5 months	9~10	8	29.7
456	Temporary	25	1162	HA	66	Scaffold	~10 days	0	1	-5.7
457	R/C	5	127	HA	41	Framing	20~30 days	2	4	15.9
458	Window	42	9	HA	34	Re-bar	6~12 months	16~20	8	32.2
459	Plastering	18	124	Apt	53	Watertight	5~6 months	11~15	8	29.2
460	Plastering	67	7	SS	59	Watertight	5~6 months	11~15	8	31.6
461	Window	2	206	SS	41	Re-bar	5~6 months	11~15	8	31.7
462	R/C	52	43	HS	20	R/C	20~30 days	2	3	5
463	R/C	87	146	HA	61	Roof	20~30 days	2	3	12.9
464	Window	39	39	HS	69	Re-bar	6~12 months	16~20	8	31.3
465	R/C	75	680	HA	58	Finishing	3~4 months	7~8	7	29.1
466	R/C	68	643	SS	21	Framing	1~2 months	3~4	5	21.2
467	Earth	24	40	Apt	66	R/C	~10 days	0	2	-1.1
468	Stone	9	118	HS	30	Common	4~5 months	9~10	7	31.2
469	R/C	20	221	HS	28	Roof	20~30 days	2	3	5.4
470	R/C	64	8	HA	60	Plaster	1~2 months	3~4	5	19.1
471	Metal	14	562	SS	37	Common	4~5 months	9~10	8	34.2
472	Earth	88	575	HA	48	R/C	~10 days	0	2	-0.9
473	Dismantling	26	156	SS	29	Plumber	6~12 months	16~20	9	12.1
474	Wall	76	35	SS	57	Common	4~5 months	9~10	8	36.7
475	Temporary	83	250	HS	52	Scaffold	~10 days	0	1	-6.7
476	Temporary	17	41	HS	36	Scaffold	~10 days	0	1	-1.2
477	R/C	23	147	HS	63	Carpenter	2~3 months	5~6	6	21.4

478	Steel Structure	63	1084	CP	28	Common	3~4 months	7~8	7	36.7
479	Finish	75	9	Apt	61	Plumber	6~12 months	16~20	9	18.7
480	Plastering	57	1053	SS	67	Common	5~6 months	11~15	8	27.1
481	R/C	61	632	HS	20	R/C	10~20 days	1	3	7.4
482	Etc.	55	1160	HS	42	Tile	6~12 months	16~20	10	5.7
483	R/C	61	110	HS	22	Finishing	3~4 months	7~8	7	27.3
484	Tile	49	1040	CP	30	Common	4~5 months	9~10	8	29.2
485	Foundation	19	210	SS	47	R/C	~10 days	0	2	-2.6
486	E/V	57	9	SS	39	Concrete finish	Over 1 years	21~	10	2.6
487	R/C	69	1098	CP	65	Roof	20~30 days	2	3	8
488	Tile	95	146	HS	40	Common	4~5 months	9~10	8	29.1
489	Etc.	57	9	CP	26	Tile	6~12 months	16~20	10	2.1
490	Steel Structure	64	127	HA	41	Sheet Metal	3~4 months	7~8	7	34.7
491	Dismantling	42	36	SS	63	Plumber	6~12 months	16~20	9	12.8
492	R/C	64	4	CP	26	Electronic	2~3 months	5~6	6	21.4
493	R/C	89	4	Apt	66	Plaster	2~3 months	5~6	5	16.9
494	R/C	24	207	HS	37	R/C	10~20 days	1	3	9.6
495	Plastering	27	125	CP	39	Common	5~6 months	11~15	8	27.7
496	R/C	52	51	Apt	54	Brick	3~4 months	7~8	7	31.9
497	R/C	84	125	SS	45	R/C	10~20 days	1	3	14.1
498	R/C	72	9	HA	56	Plaster	2~3 months	5~6	5	21.4
499	Foundation	92	34	HA	33	R/C	~10 days	0	2	2
500	E/V	47	8	HS	29	Concrete finish	Over 1 years	21~	10	6
501	R/C	9	1065	Apt	43	Electronic	2~3 months	5~6	6	27.8
502	Steel Structure	99	38	CP	56	Brick	3~4 months	7~8	7	35.7
503	Window	97	131	HA	50	Re-bar	5~6 months	11~15	8	36.2

504	Metal	9	31	SS	63	Common	4~5 months	9~10	8	29.4
505	Temporary	10	54	SS	28	Scaffold	~10 days	0	1	-2.5
506	Brick	14	184	HA	36	Common	4~5 months	9~10	7	30.2
507	Curtain Wall	80	7	HS	36	Earth	Over 1 years	21~	10	6.8
508	Steel Structure	34	520	HS	62	Sheet Metal	3~4 months	7~8	7	32.6
509	Tile	12	49	SS	58	Common	4~5 months	9~10	8	36.2
510	R/C	39	219	CP	43	Framing	1~2 months	3~4	4	13.9
511	Window	87	1089	HA	48	Re-bar	5~6 months	11~15	8	36.9
512	R/C	91	4	CP	42	R/C	20~30 days	2	3	7.8
513	R/C	55	239	HA	30	Brick	3~4 months	7~8	7	34.1
514	Window	75	250	Apt	56	Re-bar	5~6 months	11~15	8	29
515	R/C	75	9	HS	39	R/C	10~20 days	1	3	14.6
516	Wood	19	2	SS	29	Common	4~5 months	9~10	8	31.1
517	R/C	61	133	HS	58	R/C	10~20 days	1	3	9.6
518	Temporary	90	7	HA	34	Scaffold	~10 days	0	1	-2.9
519	Window	20	1001	Apt	52	Plumber	6~12 months	16~20	8	28
520	R/C	19	634	CP	41	Plaster	2~3 months	5~6	5	18.4
521	E/V	24	36	SS	58	Concrete finish	Over 1 years	21~	10	2.1
522	Plastering	83	197	Apt	46	Common	5~6 months	11~15	8	36
523	Etc.	36	50	Apt	60	Tile	6~12 months	16~20	10	2.4
524	Temporary	86	1006	SS	68	Scaffold	~10 days	0	1	-0.7
525	R/C	20	142	HA	61	R/C	10~20 days	1	3	14.3
526	Metal	57	525	HS	28	Common	4~5 months	9~10	8	32.6
527	Temporary	61	2	Apt	41	Scaffold	~10 days	0	1	-9
528	Temporary	2	3	Apt	34	R/C	~10 days	0	1	-8
529	R/C	36	7	HS	43	R/C	10~20 days	1	3	12

530	Finish	55	32	SS	63	Plumber	6~12 months	16~20	9	13
531	R/C	65	46	CP	64	Electronic	2~3 months	5~6	6	24.9
532	Plastering	38	4	Apt	56	Common	5~6 months	11~15	8	35.1
533	R/C	46	38	SS	23	Framing	1~2 months	3~4	4	17.8
534	Roof	85	1007	Apt	54	Common	4~5 months	9~10	8	30.1
535	Steel Structure	100	125	SS	29	Brick	3~4 months	7~8	7	31.4
536	Curtain Wall	65	156	HA	45	Earth	Over 1 years	21~	10	10
537	R/C	74	531	SS	63	Plaster	2~3 months	5~6	5	23
538	Curtain Wall	36	141	SS	65	Earth	Over 1 years	21~	10	1.4
539	Dismantling	56	165	HS	56	Plumber	6~12 months	16~20	9	19.1
540	Metal	93	108	SS	61	Common	4~5 months	9~10	8	29
541	R/C	65	674	Apt	21	Finishing	2~3 months	5~6	6	27.3
542	R/C	2	36	HA	51	Framing	1~2 months	3~4	4	12.4
543	R/C	89	1187	HA	50	R/C	10~20 days	1	3	8
544	R/C	56	174	HS	54	Brick	3~4 months	7~8	7	36.1
545	R/C	65	247	SS	61	Framing	1~2 months	3~4	4	17.1
546	Steel Structure	79	503	HA	32	Brick	3~4 months	7~8	7	27.5
547	R/C	65	48	SS	21	R/C	10~20 days	1	3	11.5
548	Plastering	41	9	HS	31	Watertight	5~6 months	11~15	8	34.4
549	Window	47	540	Apt	56	Re-bar	6~12 months	16~20	8	36.2
550	Window	97	233	SS	24	Plumber	6~12 months	16~20	8	35.6
551	Earth	65	33	CP	44	R/C	~10 days	0	2	-7.2
552	Roof	97	8	SS	35	Common	4~5 months	9~10	8	36.7
553	Foundation	86	1191	CP	61	R/C	10~20 days	1	2	-3.7
554	Etc.	56	189	HA	36	Tile	6~12 months	16~20	10	8.2
555	Finish	2	518	SS	24	Plumber	6~12 months	16~20	9	18.9

556	R/C	75	829	HA	45	Finishing	2~3 months	5~6	7	29
557	Steel Structure	78	687	Apt	53	Brick	3~4 months	7~8	7	32.5
558	R/C	75	19	CP	23	Roof	20~30 days	2	3	7.9
559	Curtain Wall	72	604	Apt	33	Earth	Over 1 years	21~	10	1.2
560	E/V	19	1081	HA	42	Earth	Over 1 years	21~	10	9.4
561	Foundation	74	660	HS	48	R/C	10~20 days	1	2	-2
562	R/C	75	892	SS	46	Roof	20~30 days	2	3	10.7
563	R/C	76	1162	Apt	47	Carpenter	2~3 months	5~6	6	24
564	R/C	62	34	CP	61	R/C	10~20 days	1	3	12.9
565	Curtain Wall	67	1095	SS	24	Earth	Over 1 years	21~	10	8.9
566	R/C	58	49	CP	58	Electronic	2~3 months	5~6	6	27.7
567	Window	52	1	Apt	35	Re-bar	5~6 months	11~15	8	30.7
568	Window	53	49	HA	59	Re-bar	5~6 months	11~15	8	27
569	Plastering	62	547	HA	54	Watertight	5~6 months	11~15	8	36.5
570	Plastering	65	1	HS	47	Watertight	5~6 months	11~15	8	28.5
571	Plastering	32	2	Apt	44	Watertight	5~6 months	11~15	8	34
572	R/C	32	1093	Apt	48	R/C	10~20 days	1	3	11.2
573	R/C	78	9	HS	31	Electronic	2~3 months	5~6	6	29.1
574	R/C	55	9	Apt	69	Electronic	2~3 months	5~6	6	27.5
575	R/C	56	8	SS	68	R/C	20~30 days	2	3	13.5
576	R/C	79	34	Apt	21	Carpenter	2~3 months	5~6	6	24.3
577	R/C	19	46	HA	40	Framing	1~2 months	3~4	4	18.8
578	R/C	78	54	CP	21	Roof	20~30 days	2	3	7.8
579	R/C	72	579	HS	49	Roof	20~30 days	2	3	8.3
580	R/C	63	41	Apt	36	Framing	20~30 days	2	4	18
581	R/C	10	8	HS	57	Electronic	2~3 months	5~6	6	20.5

