Mobile Location-Based Bridge Inspection Decision-Support System

Yongxin Hu

A Thesis

in

The Department

of

Building, Civil and Environmental Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Applied Science (Building Engineering) at
Concordia University
Montreal, Quebec, Canada

February 2006

CONCORDIA UNIVERSITY

Department of Building, Civil and Environmental Engineering

This is to certify that the thesis prepared

By: **Yongxin Hu**

Entitled: **Mobile Location-based Bridge Inspection Decision-Support System**

and submitted in partial fulfillment of the requirement for the degree of

Master of Applied Science (Building Engineering)

Complied with the regulations of the University and meets with the accepted standards with respect to originality and quality.

Signed by the final examining committee:

______________________________________Chair

______________________________________Examiner

______________________________________Examiner

______________________________________Examiner

______________________________________Supervisor

Approved by ______________________________________

Chair of Department or Graduate Program Director

_________20___ ______________________________________

Dean of Faculty

ABSTRACT

The conventional data collection in bridge inspection is paper-based and provides limited support for spatial visualization, which makes the data collection process inaccurate and inefficient. In addition, condition rating evaluation of bridge components from the collected data still depends on the subjective opinion of the inspector. Therefore, new technologies are motivating the development of new types of bridge inspection systems to improve the quality and efficiency of bridge inspection and to allow the inspectors to better focus on their tasks. In this research, we describe a new approach to support bridge inspection data collection using location-based computing and to evaluate condition rating using a Bayesian network that can learn from the previous bridge inspection data.

Using 3D bridge model, inspectors can visualize and analyze information related to the inspection of bridges based on various interaction methods and a database. For example, the locations of the defects found in previous inspections and registered in the database can facilitate inspection activities by guiding the inspector to the locations where new defects are likely to be found, and allowing him/her to mark new detected defects on the 3D model and save inspection information to the database automatically. In addition, elements and events involved in a specific task are registered according to their locations in a spatial database, and the activities supported by the mobile and wearable computers are aware of these locations using suitable positioning devices, such as the Global Positioning System. Moreover, a number of multimedia data sources based on Geographic Information Systems and CAD information are integrated to further extend the usability of the model. Furthermore, decision-theoretic reasoning in the system can take advantage of the availability of large amount of previous inspection data to create probabilistic cause-effect relationships using a learning-based Bayesian network. The decision-support system can help inspectors when making condition rating on site, thus reducing the subjectivity involved in this process.

The proposed approach is demonstrated by developing a prototype system that includes a database of about 300 bridges in Montreal and a 3D detailed model of one long-span bridge (The Jacques Cartier Bridge). The prototype system runs on a Tablet PC and is implemented in Java language. The results of the case studies are discussed to evaluate the usability and usefulness of the proposed approach.

ACKOWNLEGEMENTS

First, I would like to gratefully thank my supervisor, Dr. Amin Hammad, who gave me the opportunity to pursue my Master's degree and provided the initial concept for this research project. He is nice, knowledgeable, and respectable and understands how to find out the advantages and potential capabilities of his students. Without his trust, understanding, encouragement, patience, and instructions, it is hard to imagine that I could succeed in this research study. Having worked together with him for two years, I have learned from him not only how to do research, but also how to keep on moving forward to achieve goals.

Also, I would like to thank Mr. Guy Richard and Mr. René Gagnon in the structure department of the Ministry of Transportation of Quebec and Mr. Guy Mailhot from The Jacques Cartier and Champlain Bridge Incorporated for their fruitful cooperation and great help in providing me with knowledge, data, and manuals. I would like to thank Mr. Sylvie Laroche from the Ministry of Transportation of Quebec for providing the GIS data of bridges in Quebec and Mr. André Clément from the City of Montreal for providing the GIS data of Montreal.

During these demanding years of study, the support of our Infra-group was of extreme importance. I would like to thank Cheng Zhang, Elaheh Mozaffari, and Bechir Khabir for their cooperation and contribution in the development of the 3D model, database, and my programming skills. Also, I would like to thank Prasannna Devarakonda, who

provided information and helped me build the GIS and GPS parts in the project. In addition, the help of all other friends in this project is appreciated.