582	Finish	9	13	CP	57	Plumber	6~12 months	16~20	9	15
583	Isolation	79	25	HS	61	Plumber	6~12 months	16~20	9	14.6
584	R/C	79	49	SS	54	Framing	20~30 days	2	4	13.6
585	R/C	48	1074	HA	35	Plaster	1~2 months	3~4	5	19.4
586	Window	40	1189	SS	38	Re-bar	6~12 months	16~20	8	30
587	Dismantling	79	595	Apt	29	Plumber	6~12 months	16~20	9	10.7
588	R/C	63	1169	CP	58	Roof	20~30 days	2	3	13
589	R/C	68	3200	Apt	57	R/C	10~20 days	1	3	6.9
590	Plastering	40	1032	CP	62	Watertight	5~6 months	11~15	8	34.4
591	R/C	65	4	Apt	63	Electronic	2~3 months	5~6	6	26
592	Metal	29	111	CP	40	Common	4~5 months	9~10	8	27.8
593	Temporary	74	226	HS	48	Scaffold	~10 days	0	1	-5.2
594	R/C	63	213	CP	35	Framing	1~2 months	3~4	4	12.1
595	Steel Structure	68	34	CP	53	Sheet Metal	3~4 months	7~8	7	29.9
596	R/C	76	1144	HS	24	R/C	20~30 days	2	3	11.3

Appendix C: The Source Code of Non-activity Oriented Hazard Identification System

[System Main Form]

```
using System;
using System.Data;
using System.Windows.Forms;
using System.Threading;
using System.Net.Sockets;
using System.Net;
using System.Collections.Generic;
using System.Collections;
using System.Runtime.InteropServices;
using System.Text;
using System.IO;
using HAZ_IDF.BIZ;

namespace HAZ_IDF
{
        public partial class MainForm : Form
        {
                private   static   ManualResetEvent   ALLDONE   =   new
ManualResetEvent(false);
                private Utils UTILS = new Utils();
                private Thread threadServer = null;
                private Socket sServer = null;
                private string USER_ID = string.Empty;
                private string PJT_ID = string.Empty;
                public string SERVER_IP = string.Empty;
                public int SERVER_PORT = 0;
private byte[] _byte_Data;
       FileStream fs;
       BufferedStream bout;
       private int BUFFER = 32768;

       public string m_AlramNum = "2";      // 알람 횟수
       public string m_AlramInterval = "1";  // 알람 시간

        public MainForm()
                {
                InitializeComponent();

                ShowLoginForm(); // Login 화면 띄우기

                SetPJT_ID();     // 프로젝트 코드 가져오기
                SetServerIP();   // DB에 등록된 서버 아이피, 포트 세팅
                InitSystemConfig();      // 시스템 초기화
                this.WindowState = FormWindowState.Maximized;
                }

        #region ===== 로그인 관련 =====
                /// <summary>
                /// 로그인 화면 띄우기
```

```
                /// </summary>
        #region private void ShowLoginForm()
                private void ShowLoginForm()
                {
                        LoginForm frmUI = new LoginForm();
                        frmUI.Owner = this;
                        frmUI.ShowDialog();
                }
                #endregion
                #region public string UserID
                public string UserID
                {
                        get
                        {
                                return USER_ID;
                        }
                        set
                        {
                                USER_ID = value;
                        }
                }
                #endregion
                #endregion
        #region ===== 시스템 초기화 =====
                #region public string PjtID
                public string PjtID
                {
                        get
                        {
                                return PJT_ID;
                        }
                        set
                        {
                                PJT_ID = value;
                        }
                }
                #endregion
                /// <summary>
                /// 프로젝트 코드 가져오기
                /// </summary>
                #region public void SetPJT_ID()
                public void SetPJT_ID()
                {
                PJT_INFO_Biz biz = new PJT_INFO_Biz();
                DataSet ds = biz.SELECT_T_PJT_INFO(PJT_ID);
                if (ds.Tables[0].Rows.Count == 0)
                {
                SetMenuDisabled();
                stripMainPJTName.Text = "등록된 프로젝트 없음";
                        }
                        else
                        {
                        PJT_ID                                          =
ds.Tables[0].Rows[0]["PJT_ID"].ToString();
                        stripMainPJTName.Text                           =
ds.Tables[0].Rows[0]["PJT_NAME"].ToString();
                                SetMenuEnabled();
                        }
                }
```

```
#endregion
/// <summary>
/// 프로젝트 메뉴 활성화
/// </summary>
#region public void SetMenuEnabled()
public void SetMenuEnabled()
{
        toolStripServerStart.Enabled = true;
        toolStripServerStop.Enabled = true;
        toolStripPeopleInfo.Enabled = true;
        toolStripDeadWound.Enabled = true;
        toolStripDangerDegress.Enabled = true;
        toolStripConstScale.Enabled = true;
        toolStripReader.Enabled = true;
        toolStripWorkItem.Enabled = true;
        toolStripOption.Enabled = true;
        toolStripProjectDel.Enabled = true;
        toolStripMornitoring.Enabled = true;
}
#endregion

/// <summary>
/// 프로젝트 메뉴 비활성화
/// </summary>
#region public void SetMenuDisabled()
public void SetMenuDisabled()
{
        toolStripServerStart.Enabled = false;
        toolStripServerStop.Enabled = false;
        toolStripPeopleInfo.Enabled = false;
        toolStripDeadWound.Enabled = false;
        toolStripDangerDegress.Enabled = false;
        toolStripConstScale.Enabled = false;
        toolStripReader.Enabled = false;
        toolStripWorkItem.Enabled = false;
        toolStripOption.Enabled = false;
        toolStripProjectDel.Enabled = false;
        toolStripMornitoring.Enabled = false;
}
#endregion

/// <summary>
/// DB에 등록된 서버 아이피, 포트 세팅
/// </summary>
#region private void SetServerIP()
private void SetServerIP()
{
        PJT_INFO_Biz biz = new PJT_INFO_Biz();
        DataSet ds = biz.SELECT_T_CONFIG(PJT_ID);

        if (ds.Tables[0].Rows.Count == 0)
        {
        MessageBox.Show("환경설정에서 서버아이피를 입력해
주십시오.", "위험지역 검색 시스템", MessageBoxButtons.OK,
MessageBoxIcon.Information);
                return;
        }
        else
        {
```

```
                                SERVER_IP                                =
ds.Tables[0].Rows[0]["CFG_SERIP"].ToString();
                                SERVER_PORT                              =
Convert.ToInt32(ds.Tables[0].Rows[0]["CFG_PORT"]);
                        }
                }
                #endregion

                /// <summary>
                /// 시스템 초기화
                /// </summary>
                #region private void InitSystemConfig()
                private void InitSystemConfig()
                {
                        try
                        {
                                // 프로그램에서 사용할 디렉토리 생성

        UTILS.CheckPathInfo(Application.StartupPath);

                                UTILS.WriteLogFile("*****  프로그램초기화
시작 ***** ", "A");
                        }
                        catch (Exception ex)
                        {
                                UTILS.WriteLogFile("*****
프로그램초기화중 에러 발생 msg : " + ex.Message, "E");
                        }
                        UTILS.WriteLogFile("*****   프로그램초기화   종료
*****", "A");
                }
                #endregion
                #endregion
                #region ===== 메뉴 클릭 =====
                #region ----- 파일 -----

                /// <summary>
                /// 서버 시작 클릭
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region private void toolStripServerStart_Click(object
sender, EventArgs e)
                private void toolStripServerStart_Click(object sender,
EventArgs e)
                {
                        if (threadServer != null)
                        {
                                StopServer();
                        }
                        StartServerThread();
                }
                #endregion

                /// <summary>
                /// 서버 정지 클릭
                /// </summary>
                /// <param name="sender"></param>
```

```
                /// <param name="e"></param>
                #region private void toolStripServerStop_Click(object
sender, EventArgs e)
                private void toolStripServerStop_Click(object sender,
EventArgs e)
                {
                        StopServer();
                }
                #endregion

                /// <summary>
                /// 끝내기
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region private void toolStripExit_Click(object sender,
EventArgs e)
                private void toolStripExit_Click(object sender,
EventArgs e)
                {
                        if (MessageBox.Show("위험지역 검색관리 시스템을
종료하시겠습니까?", "위험지역 검색 관리", MessageBoxButtons.YesNo,
MessageBoxIcon.Information) != DialogResult.Yes)
                                return;
                        Application.Exit();
                }
                #endregion
                /// <summary>
                /// 모니터링
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region private void toolStripMornitoring_Click(object
sender, EventArgs e)
                private void toolStripMornitoring_Click(object sender,
EventArgs e)
                {
                        if
(FormFocusing(typeof(HAZ_IDF_CS.PJT_INFO.PJT_INFO_MST))) return;

                        HAZ_IDF_CS.PEOPLE.PEO_MNTR_VW frmUI = new
HAZ_IDF.HAZ_IDF_CS.PEOPLE.PEO_MNTR_VW();
                        frmUI.MdiParent = this;
                        frmUI.Show();
                }
                #endregion
                #endregion
                #region ----- 프로젝트 -----

                /// <summary>
                /// 프로젝트 관리 클릭
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region private void toolStripProjectInfo_Click(object
sender, EventArgs e)
                private void toolStripProjectInfo_Click(object sender,
EventArgs e)
                {
```

```
                        if
(FormFocusing(typeof(HAZ_IDF_CS.PJT_INFO.PJT_INFO_MST))) return;

                        HAZ_IDF_CS.PJT_INFO.PJT_INFO_MST  frmUI  =  new
HAZ_IDF.HAZ_IDF_CS.PJT_INFO.PJT_INFO_MST();
                        frmUI.MdiParent = this;
                        frmUI.Show();
                }
                #endregion

                /// <summary>
                /// 작업자 관리
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region  private  void  toolStripPeopleInfo_Click(object
sender, EventArgs e)
                private  void  toolStripPeopleInfo_Click(object  sender,
EventArgs e)
                {
                        if
(FormFocusing(typeof(HAZ_IDF.HAZ_IDF_CS.PEOPLE.PEO_INFO_MST))) return;

                        HAZ_IDF_CS.PEOPLE.PEO_INFO_MST   frmUI   =   new
HAZ_IDF.HAZ_IDF_CS.PEOPLE.PEO_INFO_MST();
                        frmUI.MdiParent = this;
                        frmUI.Show();
                }
                #endregion

                /// <summary>
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region  private  void  toolStripDeadWound_Click(object
sender, EventArgs e)
                private  void  toolStripDeadWound_Click(object  sender,
EventArgs e)
                {
                        if
(FormFocusing(typeof(HAZ_IDF_CS.PEOPLE.DISASTER_MST))) return;

                        HAZ_IDF_CS.PEOPLE.DISASTER_MST   frmUI   =   new
HAZ_IDF.HAZ_IDF_CS.PEOPLE.DISASTER_MST();
                        frmUI.MdiParent = this;
                        frmUI.Show();
                }
                #endregion

                /// <summary>
                /// 위험원 관리
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region                  private                  void
toolStripDangerDegress_Click(object sender, EventArgs e)
               private void toolStripDangerDegress_Click(object sender,
EventArgs e)
                {
```

```
                        if
(FormFocusing(typeof(HAZ_IDF_CS.PJT_INFO.DANGER_MST))) return;

                        HAZ_IDF_CS.PJT_INFO.DANGER_MST   frmUI   =   new
HAZ_IDF.HAZ_IDF_CS.PJT_INFO.DANGER_MST();
                        frmUI.MdiParent = this;
                        frmUI.Show();
                }
                #endregion

                /// <summary>
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region  private  void  toolStripConstScale_Click(object
sender, EventArgs e)
                private  void  toolStripConstScale_Click(object  sender,
EventArgs e)
                {
                        if
(FormFocusing(typeof(HAZ_IDF.HAZ_IDF_CS.PJT_INFO.SCALE_MST))) return;

                        HAZ_IDF_CS.PJT_INFO.SCALE_MST   frmUI   =   new
HAZ_IDF.HAZ_IDF_CS.PJT_INFO.SCALE_MST();
                        frmUI.MdiParent = this;
                        frmUI.Show();
                }
                #endregion

                /// <summary>
                /// 위치추적 장비 현황 관리
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region   private   void   toolStripReader_Click(object
sender, EventArgs e)
                private   void   toolStripReader_Click(object   sender,
EventArgs e)
                {
                        if
(FormFocusing(typeof(HAZ_IDF.HAZ_IDF_CS.PJT_INFO.READER_MST))) return;

                        HAZ_IDF_CS.PJT_INFO.READER_MST   frmUI   =   new
HAZ_IDF.HAZ_IDF_CS.PJT_INFO.READER_MST();
                        frmUI.MdiParent = this;
                        frmUI.Show();
                }
                #endregion

        #endregion

                #region ----- 공종 -----

                /// <summary>
                /// 위험상황 관리
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region   private   void   toolStripWorkItem_Click(object
```

```
sender, EventArgs e)
                private  void  toolStripWorkItem_Click(object  sender,
EventArgs e)
                {
                        if
(FormFocusing(typeof(HAZ_IDF.HAZ_IDF_CS.WOK_ITM.WOK_ITM_MST))) return;

                        HAZ_IDF_CS.WOK_ITM.WOK_ITM_MST  frmUI  =  new
HAZ_IDF.HAZ_IDF_CS.WOK_ITM.WOK_ITM_MST();
                        frmUI.MdiParent = this;
                        frmUI.Show();
                }
                #endregion

                #endregion

                #region ----- 설정 ------

                /// <summary>
                /// 환경 설정
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region  private  void  toolStripOption_Click(object
sender, EventArgs e)
                private  void  toolStripOption_Click(object  sender,
EventArgs e)
                {
                        if
(FormFocusing(typeof(HAZ_IDF.HAZ_IDF_CS.PJT_INFO.CONFIG_MST))) return;

                        HAZ_IDF_CS.PJT_INFO.CONFIG_MST  frmUI  =  new
HAZ_IDF.HAZ_IDF_CS.PJT_INFO.CONFIG_MST();
                        frmUI.Owner = this;
                        frmUI.ShowDialog();
                }
                #endregion

                /// <summary>
                /// 프로젝트 정보 삭제
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region  private  void  toolStripProjectDel_Click(object
sender, EventArgs e)
                private  void  toolStripProjectDel_Click(object  sender,
EventArgs e)
                {
                        if    (MessageBox.Show("프로젝트    정보를    모두
삭제하시겠습니까?",    "위험지역    검색    관리",    MessageBoxButtons.YesNo,
MessageBoxIcon.Information) != DialogResult.Yes) return;

                        PJT_INFO_Biz biz = new PJT_INFO_Biz();
                        int chkTrn = biz.DELETE_T_PJT_INFO(PJT_ID);

                        if (chkTrn == 0)
                        {
                                MessageBox.Show("삭제되었습니다.",
```

```
"위험지역 검색 관리", MessageBoxButtons.OK, MessageBoxIcon.Information);

                                SetMenuDisabled();       // 메뉴 비활성화

                                PJT_ID = "";
                        }
                        else
                        {
                                MessageBox.Show("삭제  실패.",  "위험지역
검색 관리", MessageBoxButtons.OK, MessageBoxIcon.Information);
                        }
                }
                #endregion

                #endregion
                #endregion
                #region ===== 통신 =====
                /// <summary>
                /// 서버를 시작한다.
                /// </summary>
                #region private void StartServerThread()
                private void StartServerThread()
                {
                        threadServer       =       new       Thread(new
ThreadStart(StartServer));
                        threadServer.IsBackground = true;
                        threadServer.Start();
                }
                #endregion

                /// <summary>
                /// 서버 동작을 정지시킨다.
                /// </summary>
                #region private void StopServer()
                private void StopServer()
                {
                        if (sServer != null)
                        {
            sServer.Shutdown(SocketShutdown.Both);
                                sServer.Close();
                                sServer = null;
                        }

                        if (threadServer != null)
                        {
                                threadServer.Abort();
                                threadServer = null;
                        }
                        stripMainStatus.Text = "서버 연결 안됨";