Finally, I would like to thank my wife Huiqiang Yu, my parents, and my family for their continuous encouragement and love. You are my motivation to pursue my graduate study. Without your contributions, it was impossible for me to succeed in this study.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVATIONS

Abbreviation	Description
2D	Two-dimensional
3D	Three-dimensional
AADT	Annual Average Daily Traffic
AASHTO	American Association of State Highway and Transportation Officials
ADTT	Average Daily Truck Traffic
AI	Artificial Intelligence
ANN	Artificial Neural Network
API	Application Programming Interface
BMS	Bridge Management System
BN	Bayesian Network
CBR	Case-Based Reasoning
DB	Database
DSS	Decision-Support System
ES	Expert System
FHWA	Federal Highway Administration
GIS	Geographic Information System
GPS	Global Positioning System
GUI	Graphical User Interface
HMD	Head-Mounted Display
JDBC	Java Database Connectivity
JESS	Java Expert System Shell
JMF	Java Media Framework
LBC	Location-Based Computing
LoD	Levels of Detail
MCR	Material Condition Rating
MTO	Ministry of Transportation of Ontario
MTQ	Ministry of Transportation of Quebec
NBI	National Bridge Inventory
NCHRP	National Cooperative Highway Research Program
NBIS	National Bridge Inspection Standards
OBMS	Ontario Bridge Management System
OSIMS	Ontario Structure Inspection Management System
OSIS	Ontario Structural Inventory System
PCR	Performance Condition Rating
SQL	Structured Query Language

CHAPTER 1 INTRODUCTION

1.1 GENERAL BACKGROUND

A nation's infrastructure systems play an essential role in the nation's economy, and preserving the condition of these systems is crucial for improving the economy strength. Infrastructure systems comprise a wide range of facilities that provide the public with vital services. These facilitates can be categorized into transportation, building, water and wastewater, telecommunication systems, etc. Transportation facilitates are the most demanding item among all categories of infrastructure systems.

Transportation infrastructure systems comprise highways and highway structures (mainly bridges), railways, airports, etc. Figure 1.1 shows the distribution of transportation investments in Canada over different transportation modes averaged from 1993 to 1996 (Transportation Canada, 1998). It can be seen that road investments constitute 87% of the total transportation investments ($9.3 billion) in Canada. This means that the greatest portion of the funding allocated to transportation infrastructure is assigned to highway and highway-bridge projects.

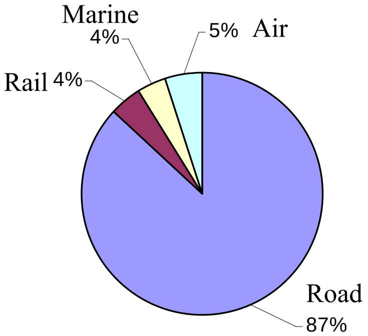


Figure 1.1 Distribution of transportation investments in Canada
(Between 1993-1996) (Source: Transportation Canada, 1998)

Although highways are the dominant element in any road network and their poor condition impairs the overall performance of the network, highway bridges are considered more critical and more vital links in any road network. This is because the full or partial failure of these links paralyses the overall performance of the network and may lead to serious catastrophes.

Research emphasis in recent years has been shifting from the design and construction of new bridges to the inspection and maintenance of existing ones in order to upgrade the existing networks and to keep them in a safe and serviceable condition. These activities demand a tremendous amount of data collection and funds. Therefore, Bridge Management Systems (BMSs) have been developed to provide decision support to improve the quality of inspection and the allocation of the available limited funds.

1.2 PROBLEM DEFINITION

The core part of a BMS is the database which is built up of information obtained from the regular bridge inspection. Inspectors are required to accurately record the defects (e.g., dimensions, locations, and quantities) based on their positions during an inspection plan. However, traditional bridge inspection is paper-based and provides only limited support for spatial visualization. Furthermore, semi-manual data re-entry at the office results in lower efficiency and input errors. In addition, bridge inspection is a knowledge-intensive process, which is becoming increasingly challenging due to the uncertainty issues implicated in the collected data related to condition evaluation. Therefore, the inspection evaluation results may vary depending on the accuracy of inspection data and the level of experience of inspectors.