                }
                #endregion

        /// <summary>
                /// 소켓을 로컬 endpoint에 연결하고 들어오는 connection을
기다린다.
                /// </summary>
                #region private void StartServer()
                private void StartServer()
```

```
                {
          try
          {
             IPEndPoint  endPointLocal  =  new  IPEndPoint(IPAddress.Any,
SERVER_PORT);
             //IPAddress serverIPAddr = IPAddress.Parse(SERVER_IP);
             //IPEndPoint  endPointLocal  =  new  IPEndPoint(serverIPAddr,
SERVER_PORT);
             sServer     =     new     Socket(AddressFamily.InterNetwork,
SocketType.Dgram, ProtocolType.Udp);

             UTILS.WriteLogFile("***** 소켓 서버 시작 *****", "A");
             stripMainStatus.Text = "서버 수신 대기중...";

             sServer.Bind(endPointLocal);

             IPEndPoint          _IPEndPointRemote          =          new
IPEndPoint(IPAddress.Any, SERVER_PORT);
             EndPoint _EP_Remote = (EndPoint)_IPEndPointRemote;


             fs    =    new    FileStream("D:\\data\\AC4D_UP_ST_ID.txt",
FileMode.OpenOrCreate, FileAccess.Write
             bout = new BufferedStream(fs);
             //write = new BinaryWriter(fs);
                //SetString(this, "0");

              _byte_Data = new byte[BUFFER];
//
//              while (true)
//              {
                 ALLDONE.Reset();         // Nonsignaled 상태로 세팅한다.

                /* sServer.BeginAccept(new AsyncCallback(AcceptCallback),
sServer);*/
                 // 데이터받기 시작, 비동기적으로 데이터를 받기 시작한다.
                 sServer.BeginReceiveFrom(_byte_Data,                 0,
_byte_Data.Length, SocketFlags.None
                                          ,     ref     _EP_Remote,     new
AsyncCallback(AcceptCallback), _EP_Remote);
//
                 ALLDONE.WaitOne();       //  signal을  받을  때까지  현재
쓰레드를 차단한다.
//              }
          }
          catch (Exception ex)
          {
             UTILS.WriteLogFile("(StartServer)  msg  :  "  +  ex.Message,
"E");
          }
          finally
          {
             UTILS.WriteLogFile("***** 소켓 서버 종료 *****", "A");
          }
                 }
                 #endregion

//        #define getbit(x, y)   ((x)>>(y)) & (0x1)
```

```
//        #define setbit(x, y)   (x) |= ((0x1)<<(y))
//        #define clrbit(x, y)   (x) &= ~((0x1)<<(y))
//        unsigned char x = 0x80
//        getbit(x, 7);
//        setbit(x, 0);.
//        clrbit(x, 3);

       private int getbit(int data, int s)
       {
          return ((data)>>(s)) & (0x1);
       }
        private int setbit(int data, int s)
       {
          return (data) |= ((0x1)<<(s));
       }
       private int clrbit(int data, int s)
       {
          return (data) &= ~((0x1)<<(s));
       }
                /// <summary>
                /// 서버가 클라이언트로부터 요청을 받을 때 콜백 함수.
                /// </summary>
                /// <param name="ar"></param>
                #region public void AcceptCallback(IAsyncResult ar)
                public void AcceptCallback(IAsyncResult ar)
                {
                        //Socket handler = (Socket)ar;

                        try
                        {
             EndPoint _EP_Remote = new IPEndPoint(IPAddress.Any, 0);

             // 클라이언트로부터 메시지를 받는다.
             int  _int_ReceivedSize  =  sServer.EndReceiveFrom(ar,  ref
_EP_Remote);

             // 0보다 커야만 정상적으로 계속해서 받기 수행
             if (_int_ReceivedSize > 0)
             {
                string                 strReceiveMsg                 =
Encoding.UTF8.GetString(_byte_Data, 0, _int_ReceivedSize);

                #region 데이터 전송 형식
                //    ProjectID(2Byte),    TagID(3Byte),    X좌표(3Byte),
Y좌표(3Byte), 층정보(4Byte), 시각(7Byte), CheckSum(1Byte)
                #endregion


                short s_pjt_id = _byte_Data[1];
                short s_tag_id = _byte_Data[3];
                double n16_1 = 16.0 * 16.0;
                double n16_2 = n16_1 * n16_1;
                double n16_3 = n16_1 * n16_1 * n16_1;
                double n16_4 = n16_1 * n16_1 * n16_1 * n16_1;

                double   s_x   =   Convert.ToDouble(_byte_Data[9])   /
1000.000;
                s_x  +=  Convert.ToDouble(_byte_Data[8])  *  (n16_1  /
1000.000);
```

```
                s_x  +=  Convert.ToDouble(_byte_Data[7])  *  (n16_2  /
1000.000);
                s_x  +=  Convert.ToDouble(_byte_Data[6])  *  (n16_3  /
1000.000);
                //s_x = s_x / 1000;
                //s_x = s_x;   // mm => m 단위 변경
                double   s_y   =   Convert.ToDouble(_byte_Data[13])   /
1000.000;
                s_y  +=  Convert.ToDouble(_byte_Data[12])  *  (n16_1  /
1000.000);
                s_y  +=  Convert.ToDouble(_byte_Data[11])  *  (n16_2  /
1000.000);
                s_y  +=  Convert.ToDouble(_byte_Data[10])  *  (n16_3  /
1000.000);
                //s_y = s_y / 1000;   // mm => m 단위 변경
                //byte s_sign = _byte_Data[11];
                int s_floor = _byte_Data[14];

                int nbit = getbit(s_floor, 7);

                string strStory;

                if (nbit == 1)
                {
                   s_floor = clrbit(s_floor, 7);
                   strStory = "지하 " + s_floor.ToString() + "층";
                }
                else
                {
                   strStory = "지상 " + s_floor.ToString() + "층";
                }
                  // 2진수로 변경하여 맨앞 1bit는 지하, 지상 판단 비트
                // 0 = 지상
                // 1 = 지하

                int  s_year  =  _byte_Data[16]  +  (_byte_Data[15]  *
Convert.ToInt32(n16_1));
                byte s_month = _byte_Data[17];
                byte s_day = _byte_Data[18];
                byte s_hour = _byte_Data[19];
                byte s_minute = _byte_Data[20];
                byte s_second = _byte_Data[21];
               // byte s_sum = _byte_Data[22];

                DateTime dtTag = new DateTime(s_year, s_month, s_day,
s_hour, s_minute, s_second);

                int floor = s_floor;

//                 if (s_sign == 0)
//                    floor = s_floor * -1;

                string                  strRcvData                  =
string.Format("%d, %d, %d, %d, %d, %s", s_pjt_id, s_tag_id, s_x, s_y,
floor, dtTag.ToString());
                UTILS.WriteLogFile(_EP_Remote.ToString()  +  "로  부터
메세지 수신 : " + strRcvData, "C");
                stripMainStatus.Text = _EP_Remote.ToString() + " 로 부터
```

```
메세지 수신 종료";

                #region 수신데이터 DB 저장

                PEOPLE_Biz biz = new PEOPLE_Biz();
                int chkTrn = biz.INSERT_T_TAG_VALUE(s_pjt_id.ToString(),
s_tag_id.ToString(),   dtTag,   0,   0,   floor,   Convert.ToSingle(s_x),
Convert.ToSingle(s_y));

                if (chkTrn > 0)
                {
                   UTILS.WriteLogFile(DateTime.Now.ToString()   +   "TAG
정보 저장 실패.","S");
                }

                #endregion

             }
             // Receive
             sServer.BeginReceiveFrom(_byte_Data, 0, _byte_Data.Length,
SocketFlags.None, ref _EP_Remote,
                new AsyncCallback(AcceptCallback), _EP_Remote);
                        }
                        catch (Exception ex)
                        {
                                UTILS.WriteLogFile("(AcceptCallback)
msg :" + ex.Message, "E");
                        }
//                      finally
//                      {
//              if (sServer != null)
//                              {
//                 sServer.Shutdown(SocketShutdown.Both);
//                 sServer.Close();
//                 sServer = null;
//                              }
//                      }
                }
                #endregion

                /// <summary>
                /// 클라이언트로 부터 데이터 수신 콜백 함수
                /// </summary>
                /// <param name="ar"></param>
                #region public void ReadCallback(IAsyncResult ar)
                public void ReadCallback(IAsyncResult ar)
                {
                        StateObject state = (StateObject)ar.AsyncState;
                        Socket handler = state.workSocket;
          EndPoint _EP_Remote = new IPEndPoint(IPAddress.Any, 0);

                        try
                        {
                           // 클라이언트 소켓으로부터 데이터를 읽는다.
             int size = handler.EndReceiveFrom(ar, ref _EP_Remote);
                          // int size = handler.EndReceive(ar);

                           if (size > 0)
                                {
```

```
                                        //   클라이언트로부터   온   데이터
(state.buffer)를 (state.rcvBuffer)에 저장해 둔다.
                                        for (int i = 0; i < size; i++)
                                        {
                                                // byte[]에서 byte[]로
저장하려고 Marshal을 썼는데, 더 좋은 방법을 모르겠다 +_+

        Marshal.WriteByte(state.rcvBuffer,     state.rcvSize     +     i,
state.buffer[i]);
                                        }

                                        state.rcvSize += size;


                                        if (state.rcvSize == 23)
                                        {
                                                #region 데이터 전송 형식
                                                //   ProjectID(2Byte),
TagID(3Byte), X좌표(3Byte), Y좌표(3Byte), 층정보(4Byte)+지상 및 지하 포함하여
4Byte, 시각(7Byte), CheckSum(1Byte)
                                                #endregion

                                                short    s_pjt_id    =
BitConverter.ToInt16(state.rcvBuffer, 0);
                                                short    s_tag_id    =
BitConverter.ToInt16(state.rcvBuffer, 2);
                                                //float      s_x      =
BitConverter.ToSingle(state.rcvBuffer, 5);
                                                //float      s_y      =
BitConverter.ToSingle(state.rcvBuffer, 8);
                                                short       s_x       =
BitConverter.ToInt16(state.rcvBuffer, 5);
                                                short       s_y       =
BitConverter.ToInt16(state.rcvBuffer, 8);
                                                byte      s_sign      =
state.rcvBuffer[11];
                                                short     s_floor     =
BitConverter.ToInt16(state.rcvBuffer, 12);
                                                short     s_year      =
BitConverter.ToInt16(state.rcvBuffer, 15);
                                                byte     s_month      =
state.rcvBuffer[17];
                                                byte      s_day       =
state.rcvBuffer[18];
                                                byte      s_hour      =
state.rcvBuffer[19];
                                                byte     s_minute     =
state.rcvBuffer[20];
                                                byte     s_second     =
state.rcvBuffer[21];
                                                byte      s_sum       =
state.rcvBuffer[22];

                                                DateTime  dtTag  =  new
DateTime(s_year, s_month, s_day, s_hour, s_minute, s_second);

                                                int floor = s_floor;
```

```
                                                if (s_sign == 0)
                                                        floor        =
s_floor * -1;

                                                #region yoyoyo
                                                /*
                                                char[]    c_pjt_id    =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 0, 2);
                                                char[]    c_tag_id    =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 2, 3);
                                                char[]      c_x       =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 5, 3);
                                                char[]      c_y       =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 8, 3);
                                                char[]      c_ug      =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 11, 1);
                                                char[]    c_floor     =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 12, 3);
                                                char[]     c_year     =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 15, 2);
                                                char[]    c_month     =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 17, 1);
                                                char[]     c_day      =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 18, 1);
                                                char[]     c_time     =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 19, 1);
                                                char[]    c_minute    =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 20, 1);
                                                char[]    c_second    =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 21, 1);
                                                char[]     c_sum      =
System.Text.Encoding.Default.GetChars(state.rcvBuffer, 22, 1);

                                                string   str_pjt_id   =
ExtendedTrim(new string(c_pjt_id));
                                                string   str_tag_id   =
ExtendedTrim(new string(c_tag_id));
                                                string     str_x      =
ExtendedTrim(new string(c_x));
                                                string     str_y      =
ExtendedTrim(new string(c_y));
                                                string   str_floor    =
ExtendedTrim(new string(c_floor));
                                                string    str_date    =
ExtendedTrim(new string(c_floor));
                                                string    str_sum     =
ExtendedTrim(new string(c_sum));
                                                 * */
                                                #endregion

                                                string   strRcvData   =
string.Format("%d, %d, %d, %d, %d, %s, %d", s_pjt_id, s_tag_id, s_x, s_y,
floor, dtTag.ToString(), s_sum.ToString());

        UTILS.WriteLogFile(handler.RemoteEndPoint.ToString() + "로 부터
메세지 수신 : " + strRcvData, "C");
                                                stripMainStatus.Text =
handler.RemoteEndPoint.ToString() + " 로 부터 메세지 수신 종료";
```

```
                                                    #region   수신데이터   DB
저장

                                                    PEOPLE_Biz  biz  =  new
PEOPLE_Biz();
                                                    int      chkTrn      =
biz.INSERT_T_TAG_VALUE(s_pjt_id.ToString(),  s_tag_id.ToString(),  dtTag,
0, 0, floor, s_x, s_y);

                                                    if (chkTrn > 0)
                                                    {

MessageBox.Show("TAG     정보     저장     실패.",     "위험지역     검색관리",
MessageBoxButtons.OK, MessageBoxIcon.Information);
                                                    }

                                                    #endregion

                                                    #region          UDP로
클라이언트에 데이터 전송

                                                    #endregion
                                            }
                                else
                                {

                                 //handler.BeginReceive(state.buffer,  0,
state.buffer.Length, 0, new AsyncCallback(ReadCallback), state);

                     handler.BeginReceiveFrom(_byte_Data,              0,
_byte_Data.Length, SocketFlags.None, ref _EP_Remote,
                     new AsyncCallback(AcceptCallback), _EP_Remote);
                                }
                            }
                        }
                        catch (Exception ex)
                        {

UTILS.WriteLogFile(handler.RemoteEndPoint.ToString() + " (ReadCallback)
Client부터 데이타 수신중 에러 발생 msg : " + ex.Message, "E");
                        }
                }
                #endregion

                // string에서 공백을 제거한다.
                #region private string ExtendedTrim(string source)
                private string ExtendedTrim(string source)
                {
                        return source.TrimEnd('\0').Trim();
                }
                #endregion

                #endregion
        }
}
```