To summarize, bridge inspection data collection is paper-based and semi-automated, which makes the data collection inaccurate and inefficient. The collected data can directly affect bridge condition evaluation and maintenance decisions. Furthermore, expert experience is prevailing over the condition rating evaluation, which heavily depends on the subjective awareness of the inspector and lacks of the interaction among different factors.

1.3 RESEARCH OBJECTIVES

In order to overcome the above mentioned limitations, efforts in this research are directed to develop a more efficient and accurate bridge inspection system to facilitate on-site data collection and to support inspectors when making decisions for element condition evaluation. Hence, this research's objectives are:

(1) To design a mobile location-based bridge inspection system which integrates various visualization and interaction models to facilitate on site data collection.

(2) To develop a decision-support expert system using Artificial Intelligence (AI) technologies to evaluate condition rating when inspectors make decisions on site. The system should be compatible with bridge data that are available in different transportation agencies and should be updatable whenever new data are available.

1.4 THESIS ORGANIZATION

This study will be presented as follows:

Chapter 2 Literature Review: This chapter presents the major BMSs used in North America and current bridge inspection situation. Then, mobile Location-Based Computing (LBC) technologies used in bridge inspection are introduced. In addition, several AI technologies are compared.

Chapter 3 Mobile Location-based Bridge Inspection: This chapter introduces the detailed design of LBC for visual bridge inspection. Mobile computing is used to re-engineer bridge inspection processes. The usage of tracking methods for finding the inspector's position and inspection targets, and a 3D bridge model with interaction functions for locating defects are discussed.

Chapter 4 Bridge Inspection Decision-Support System: This chapter presents a Decision-Support System (DSS) application for bridge deck deterioration diagnosis. A learning-based Bayesian network (BN) is developed based on previous inspection records to evaluate the current element condition rating from the observed inspection information.

Chapter 5 Implementation and Case Studies: In this chapter, several case studies are used to demonstrate the prototype system using the proposed LBC and DSS approaches.

Chapter 6 Summary, Conclusions, Contributions, and Future work: This chapter summarizes and concludes the present research work, highlights its contributions, and suggests recommendations for future research.

CHAPTER 2 LITERATURE REVIEW

2.1 BACKGROUND

During the bridge construction boom of the 1950's and 1960's, little emphasis was placed on safety inspection and maintenance of bridges. This changed when the 681 m Silver Bridge at Point Pleasant, West Virginia, collapsed into the Ohio River on December 15, 1967, killing 46 people (Czepiel, 1995). This tragic collapse aroused national interest in the safety inspection and maintenance of bridges. The U.S. Congress was prompted to add a section to the "Federal Highway Act of 1968" which required the Secretary of Transportation to establish a national bridge inspection standard. Thus, in 1971, the National Bridge Inspection Standards (NBIS) came into being (FHWA, 2000). The NBIS established national policy regarding inspection procedures, frequency of inspections, qualifications of personnel, inspection reports, and maintenance of the National Bridge Inventory (NBI). Other manuals were subsequently developed by the Federal Highway Administration (FHWA, 2000) and the American Association of State Highway and Transportation Officials (AASHTO, 1993).

2.2 BRIDGE MANAGEMENT SYSTEMS IN NORTH AMERICA

Bridge Management Systems (BMSs) aim to perform the management functionalities related to bridges from the conceptual stage to the end of their useful life, through the design, construction, operation and maintenance stages. The major tasks in bridge management are: (1) collection of inventory data, (2) inspection, (3) assessment of condition and strength, (4) repair, strengthening or replacement, and (5) prioritizing the allocation of funds.

BMSs are means of managing information of bridges to support decision making that assures their long-term health and to formulate maintenance programs in line with budgetary constraints and funding limitations. BMSs include four basic components: data storage, cost and deterioration models, optimization and analysis models, and updating functions (Czepiel, 1995; Ryall, 2001). The core part of a BMS is the database which is built up of information obtained from the regular inspection and maintenance activities. Bridge database management includes the collection, updating, integration, and archiving of the following information: (1) bridge general information (location, name, type, load capacity, etc.), (2) design information and physical properties of the elements, (3) inventory data, (4) regular inspection records, (5) condition and strength assessment reports, (6) repair and maintenance records, and (7) cost records. A simplified BMS structure is demonstrated in Figure 2.1.

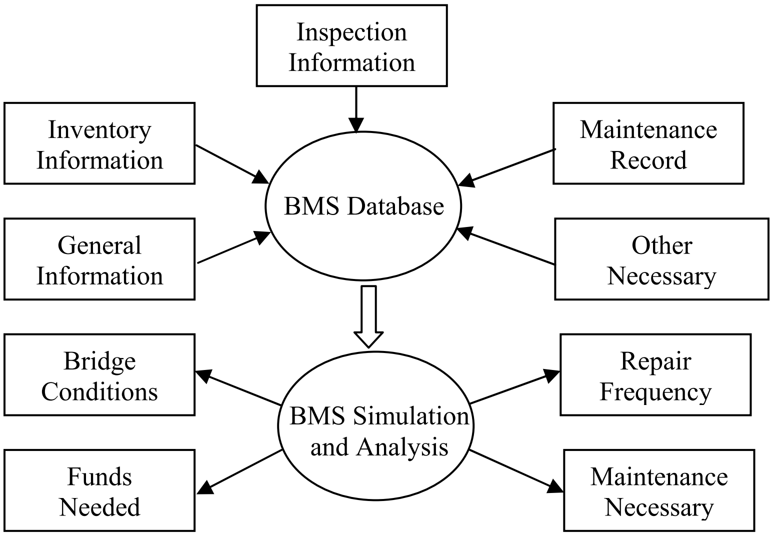
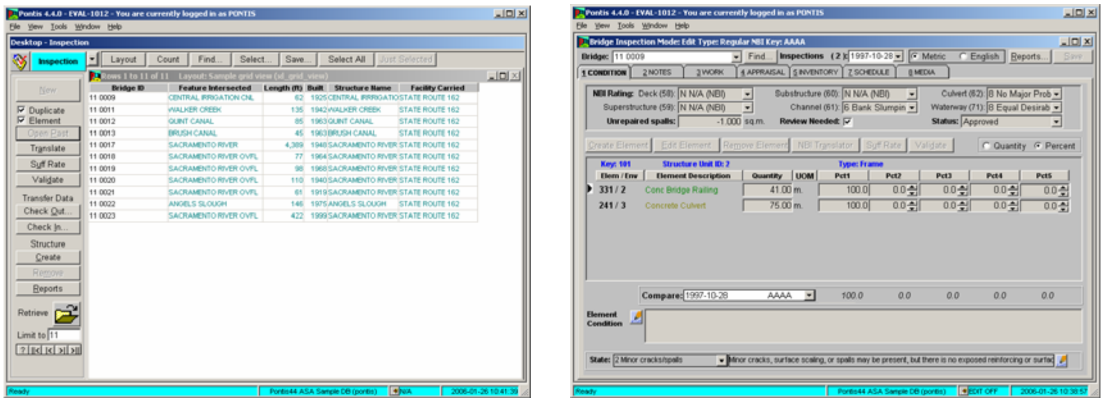


Figure 2.1 General structures of BMSs (Source: Czepiel, 1995)

The two most popular BMSs are Pontis (AASHTO, 1999) and BRIDGIT (NET, 1994). These systems have a generic design that can be adapted to accommodate the individual needs of most transportation agencies. Pontis is the predominant BMS employed in the U.S. Figure 2.2 shows the interface of the bridge inspection model used in Pontis. Its first version was released in 1992 as a result of the cooperation between the FHWA and several state Departments of Transportations (DOTs). In August 1993, Pontis became a product of AASHTO which released several versions. Pontis is a comprehensive BMS that supports the following: (1) collecting bridge inventory and inspection data; (2) formulating network-wide preservation and improving policies for use in evaluating the needs of each bridge; and (3) developing recommendation for an agency's capital plan to derive the maximum benefit from its limited funds. BRIDGIT is a BMS developed by the National Cooperative Highway Research Program (NCHRP) in the U.S. It is very similar to Pontis in terms of its functions and capabilities. The primary difference between them lies in the optimization model. BRIDGIT adopted the bottom-up approach to optimization while Pontis uses the top-down approach. The advantage of the former is that BRIDGIT can perform multi-year analysis and consider delaying actions on a particular bridge to a later date.