[Non-activity-Oriented Hazard Identification System: System Code]

```
using System;
using System.Data;
using System.Drawing;
using System.Windows.Forms;
using System.Text;
using System.Net.Sockets;
using HAZ_IDF.BIZ;

namespace HAZ_IDF.HAZ_IDF_CS.PEOPLE
{
   public struct AcPoint
   {
      public double x;
      public double y;
   };

   public struct div_t
   {
      public int quot;
      public int rem;
   };

   public struct tag_data_t
   {
      public string   name;
      public int     tag_id;
      public DateTime tag_date;
      public string   strStory;
      public double   x;
      public double   y;
   };



        public partial class PEO_MNTR_VW : Form
        {
       string m_strTagID = "";
      string m_strStory = "";
      string m_strName = "";
//       string m_strTagID = "32";
//       string m_strStory = "지상 1층";
//       string m_strName = "김현수";
      //
      DateTime m_LoadTime;
      tag_data_t[] m_arTagData;
      int m_nInCount = 0;
      int m_nMaxCount = 0;
      double m_dMBaseX = 0.0;
      double m_dMBaseY = 0.0;
      double m_TagetX = 0.0;
      double m_TagetY = 0.0;

      //= new tag_data_t[1000];
```

```
        private div_t  div (int numer, int denom)
        {
           div_t result;

           result.quot = numer / denom;
           result.rem = numer % denom;

           if (numer < 0 && result.rem > 0) {
                  /* did division wrong; must fix up */
                  ++result.quot;
                  result.rem -= denom;
           }

           return result;
        }

        public double dtr(double dgr)
        {
           if (Math.Abs(dgr) < 1e-5) return 0.0;
           return dgr * Math.PI / 180.0;
        }

        public double Dist2D(AcPoint pnt1, AcPoint pnt2)
        {
           double dx = Math.Abs(pnt1.x - pnt2.x);
           double dy = Math.Abs(pnt1.y - pnt2.y);
           return Math.Sqrt((dx * dx) + (dy * dy));
        }
        public AcPoint Polar(AcPoint pnt1, double ang, double dist)
        {
           AcPoint pc;
           if (Math.Abs(dist) < 1e-5)
           {
              pc.x = pnt1.x;
              pc.y = pnt1.y;
              return pc;
           }
           double ca = Math.Cos(ang);
           double sa = Math.Sin(ang);
           if (Math.Abs(ca) < 1e-5) pc.x = pnt1.x;
           else pc.x = pnt1.x + (dist * ca);

           if (Math.Abs(sa) < 1e-5) pc.y = pnt1.y;
           else pc.y = pnt1.y + (dist * sa);

           return pc;
        }

                  public PEO_MNTR_VW()
                  {
                          InitializeComponent();
                          //this.WindowState = FormWindowState.Maximized;
                  }

                  #region  private  void  PEO_MNTR_VW_Load(object  sender,
EventArgs e)
                  private  void  PEO_MNTR_VW_Load(object  sender,  EventArgs
e)
                  {
           PEOPLE_Biz biz = new PEOPLE_Biz();
```

```
          DataSet                          ds                          =
biz.SELECT_T_TAG_VALUE_ARRAY(((MainForm)this.MdiParent).PjtID);

          m_nMaxCount = ds.Tables[0].Rows.Count;
          m_arTagData = new tag_data_t[m_nMaxCount];
          for (int i = 0; i < m_nMaxCount; i++)
          {
             m_arTagData[i].name                                      =
ds.Tables[0].Rows[i]["PEO_NAME"].ToString();
             m_arTagData[i].tag_id                                    =
Convert.ToInt32(ds.Tables[0].Rows[i]["TAG_ID"].ToString());
             m_arTagData[i].tag_date                                  =
Convert.ToDateTime(ds.Tables[0].Rows[i]["TAG_DATE"].ToString());
             m_arTagData[i].strStory                                  =
ds.Tables[0].Rows[i]["TAG_STORY"].ToString();
             m_arTagData[i].x                                         =
Convert.ToDouble(ds.Tables[0].Rows[i]["TAG_X"].ToString());
             m_arTagData[i].y                                         =
Convert.ToDouble(ds.Tables[0].Rows[i]["TAG_Y"].ToString());
          }

          DataSet                      PjtInfoDs                      =
biz.SELECT_T_PJT_INFO(((MainForm)this.MdiParent).PjtID);
          m_dMBaseX                                                   =
Convert.ToDouble(PjtInfoDs.Tables[0].Rows[0]["CFG_BASEX"].ToString());
          m_dMBaseY                                                   =
Convert.ToDouble(PjtInfoDs.Tables[0].Rows[0]["CFG_BASEY"].ToString());

                         //-----------------------------
                         // timer 설정
                         timer1.Interval = 1000;
                         timer1.Start();
                         //-----------------------------

         // m_LoadTime = DateTime.Now;

          gridMonitorDataStepBind();
                 }
                 #endregion


       private int Clockwise(AcPoint pt, AcPoint p1, AcPoint p2)
       {
                // 리턴값: 1=시계방향, -1=반시계방향, 0=3점직선
          double value = p1.x * p2.y - pt.y * p1.x - pt.x * p2.y - p1.y
* p2.x + pt.x * p1.y + pt.y * p2.x;

          if (Math.Abs(value) < 1e-5) return 0; // 방향성이없음 즉 3점이
직선임
               if (value > 0) return -1; // 반시계방향
               else if (value < 0) return 1; // 시계방향
               return 0; // 방향성이없음 즉 3점이 직선임
       }


//////////////////////////////////////////////////////////////////////////
//////////////////
       // pt1~pt2, pt3~pt4 두 선분의 교차여부 계산
```

```
	// 리턴값: -1 = 교차 안함
	//                  :  0 = 한 선분의 끝점이 다른 선분에 포함됨
	//                  :  1 = 교차
	private int intersect_ccw(AcPoint pt1, AcPoint pt2, AcPoint pt3,
AcPoint pt4)
	{
		AcPoint p1 = pt1;
		AcPoint p2 = pt2;
		AcPoint p3 = pt3;
		AcPoint p4 = pt4;

		if (p1.x > p2.x)
		{
			double temp;
			temp = p1.x;
			p1.x = p2.x;
			p2.x = temp;


			temp = p1.y;
			p1.x = p2.y;
			p2.y = temp;
		}
		if (p3.x > p4.x)
		{
			double temp;
			temp = p3.x;
			p3.x = p4.x;
			p4.x = temp;


			temp = p3.y;
			p3.x = p4.y;
			p4.y = temp;
		}

		int r123 = Clockwise(p1, p2, p3);
		int r124 = Clockwise(p1, p2, p4);
		int r341 = Clockwise(p3, p4, p1);
		int r342 = Clockwise(p3, p4, p2);

		//DBPRINTF (">> r123=%d, r124=%d, r341=%d, r342=%d \n", r123,
r124, r341, r342);

		// 교차하는경우
		if ((r123 * r124) < 0 && (r341 * r342) < 0) return 1;

		// 평행인경우
		if (r123 == 0 && r124 == 0)
		{
			// 두선분이 평행이면서 overlapping된 상태

			//if (!(p3.x > p2.x || p1.x > p4.x)) return 0;
			if (!((p3.x - p2.x) >= 1e-6 || (p1.x - p4.x) >= 1e-6))
return 0;

			// 두선분이 서로 평행이면서 서로 떨어져 있음
			else return -1;
```

```
        }

        // 한점이 다른 선분의 위에 있거나 연장선 상에 있는 경우
        if (r123 == 0)
        {
            // p3가 p1-p2선 상에 있는 경우

            //if (p3.x >= p1.x && p3.x <= p2.x && p3.y >= p1.y && p3.y
<= p2.y) return 0;
            if ((p3.x - p1.x) >= 1e-6 && (p2.x - p3.x) >= 1e-6 &&
(p3.y - p1.y) >= 1e-6 && (p2.y - p3.y) >= 1e-6) return 0;

            // p3가 p1-p2선 상의 전이나 후에 있는 경우
            else return -1;
        }
        if (r124 == 0)
        {
            // p4가 p1-p2선 상에 있는 경우
            //if (p4.x >= p1.x && p4.x <= p2.x && p4.y >= p1.y && p4.y
<= p2.y) return 0;
            if ((p4.x - p1.x) >= 1e-6 && (p2.x - p4.x) >= 1e-6 &&
(p4.y - p1.y) >= 1e-6 && (p2.y - p4.y) >= 1e-6) return 0;
            // p4가 p1-p2선 상의 전이나 후에 있는 경우
            else return -1;
        }
        if (r341 == 0)
        {
            // p1이 p3-p4선 상에 있는 경우
            //if (p1.x >= p3.x && p1.x <= p4.x && p1.y >= p3.y && p1.y
<= p4.y) return 0;
            if ((p1.x - p3.x) >= 1e-6 && (p4.x - p1.x) >= 1e-6 &&
(p1.y - p3.y) >= 1e-6 && (p4.y - p1.y) >= 1e-6) return 0;
            // p1이 p3-p4선 상의 전이나 후에 있는 경우
            else return -1;
        }
        if (r342 == 0)
        {
            // p2가 p3-p4선 상에 있는 경우
            //if (p2.x >= p3.x && p2.x <= p4.x && p2.y >= p3.y && p2.y
<= p4.y) return 0;
            if ((p2.x - p3.x) >= 1e-6 && (p4.x - p2.x) >= 1e-6 &&
(p2.y - p3.y) >= 1e-6 && (p4.y - p2.y) >= 1e-6) return 0;
            // p2가 p3-p4선 상의 전이나 후에 있는 경우
            else return -1;
        }
        // 교차하지 않는 경우
        return -1;
    }


    #region private bool isInArea(Point[] arPt, Point pt)
    private bool isInArea(AcPoint[] arPt, AcPoint pt)
    {
        Int32 count = arPt.Length;
        if (count == 0) return false;


        AcPoint p1, p2, q1, q2, comp;
```

```
comp = arPt[0];
double len = 0.0;
long i;
for (i = 0; i < count - 1; i++)
{
   p1 = arPt[i];
   p2 = arPt[i + 1];
   if ((i != 0) && (comp.x == p1.x) && (comp.y == p1.y))
   {
      break;
   }
   len += Dist2D(p1, p2);
}

p1 = pt;
p2 = Polar(p1, dtr(0.0), len);

int endp = 0;
int nx = 0;
int rc;
for (i = 0; i < count - 1; i++)
{
   q1 = arPt[i];
   q2 = arPt[i + 1];
   if ((i != 0) && (comp.x == q1.x) && (comp.y == q1.y))
   {
      break;
   }

   rc = intersect_ccw(p1, p2, q1, q2);
   if (rc > 0) nx++;
   else if (rc == 0)
   {
      endp++;
      if (endp >= 2)
      {
         nx++;
         endp = 0;
      }
   }
}

p2 = Polar(p1, dtr(90.0), len);

endp = 0;
int ny = 0;
for (i = 0; i < count - 1; i++)
{
   q1 = arPt[i];
   q2 = arPt[i + 1];
   if ((i != 0) && (comp.x == q1.x) && (comp.y == q1.y))
   {
      break;
   }

   rc = intersect_ccw(p1, p2, q1, q2);
   if (rc > 0) ny++;
   else if (rc == 0)
   {
```

```
                endp++;
                if (endp >= 2)
                {
                    ny++;
                    endp = 0;
                }
            }
        }

        //DBPRINTF (">> len=%f, nx=%d, ny=%d \n", len, nx, ny);
        div_t dvx = div(nx, 2);
        div_t dvy = div(ny, 2);

        if (dvx.rem == 1 && dvy.rem == 1) return true;

        return false;
    }
    #endregion

    // 위험지역은 사각형 박스라고 가정하에 처리함
    #region private bool isAreaInTagPt(Point[] arPt, Point pt)
    private  bool  isAreaInPt(AcPoint  ptLB,  AcPoint  ptRT,  AcPoint
ptTag)
    {
        if (((ptLB.x < ptTag.x) && (ptRT.x > ptTag.x))
        && ((ptLB.y < ptTag.y) && (ptRT.y > ptTag.y)))
        {
            return true;
        }
        return false;
    }
    #endregion


    #region private void gridMonitorDataBind()
    private void gridMonitorDataBind()
    {
         //P.PEO_NAME, V.TAG_STORY, V.TAG_X, V.TAG_Y, V.TAG_DATE
        PEOPLE_Biz biz = new PEOPLE_Biz();
        DataSet                          ds                          =
biz.SELECT_T_TAG_VALUE(((MainForm)this.MdiParent).PjtID, m_LoadTime);
        Int32 nCount = ds.Tables[0].Rows.Count;
        if (nCount > 0)
        {
            DataSet                     PjtInfoDs                    =
biz.SELECT_T_PJT_INFO(((MainForm)this.MdiParent).PjtID);
            double                      dMBaseX                      =
Convert.ToDouble(PjtInfoDs.Tables[0].Rows[0]["CFG_BASEX"].ToString());
            double                      dMBaseY                      =
Convert.ToDouble(PjtInfoDs.Tables[0].Rows[0]["CFG_BASEY"].ToString());
            for (int i = 0; i < nCount; i++)
            {
                string                      Name                     =
ds.Tables[0].Rows[i]["PEO_NAME"].ToString();
                string                      Story                    =
ds.Tables[0].Rows[i]["TAG_STORY"].ToString();
                string x = ds.Tables[0].Rows[i]["TAG_X"].ToString();
                string y = ds.Tables[0].Rows[i]["TAG_Y"].ToString();
                string                      time                     =
```

```
ds.Tables[0].Rows[i]["TAG_DATE"].ToString();

                DataGridViewRowCollection            rows            =
LIST_TAGRECEIVE_INFO.Rows;
                string[] row = { Name, Story, x, y, time };
                rows.Add(row);

                AcPoint pt;
                pt.x = Convert.ToDouble(x.ToString());
                pt.y = Convert.ToDouble(y.ToString());

                // 위험지역에 들어섰는지 체크
                DataSet                     dsDan                    =
biz.SELECT_T_STORY_TO_STY_ID(((MainForm)this.MdiParent).PjtID,   "A단지",
"A동", Story);

                int nCountTmp = dsDan.Tables[0].Rows.Count;
                if (nCountTmp < 1)
                {
                   return;
                }

                string                   strStoryID                  =
dsDan.Tables[0].Rows[0]["STY_ID"].ToString();