☑ Duplicate
☑ Element
Limit to 11

Bridge ID	Feature Intersected	Length (ft)	Built	Structure Name	Facility Carried
11 0009	CENTRAL IRRIGATION CNL	62	1925	CENTRAL IRRIGATIO	STATE ROUTE 162
11 0011	WALKER CREEK	135	1942	WALKER CREEK	STATE ROUTE 162
11 0012	QUINT CANAL	85	1963	QUINT CANAL	STATE ROUTE 162
11 0013	BRUSH CANAL	45	1963	BRUSH CANAL	STATE ROUTE 162
11 0017	SACRAMENTO RIVER	4,389	1948	SACRAMENTO RIVER	STATE ROUTE 162
11 0018	SACRAMENTO RIVER OVFL	77	1964	SACRAMENTO RIVER	STATE ROUTE 162
11 0019	SACRAMENTO RIVER OVFL	98	1968	SACRAMENTO RIVER	STATE ROUTE 162
11 0020	SACRAMENTO RIVER OVFL	110	1940	SACRAMENTO RIVER	STATE ROUTE 162
11 0021	SACRAMENTO RIVER OVFL	61	1919	SACRAMENTO RIVER	STATE ROUTE 162
11 0022	ANGELS SLOUGH	146	1975	ANGELS SLOUGH	STATE ROUTE 162
11 0023	SACRAMENTO RIVER OVFL	422	1999	SACRAMENTO RIVER	STATE ROUTE 162

Bridge: 11 0009
Inspections (2): 1997-10-28
● Metric ○ English

NBI Rating: Deck (58): N N/A (NBI)
Superstructure (59): N N/A (NBI)
Unrepaired spalls: -1.000 sq.m.
Substructure (60): N N/A (NBI)
Channel (61): 6 Bank Slumpin
Review Needed: ☑
Culvert (62): 8 No Major Prob
Waterway (71): 8 Equal Desirab
Status: Approved

○ Quantity ● Percent

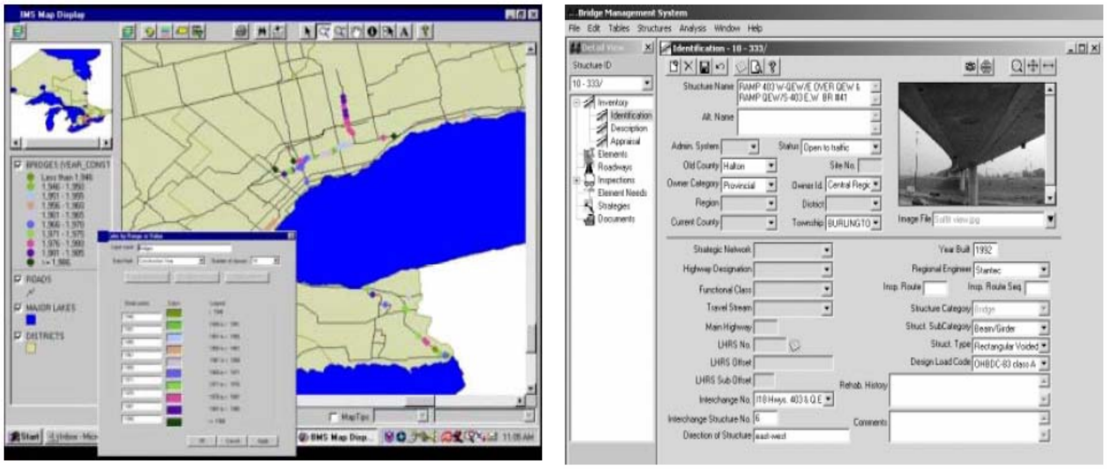
Key: 101 Structure Unit ID: 2 Type: Frame

Elem / Env	Element Description	Quantity	UOM	Pct1	Pct2	Pct3	Pct4	Pct5
331 / 2	Conc Bridge Railing	41.00	m.	100.0	0.0	0.0	0.0	0.0
241 / 3	Concrete Culvert	75.00	m.	100.0	0.0	0.0	0.0	0.0
Compare: 1997-10-28 AAAA				100.0	0.0	0.0	0.0	0.0

Element Condition

State: 2 Minor cracks/spalls Minor cracks, surface scaling, or spalls may be present, but there is no exposed reinforcing or surfac

Figure 2.2 Pontis 4.4 bridge inspection interface (Resource: Pontis, 2005)



Identification - 10 - 333/

Structure ID: 10 - 333/

Structure Name: RAMP 403 W-QEW/E OVER QEW & RAMP QEW/S-403 E.W BR #41

Alt. Name:

Admin. System:

Status: Open to traffic

Old County: Halton

Site No.:

Owner Category: Provincial

Owner Id: Central Regic

Region:

District:

Current County:

Township: BURLINGTO

Image File: Soffit view.jpg

Strategic Network:

Highway Designation:

Functional Class:

Travel Stream:

Main Highway:

LHRS No.:

LHRS Offset:

LHRS Sub Offset:

Interchange No.: 118 Hwys. 403 & Q.E

Interchange Structure No.: 6

Direction of Structure: east-west

Year Built: 1992

Regional Engineer: Stantec

Insp. Route:

Insp. Route Seq.:

Structure Category: Bridge

Struct. SubCategory: Beam/Girder

Struct. Type: Rectangular Voided

Design Load Code: OHBDC-83 class A

Rehab. History:

Comments:

(a) GIS map view interface

(b) Inventory and inspection data interface

Figure 2.3 OBMS interface (Source: Thompson et al., 1998)

In Canada, there is no dominant BMS that is used by all the provincial ministries of transportation and most of them have their own BMSs. The Ministry of Transportation of Ontario (MTO), which is responsible for the management of approximately 3000 bridges on the provincial highway network, initiated the Ontario Bridge Management System (OBMS) in 1998 and updated it in 1999 (Figure 2.3) (Thompson et al., 1998). OBMS consists of the Ontario Structural Inventory System (OSIS), which includes all design, construction, and technical data for Ontario's highway structures (bridges, culverts, retaining walls, etc.) and the Ontario Structure Inspection Management System (OSIMS, 2000), which includes condition ratings for all structural and non-structural elements. OBMS is written in Visual Basic for client-server development. It features a knowledge-based approach to treatment selection and cost estimation. The system includes three major models: element-level model, project-level model, and network-level model.

- *Element-level* model focuses on element-level bridge inspection, which is the primary data source for OBMS.

- *Project-level* model begins with the identification of needs of individual elements, driven by element conditions as determined in recent inspections. Based on element conditions, a knowledge-based model identifies feasible treatment alternatives. Each possible combination of element-level treatments of a bridge is a potential project alternative.
- *Network-level* analysis of OBMS finds the set of project alternatives that maximizes benefit with budget constraints.

Furthermore, OBMS is integrated into Bridge Document Image Management System (BDIMS), which enable users to access related documents for viewing and the Ministry's ArcInfo Geographic Information System (GIS).

In Quebec, the Ministry of Transportation of Quebec (MTQ, 2004) initiated its BMS, which includes 10335 highway structures, in 1996. The system is titled "Système de Gestion des Structure" (SGS). More recently, MTQ is developing a new BMS following the same idea of OBMS and making suitable adjustments to satisfy its own management system. Most of the functions in OBMS are implemented in the new BMS of MTQ, and the system is expected to become ready for testing in 2007. In western Canada, six municipalities developed a BMS for municipal sized inventories (Czepiel, 1995). This system consists of two database modules: static for inventory data, and dynamic for visual inspection data. Analysis routines are also provided to compute sufficient ratings of bridges, establish network-wide maintenance strategy options, and facilitate present value computations and the prediction of least-cost long-term maintenance.