                dsDan                                                =
biz.SELECT_T_DANGERAREA(((MainForm)this.MdiParent).PjtID, strStoryID);
                nCountTmp = dsDan.Tables[0].Rows.Count;
                if (nCountTmp < 1)
                {
                   return;
                }
                for (int j = 0; j < nCountTmp; j++)
                {
                   double                  dBoundary                 =
Convert.ToDouble(dsDan.Tables[0].Rows[j]["DGA_DEG"].ToString());
                   double                    MinX                    =
Convert.ToDouble(dsDan.Tables[0].Rows[j]["DGA_MIN_X"].ToString());
                   double                    MaxX                    =
Convert.ToDouble(dsDan.Tables[0].Rows[j]["DGA_MAX_X"].ToString());
                   double                    MinY                    =
Convert.ToDouble(dsDan.Tables[0].Rows[j]["DGA_MIN_Y"].ToString());
                   double                    MaxY                    =
Convert.ToDouble(dsDan.Tables[0].Rows[j]["DGA_MAX_Y"].ToString());
                   // 장비 기준점 빼고 범위 늘리고
                   AcPoint ptLB, ptRT;
                   ptLB.x = MinX - dBoundary - dMBaseX;
                   ptLB.y = MinY - dBoundary - dMBaseY;
                   ptRT.x = MaxX + dBoundary - dMBaseX;
                   ptRT.y = MaxY + dBoundary - dMBaseY;

                   if(isAreaInPt(ptLB, ptRT, pt))   // 위험지역에 들어섬
                   {
                      DataSet                  dsTemp                 =
biz.SELECT_T_DANGERKIND_TO_DAKNAME(((MainForm)this.MdiParent).PjtID,
dsDan.Tables[0].Rows[j]["DAK_ID"].ToString());
                      string                 strDakName               =
dsTemp.Tables[0].Rows[0]["DAK_NAME"].ToString();
                      DataGridViewRowCollection        rowsDan        =
LIST_DANGER_INFO.Rows;
```

```
                        // 위험원 하드코딩
                        DataSet                dsPeoDanVal                =
biz.SELECT_T_PEOPLE_DANGER_VAL(((MainForm)this.MdiParent).PjtID,
Name.ToString());
                        float                   danVal                    =
Convert.ToSingle(dsPeoDanVal.Tables[0].Rows[0]["PEO_DANGER_VAL"].ToStrin
g());
                        Int32 nDanVal = Convert.ToInt32(danVal * 100.0);

                        // 위험지역에 이미 들어섰는지 체크
                        int nRowCount = LIST_DANGER_INFO.Rows.Count - 1;
                        bool bOn = false;
                        for (int d = nRowCount; d >= 0; d--)
                        {
                            string                 strTemp                =
LIST_DANGER_INFO.Rows[d].Cells[0].Value.ToString();
                            if (strTemp == Name)
                            {
                                bOn = true;
                            }
                        }
                        if (!bOn)
                        {
                            string[] rowDan = { Name, Story, strDakName,
nDanVal.ToString() };
                            rowsDan.Add(rowDan);
                        }
                    }
                    else                                // 위험지역이 아니면
위험지역에서 빠져나간건지 확인 체크
                    {
                        int nRowCount = LIST_DANGER_INFO.Rows.Count - 1;
                        for (int d = nRowCount; d >= 0; d--)
                        {
                            string                 strTemp                =
LIST_DANGER_INFO.Rows[d].Cells[0].Value.ToString();
                           if(strTemp == Name)
                           {
                               LIST_DANGER_INFO.Rows.RemoveAt(d);
                           }
                        }
                    }
                }
                //

            }
            m_LoadTime = DateTime.Now;
        }
    }
    #endregion


    #region private void gridMonitorDataStepBind()
    private void gridMonitorDataStepBind()
    {
        //P.PEO_NAME, V.TAG_STORY, V.TAG_X, V.TAG_Y, V.TAG_DATE

        if (m_nMaxCount <= m_nInCount)
        {
```

```
            return;
        }

        string Name = m_arTagData[m_nInCount].name;
        string Story = m_arTagData[m_nInCount].strStory;
        double x = m_arTagData[m_nInCount].x;
        double y = m_arTagData[m_nInCount].y;
        DateTime time = m_arTagData[m_nInCount].tag_date;

        DataGridViewRowCollection rows = LIST_TAGRECEIVE_INFO.Rows;
        string[]  row  =  {  Name,  Story,  x.ToString(),  y.ToString(),
time.ToString() };
        int nAdd = rows.Add(row);
        rows[nAdd].Selected = true;
        LIST_TAGRECEIVE_INFO.CurrentCell                              =
LIST_TAGRECEIVE_INFO.SelectedRows[0].Cells[0];

        AcPoint pt;
        pt.x = x;
        pt.y = y;

        // 위험지역에 들어섰는지 체크
        PEOPLE_Biz biz = new PEOPLE_Biz();
        DataSet                         dsDan                         =
biz.SELECT_T_STORY_TO_STY_ID(((MainForm)this.MdiParent).PjtID,   "A단지",
"A동", Story);

        int nCountTmp = dsDan.Tables[0].Rows.Count;
        if (nCountTmp < 1)
        {
            return;
        }

        string                       strStoryID                       =
dsDan.Tables[0].Rows[0]["STY_ID"].ToString();
        dsDan                                                         =
biz.SELECT_T_DANGERAREA(((MainForm)this.MdiParent).PjtID, strStoryID);
        nCountTmp = dsDan.Tables[0].Rows.Count;
        if (nCountTmp < 1)
        {
            return;
        }
        for (int j = 0; j < nCountTmp; j++)
        {
            double                     dBoundary                      =
Convert.ToDouble(dsDan.Tables[0].Rows[j]["DGA_DEG"].ToString());
            double                       MinX                         =
Convert.ToDouble(dsDan.Tables[0].Rows[j]["DGA_MIN_X"].ToString());
            double                       MaxX                         =
Convert.ToDouble(dsDan.Tables[0].Rows[j]["DGA_MAX_X"].ToString());
            double                       MinY                         =
Convert.ToDouble(dsDan.Tables[0].Rows[j]["DGA_MIN_Y"].ToString());
            double                       MaxY                         =
Convert.ToDouble(dsDan.Tables[0].Rows[j]["DGA_MAX_Y"].ToString());
            // 장비 기준점 빼고 범위 늘리고
            AcPoint ptLB, ptRT;
            ptLB.x = MinX - dBoundary - m_dMBaseX;
            ptLB.y = MinY - dBoundary - m_dMBaseY;
            ptRT.x = MaxX + dBoundary - m_dMBaseX;
```

```
            ptRT.y = MaxY + dBoundary - m_dMBaseY;

            if (isAreaInPt(ptLB, ptRT, pt))   // 위험지역에 들어섬
            {
                DataSet                    dsTemp                    =
biz.SELECT_T_DANGERKIND_TO_DAKNAME(((MainForm)this.MdiParent).PjtID,
dsDan.Tables[0].Rows[j]["DAK_ID"].ToString());
                string                   strDakName                  =
dsTemp.Tables[0].Rows[0]["DAK_NAME"].ToString();
                DataGridViewRowCollection           rowsDan          =
LIST_DANGER_INFO.Rows;
                // 위험원 하드코딩
                DataSet                  dsPeoDanVal                 =
biz.SELECT_T_PEOPLE_DANGER_VAL(((MainForm)this.MdiParent).PjtID,
Name.ToString());
                float                     danVal                     =
Convert.ToSingle(dsPeoDanVal.Tables[0].Rows[0]["PEO_DANGER_VAL"].ToStrin
g());
                Int32 nDanVal = Convert.ToInt32(danVal * 100.0);

                // 위험지역에 이미 들어섰는지 체크
                int nRowCount = LIST_DANGER_INFO.Rows.Count - 1;
                bool bOn = false;
                for (int d = nRowCount; d >= 0; d--)
                {
                    string                  strTemp                  =
LIST_DANGER_INFO.Rows[d].Cells[0].Value.ToString();
                    if (strTemp == Name)
                    {
                        bOn = true;
                    }
                }
                if (!bOn)
                {
                    string[]  rowDan  =  {  Name,  Story,  strDakName,
nDanVal.ToString() };
                    rowsDan.Add(rowDan);
                }
            }
            else                          // 위험지역이 아니면 위험지역에서
빠져나간건지 확인 체크
            {
                int nRowCount = LIST_DANGER_INFO.Rows.Count - 1;
                for (int d = nRowCount; d >= 0; d--)
                {
                    string                  strTemp                  =
LIST_DANGER_INFO.Rows[d].Cells[0].Value.ToString();
                    if (strTemp == Name)
                    {
                        LIST_DANGER_INFO.Rows.RemoveAt(d);
                    }
                }
            }
        }
        //
    }
    #endregion

//              #region                  private                  void
```

```
gridMonitor_AfterSelectChange(object                                sender,
Infragistics.Win.UltraWinGrid.AfterSelectChangeEventArgs e)
//              private    void    gridMonitor_AfterSelectChange(object
sender, Infragistics.Win.UltraWinGrid.AfterSelectChangeEventArgs e)
//              {
//                      if   (gridMonitor.Selected.Rows.Count   ==   0)
return;
//
//                      txtName.Text                                    =
gridMonitor.Selected.Rows[0].Cells["PEO_NAME"].Value.ToString();
//                      txtCareer.Text                                  =
gridMonitor.Selected.Rows[0].Cells["PEO_CAREER"].Value.ToString()   +   "
년";
//                      txtComp.Text                                    =
gridMonitor.Selected.Rows[0].Cells["PEO_COMP"].Value.ToString();
//                      txtWorkItem.Text                                =
gridMonitor.Selected.Rows[0].Cells["WRK_NAME"].Value.ToString();
//              }
//              #endregion
//
//              #region  private  void  dateMonitor_ValueChanged(object
sender, EventArgs e)
//              private  void  dateMonitor_ValueChanged(object  sender,
EventArgs e)
//              {
//                      gridMonitorDataBind();
//                      pbDanger.Refresh();
//              }
//              #endregion

                /// <summary>
                /// 위치정보를 그리자
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region  private  void  pbDanger_Paint(object  sender,
PaintEventArgs e)
                private     void     pbDanger_Paint(object     sender,
PaintEventArgs e)
                {
          if (m_strTagID == "")
          {
             return;
          }
          PEOPLE_Biz biz = new PEOPLE_Biz();

          DataSet                         ds                          =
biz.SELECT_T_OUTLINE(((MainForm)this.MdiParent).PjtID, m_strStory);

          int nCount = ds.Tables[0].Rows.Count;
          if (nCount < 1)
          {
             return;
          }

          string strPoly = ds.Tables[0].Rows[0]["OTL_POLY"].ToString();
          int nSize = 0;
          for (int s = 0; s < strPoly.Length; s++)
          {
```

```
            if (strPoly[s] == ',')
            {
                nSize++;
            }
        }


        AcPoint[] arPt = new AcPoint[nSize / 2];
        arPt.Initialize();
        int nIndex = strPoly.IndexOf(',');
        int nInsert = 0;
        while (nIndex > 0)
        {
            arPt[nInsert].x   =   Convert.ToDouble(strPoly.Substring(0,
nIndex).ToString());
            strPoly = strPoly.Remove(0, nIndex + 1);
            nIndex = strPoly.IndexOf(',');
            arPt[nInsert].y   =   Convert.ToDouble(strPoly.Substring(0,
nIndex).ToString());
            strPoly = strPoly.Remove(0, nIndex + 1);
            nIndex = strPoly.IndexOf(',');
            nInsert++;
        }

        int nLength = arPt.Length;

        // 외곽라인의 최대, 최소값을 가져온다.
        AcPoint pt, comp;
        comp = arPt[0];

        double max_x = comp.x;
        double max_y = comp.y;
        double min_x = comp.x;
        double min_y = comp.y;
        float len_x = 0.0f;
        float len_y = 0.0f;

        for (int i = 0; i < nLength; i++)
        {
            pt = arPt[i];
            if ((i != 0) && (pt.x == comp.x) && (pt.y == comp.y))
            {
                break;
            }
            if (max_x < pt.x)
            {
                max_x = pt.x;
            }
            if (min_x > pt.x)