2.3 CURRENT BRIDGE INSPECTION SITUATION

The core part of a BMS is the database which is built up of information obtained from the regular inspection. Among the various tasks of bridge management, field inspection is essential for evaluating the current condition of a bridge. Bridge management departments have come to realize that in order to make sound infrastructure management decisions, they need to base their decisions on predictive models developed from accurate condition data collected in the field. Effective bridge management is thus heavily dependent on field inspectors to collect detailed condition information for all of the individual elements of a bridge and enter these data into a BMS database. Therefore, a BMS must utilize accurate and accessible inspection information to evaluate a bridge's structural conditions.

The type of inspection may vary over the useful life of a bridge to reflect the intensity of inspection required. The five types of inspections listed below allow a bridge manager to establish appropriate inspection levels consistent with the inspection frequency and the type of structure and details (FHWA, 2000).

(1) *Initial Inspection:* An initial inspection is the first inspection of a bridge as it becomes a part of a bridge file. It provides baseline structural conditions and identification of existing problems.

(2) *Routine Inspection:* Routine inspections are regularly scheduled inspections consisting of observations and/or measurements needed to determine the physical and functional conditions of the bridge, to identify any changes from "initial" or

previously recorded conditions, and to ensure that the structure continues to satisfy present service conditions.

(3) *Damage Inspection:* A damage inspection is an unscheduled inspection to assess structural damage resulting from environmental factors or human actions.

(4) *In-depth Inspection:* An in-depth inspection is a close-up, hands-on inspection of one or more members above or below the water level to identify any deficiencies not readily detectable using routine inspection procedures.

(5) *Special Inspection:* A special inspection is an inspection scheduled at the discretion of the bridge manager. It is used to monitor a particular known or suspected deficiency.

A routine inspection is defined by the National Bridge Inspection Standards (NBIS) as a regularly scheduled, intermediate level inspection consisting of sufficient observations and/or measurements to determine the physical and functional condition of the bridge, to identify any developing problems and/or change from inventory or previously recorded conditions and to ensure that the structure continues to satisfy present service requirements (NBIS, 1996). Our present research is based on the routine visual bridge inspection. A bridge/structure routine inspection is a multi-task process that eventuates through three distinct stages and their actions (Mills and Wakefield, 2004; Thomas and Ronald, 2004) (Table 2.1). Aspects of each of the three functions may occur during an inspector's daily work processes. Figure 2.4 represents the work processes in details.

Table 2.1 Current bridge inspection process stages and actions

Stage	Action
Inspection Management	• Periodically outputs scheduled inspections from inventory data to paper list; • Manually sorts paper list of scheduled inspections; • Marks up paper drawings with inspection locations. Stores reports and drawings in folder.
Inspection and Condition Assessment	• Collects on-site data into pre-configured formats by marking up previous paper reports; • Only the information carried to the site is available for review; • Produces a photographic record of the structure.
Reporting, Distribution and Archiving	• Reports the collected data into electronic formats and database; • Paper copies are made and distributed via mail or hand delivered to end users; • Electronic data is archived.