            {
                min_x = pt.x;
            }
            if (max_y < pt.y)
            {
                max_y = pt.y;
            }
            if (min_y > pt.y)
            {
                min_y = pt.y;
```

```
            }
        }

        len_x = Convert.ToSingle(max_x - min_x);
        len_y = Convert.ToSingle(max_y - min_y);

        float max_sc_x = pbDanger.Width - 1;
        float max_sc_y = pbDanger.Height - 1;

        // 축소비율
        float rate_x = max_sc_x / len_x;
        float rate_y = max_sc_y / len_y;

        Graphics g = e.Graphics;
        Pen pen = new Pen(Color.Black, 1);

        // 외곽라인 표시 (T_OUTPOINT에 저장된 값을 가져온다)
        PointF[] polygon = new PointF[nLength];
        for (int i = 0; i < nLength; i++)
        {
            AcPoint ptT;
            ptT = arPt[i];
            float x = Convert.ToSingle(ptT.x - min_x);
            float y = Convert.ToSingle(ptT.y - min_y);

            PointF ptF = new PointF(x * rate_x, max_sc_y - y *
rate_y);
            polygon[i] = ptF;

            //g.DrawLine(pen, x1 * rate_x, pbDanger.Height - y1 *
rate_y, x2 * rate_x, pbDanger.Height - y2 * rate_y);
            //g.DrawLine(pen, x1 * rate_x, y1 * rate_y, x2 * rate_x,
y2 * rate_y);
        }
        SolidBrush blueBrush = new SolidBrush(Color.White);
        g.FillPolygon(blueBrush, polygon);


        // 위험지역 그리기
        ds                                                          =
biz.SELECT_T_STORY_TO_STY_ID(((MainForm)this.MdiParent).PjtID,   "A단지",
"A동", m_strStory);

        nCount = ds.Tables[0].Rows.Count;
        if (nCount < 1)
        {
            return;
        }

        string strStoryID = ds.Tables[0].Rows[0]["STY_ID"].ToString();
        ds = biz.SELECT_T_DANGERAREA(((MainForm)this.MdiParent).PjtID,
strStoryID);
        nCount = ds.Tables[0].Rows.Count;
        if (nCount < 1)
        {
            return;
        }
        Pen penDan = new Pen(Color.Red, 1);
        for (int i = 0; i < nCount; i++ )
```

```
          {
             double                          MinX                          =
Convert.ToDouble(ds.Tables[0].Rows[i]["DGA_MIN_X"].ToString());
             double                          MaxX                          =
Convert.ToDouble(ds.Tables[0].Rows[i]["DGA_MAX_X"].ToString());
             double                          MinY                          =
Convert.ToDouble(ds.Tables[0].Rows[i]["DGA_MIN_Y"].ToString());
             double                          MaxY                          =
Convert.ToDouble(ds.Tables[0].Rows[i]["DGA_MAX_Y"].ToString());

             float fMinX = Convert.ToSingle(MinX - min_x);
             float fMaxX = Convert.ToSingle(MaxX - min_x);
             float fMinY = Convert.ToSingle(MinY - min_y);
             float fMaxY = Convert.ToSingle(MaxY - min_y);
             float fWidth = Convert.ToSingle(Math.Abs(MaxX - MinX));
             float fHeight = Convert.ToSingle(Math.Abs(MaxY - MinY));

             // 해당 위험지역의 컬러
             SolidBrush dgaBrush = new SolidBrush(Color.Red);
             int                          nColor                          =
Convert.ToInt32(ds.Tables[0].Rows[i]["DGA_COLOR"].ToString());
             switch (nColor)
             {
                case 1:
                   dgaBrush.Color = Color.Red;
                   break;
                case 2:
                   dgaBrush.Color = Color.Orange;
                   break;
                case 3:
                   dgaBrush.Color = Color.Yellow;
                   break;
                case 4:
                   dgaBrush.Color = Color.Green;
                   break;
                case 5:
                   dgaBrush.Color = Color.Blue;
                   break;
             }

             float fDgrMinX = fMinX * rate_x;
             float fDgrMaxY = max_sc_y - fMaxY * rate_y;
             float fDgrWidth = fWidth * rate_x;
             float fDgrHeight = fHeight * rate_y;

             g.FillRectangle(dgaBrush,  fDgrMinX,  fDgrMaxY,  fDgrWidth,
fDgrHeight);
             g.DrawRectangle(pen,   fDgrMinX,   fDgrMaxY,   fDgrWidth,
fDgrHeight);
             //g.DrawLine(pen,  x1  *  rate_x,  pbDanger.Height  -  y1  *
rate_y, x2 * rate_x, pbDanger.Height - y2 * rate_y);
             //g.DrawLine(pen, x1 * rate_x, y1 * rate_y, x2 * rate_x,
y2 * rate_y);
          }

          g.DrawPolygon(pen, polygon);

          // 사람 표시
          ds                                                             =
```

```
biz.SELECT_T_TAG_VALUE_TO_TAG(((MainForm)this.MdiParent).PjtID,
m_strTagID);
          nCount = ds.Tables[0].Rows.Count;
          if (nCount < 1)
          {
             return;
          }
          float TagX = Convert.ToSingle(m_TagetX);
          float TagY = Convert.ToSingle(m_TagetY);
          //TagX = TagX - Convert.ToSingle(min_x);
          //TagY = TagY - Convert.ToSingle(min_y);

          Image imgPerson = Image.FromFile("Icon\\Person.ico");
          float drawW = (TagX * rate_x) - (imgPerson.Width / 2);
          float   drawH   =   pbDanger.Height   -   (TagY   *   rate_y)   -
(imgPerson.Height / 2);
          g.DrawImage(imgPerson, drawW, drawH);

          // 이름 Display
          string strName = m_strName;
          Font font = new Font("Arial", 9);
          SolidBrush solidBrush = new SolidBrush(Color.Black);

          g.DrawString(strName,  font,  solidBrush,  TagX  *  rate_x  -
imgPerson.Width / 2, pbDanger.Height - TagY * rate_y + imgPerson.Height
/ 2);
                }
                #endregion

                /// <summary>
                /// 일정시간마다 데이터 업뎃
                /// </summary>
                /// <param name="sender"></param>
                /// <param name="e"></param>
                #region   private   void   timer1_Tick(object   sender,
EventArgs e)
                private void timer1_Tick(object sender, EventArgs e)
                {
                        //gridMonitorDataBind();
          gridMonitorDataStepBind();
                        pbDanger.Refresh();
          m_nInCount++;
                }
                #endregion

                #region  private  void  btnClose_Click(object  sender,
EventArgs e)
                private void btnClose_Click(object sender, EventArgs e)
                {
                        this.Close();
                }
                #endregion

       private void btnAlarm_Click(object sender, EventArgs e)
       {
          int iSelCount = LIST_DANGER_INFO.SelectedRows.Count;
          //int iRow = LIST_DANGER_INFO.SelectedCells[0].RowIndex;
          if (iSelCount == 0)
          {
```

```
            MessageBox.Show("선택한 위험 정보가 없습니다.", "위험지역 검색
관리", MessageBoxButtons.OK, MessageBoxIcon.Information);
        }

        string                        strZone                        =
LIST_DANGER_INFO.SelectedRows[0].Cells[2].Value.ToString();
        PEOPLE_Biz biz = new PEOPLE_Biz();

        // TODO : 총 임시 하드코딩 (2009-06-17)
        DataSet ds = biz.SELECT_T_DANGERKIND_TO_DAKIDX(strZone);

        string strDakIdx = ds.Tables[0].Rows[0]["DAK_IDX"].ToString();

        PJT_INFO_Biz bizPjt = new PJT_INFO_Biz();
        DataSet                        dsPjt                          =
bizPjt.SELECT_T_CONFIG(((MainForm)this.MdiParent).PjtID);

        string                      strAlramNum                       =
dsPjt.Tables[0].Rows[0]["CFG_ALRAM_NUM"].ToString();
        string                    strAlramInterval                    =
dsPjt.Tables[0].Rows[0]["CFG_ALRAM_INTERVAL"].ToString();

        UdpClient udpClient = new UdpClient();
        Byte btTagID = Convert.ToByte(m_strTagID);
        Byte[]          sendBytes          =          {          0,
Convert.ToByte(((MainForm)this.MdiParent).PjtID), 0, btTagID, 0, btTagID,
Convert.ToByte(strDakIdx),                    Convert.ToByte(strDakIdx),
Convert.ToByte(strAlramInterval) };
        try
        {
            Int32 iState = udpClient.Send(sendBytes, sendBytes.Length,
((MainForm)this.MdiParent).SERVER_IP, 4001);
        }
        catch (System.Exception ex)
        {
            MessageBox.Show("알람 메시지가 전송되지 못하였습니다.", "위험지역
검색     관리",     MessageBoxButtons.OK,     MessageBoxIcon.Information);

        }
    }

    // 위험정보 리스트 클릭시
    #region private void LIST_DANGER_INFO_SelectionChanged(object
sender, EventArgs e)
    private void LIST_DANGER_INFO_SelectionChanged(object sender,
EventArgs e)
    {
        //int iSelCount = LIST_DANGER_INFO.SelectedRows.Count;
        ////int                      selCellCount                     =
LIST_DANGER_INFO.GetCellCount(DataGridViewElementStates.Selected);

        //if (iSelCount == 1)
        //{
        //                              string       strName        =
LIST_DANGER_INFO.SelectedRows[0].Cells[0].Value.ToString();
        //   PEOPLE_Biz biz = new PEOPLE_Biz();

        //   DataSet ds = biz.SELECT_T_PEOPLE_TO_TAGID(strName);
```

```
          //                                    if(m_strTagID         !=
ds.Tables[0].Rows[0]["TAG_ID"].ToString())
          //    {
          //                                          m_strTagID      =
ds.Tables[0].Rows[0]["TAG_ID"].ToString();
          //                                          m_strStory      =
LIST_DANGER_INFO.SelectedRows[0].Cells[1].Value.ToString();
          //                                           m_strName      =
LIST_DANGER_INFO.SelectedRows[0].Cells[0].Value.ToString();
          //       pbDanger.Refresh();
          //    }

          //}
       }
       #endregion


       // 위험 리스트에서 아이템이 해제가 되면 PictureBox Draw 해제 여부 판단하여
제거
       #region private void LIST_DANGER_INFO_RowsRemoved(object sender,
DataGridViewRowsRemovedEventArgs e)
       private    void    LIST_DANGER_INFO_RowsRemoved(object    sender,
DataGridViewRowsRemovedEventArgs e)
       {
          //if (LIST_DANGER_INFO.SelectedRows.Count == 0)
          //{
          //    m_strTagID = "";
          //    m_strStory = "";
          //    m_strName = "";
          //    pbDanger.Refresh();
          //}
       }
       #endregion

       #region private void LIST_TAGRECEIVE_INFO_SelectionChanged(object
sender, EventArgs e)
       private void LIST_TAGRECEIVE_INFO_SelectionChanged(object sender,
EventArgs e)
       {
          int iSelCount = LIST_TAGRECEIVE_INFO.SelectedRows.Count;
          //int                       selCellCount                      =
LIST_DANGER_INFO.GetCellCount(DataGridViewElementStates.Selected);

          if (iSelCount == 0) return;

          string                        strName                         =
LIST_TAGRECEIVE_INFO.SelectedRows[0].Cells[0].Value.ToString();
          PEOPLE_Biz biz = new PEOPLE_Biz();
          DataSet                           ds                          =
biz.SELECT_T_PEOPLE_INFO(((MainForm)this.MdiParent).PjtID, strName);

          txtName.Text = ds.Tables[0].Rows[0]["PEO_NAME"].ToString();
          txtCareer.Text = ds.Tables[0].Rows[0]["PEO_CAREER"].ToString()
+ " 년";
          txtComp.Text = ds.Tables[0].Rows[0]["PEO_COMP"].ToString();
          txtWorkItem.Text                                              =
ds.Tables[0].Rows[0]["WRK_NAME"].ToString();

          // TODO : 층 임시 하드코딩 (2009-06-17)
```

```
          DataSet dsTagId = biz.SELECT_T_PEOPLE_TO_TAGID(strName);

          //if                      (m_strTagID                       !=
dsTagId.Tables[0].Rows[0]["TAG_ID"].ToString())
          //{
             m_strTagID                                                 =
dsTagId.Tables[0].Rows[0]["TAG_ID"].ToString();
             m_strStory                                                 =
LIST_TAGRECEIVE_INFO.SelectedRows[0].Cells[1].Value.ToString();
             m_strName                                                  =
LIST_TAGRECEIVE_INFO.SelectedRows[0].Cells[0].Value.ToString();
//                 pbDanger.Refresh();
             m_TagetX                                                   =
Convert.ToDouble(LIST_TAGRECEIVE_INFO.SelectedRows[0].Cells[2].Value.ToS
tring());
             m_TagetY                                                   =
Convert.ToDouble(LIST_TAGRECEIVE_INFO.SelectedRows[0].Cells[3].Value.ToS
tring());
          //}

       }
       #endregion
    }
}
```

초 록

미국 국립산업안전보건연구원은 위험원 인지 부족, 작업 위치 미파악, 미숙련 근로자 등의 건설 근로자의 부상 및 사망에 영향을 주는 요소들을 제시하였다. 이러한 요소들 중에서 위험원 인지는 건설안전관리 수행을 위한 가장 첫 단계라고 할 수 있다. 건설안전관리는 위험원 인지, 위험성 평가, 그리고 위험원 대응으로 구성되어 있다. 각각의 단계가 모두 중요하지만, 위험원을 인지하는 것은 가장 근본적인 단계이다. 인지되지 않은 위험원은 위험성 평가와 위험원 대응 단계에서 고려대상이 되지 않기 때문이다. 이는 발견되지 않은 위험은 위험도가 산정되지 않으며, 대응책도 수립되지 못하는 것을 의미한다. 그 결과로, 인지되지 않은 위험원은 그 자체가 제거될 때까지 건설 근로자에게 사고를 발생시킬 수 있다.

건설 근로자가 그들의 업무를 시작할 때, 근로자는 작업지역을 향해 움직이며, 도착한 후에 작업을 시작한다. 이러한 상황에서는 건설 근로자는 이동 과정과 작업을 수행하는 과정에서 위험원에 노출 될 수 있다. 본 연구는 건설 근로자가 이동하거나 작업 중에 직면할 수 있는 위험원을 작업에 의해 생성된 위험원과 비작업에 의해 생성된 위험원 두 가지로 분류한다.

본 연구는 두 가지 종류의 위험원을 식별할 수 있는 시스템을 개발한다. 두 가지의 위험원을 식별하기 위해 각기 다른 접근법이 시도

되었다. 작업에 의해 생성된 위험원은 과거 재해 사례를 통해 식별할 수 있다. 과거의 유사 재해사례를 검색하고 해당 현장에 적용하기 위해서는 질의문을 만들기 위한 색인들이 필요하다. 색인들은 BIM과 PMIS에서 추출된다. 추출된 색인들을 기반으로, 과거 유사 재해사례를 검색하기 위한 질의문이 생성될 수 있다. 질의문의 생성 이후, 검색 엔진은 유사 재해 사례를 찾고 유사도를 산정한다. 최종적인 결과는 현재 현장의 작업 중에 과거 재해사례와 가장 유사한 작업, 그 작업의 수행시간과 위치정보를 포함한다. 또한 이러한 결과물은 푸시(Push) 시스템을 통해 자동으로 건설 근로자에게 제공된다.

작업에 의해 생성되지 않은 위험원 시스템은 작업자의 위치추적 정보와 BIM을 통해서 위험가능지역을 식별한다. 제안하는 방법은 BIM으로부터 추출한 노드를 통해 결정되는 최적 이동경로 (최단이동경로)와 RFID를 통한 실제 작업자의 이동 경로의 차이를 활용하여 위험가능지역을 발견한다. 이러한 프로세스를 거치고 나면, 위험가능지역은 미처리된 위험가능지역과 기처리된 위험가능 지역으로 나눌 수 있다. 미처리된 위험가능 지역에 관한 정보는 안전관리자들에게 전달되며, 안전관리자들은 사고예방대책을 수립할 수 있고 위험지역을 관리할 수 있다.

결론적으로 본 연구는 반복적인 작업을 배제한 자동화된 데이터 프로세싱을 통한 건설 안전관리에서의 위험원 식별의 효율을 증대시키는데 공헌을 할 수 있다. 자동화된 프로시저는 위험원을 신속히

찾을 수 있으며, 위험원이 대응책 없이 존치하는 시간을 감소할 수 있을 것이다. 이는 건설 근로자가 위험에 노출되는 시간을 줄일 수 있다.

주요어: 위험원 식별, 안전관리, 정보검색, 자동화 데이터 수집, 실시간 위치추적 시스템
학 번: 2010-30159