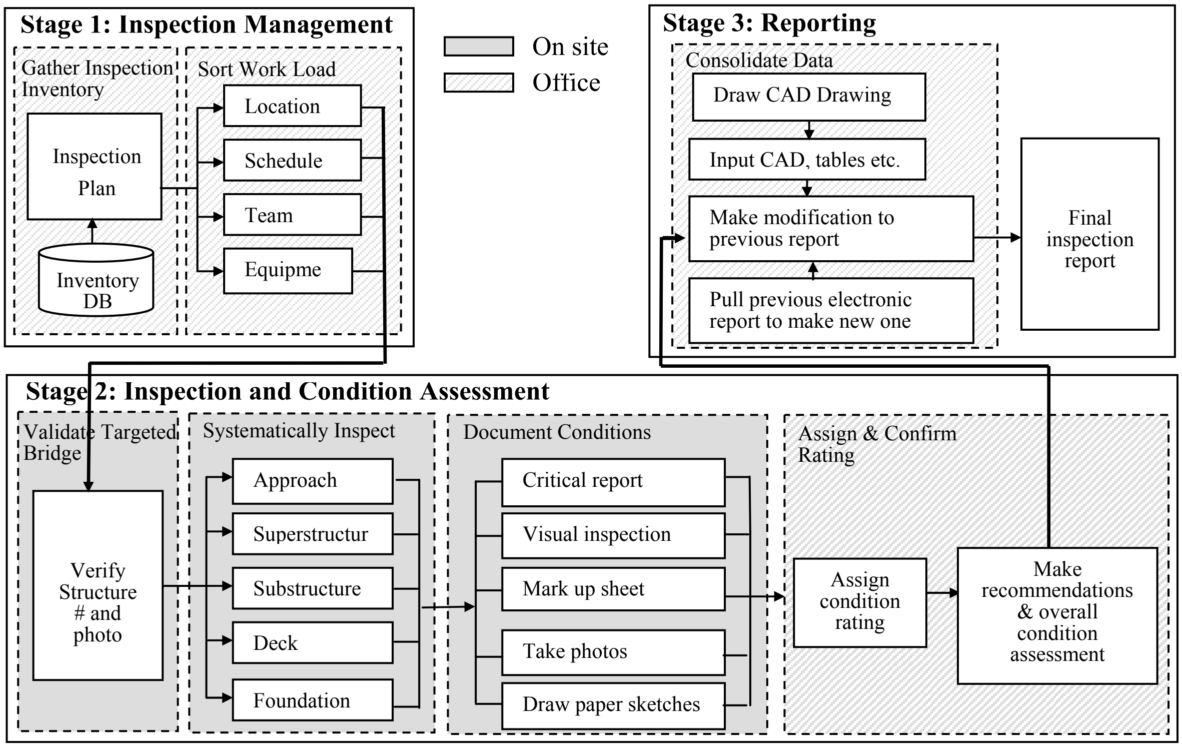


Figure 2.4 Conventional visual routine bridge inspection processes
(Source: Mills and Wakefield, 2004; Thomas and Ronald, 2004)

In stage 1, the bridge inventory data is downloaded from the database for preparing the inspection plan. Then, the inspection process is sorted by the inventory results according to bridge location, inspection schedule, team members, and necessary equipment. Finally, the inspection team takes the printed archive file and the previous reports for all the structures scheduled for field inspection.

In stage 2, the actual on-site inspection and condition assessment occurs in a sequence of four basic tasks:

(1) Validate targeted bridge: upon arriving at the site, the inspection team verifies that the bridge matches the scheduled inspection and the previous inspection report;

(2) Systematically inspect: Bridge component inspection is performed combining visual and non-destructive methods. The inspection proceeds aided by the previous inspection report and predefined categories;

(3) Document conditions: documenting the currently observed new conditions based on previous report (e.g., mark up inspection sheets, take photos, and draw sketches). Any inspection that deems a structure to be in a critical state or need in-depth inspection requires immediate completion of a report;

(4) Assign and confirm rating: each element is quantified and measured to ascertain conditions of deterioration for assigning an overall assessment of the structure.

In stage 3, all the current data from on-site inspection will be consolidated with relevant CAD drawings and previous reports to generate the final report. Report copies are distributed to the end-users and the original report is placed in the archive folder.

2.4 MOBILE LOCATION-BASED COMPUTING TECHNIQUES

As mentioned in Section 2.3 the conventional bridge inspection is paper-based and requires data re-entry at the office, which provides only limited support for spatial visualization. In addition, field workers need to read maps and drawings and to access and update information while moving and climbing on the bridge during inspection. Some problems have been exposed when using the conventional methods which make the data collection process inaccurate and inefficient (Mills and Wakefield, 2004). Therefore, with recent advances in computer technology, computer systems are being employed and used in the field to help inspectors collect inspection information.

The use of computer systems in the field for helping inspectors during inspection can greatly improve the effectiveness of the inspection activities. Firstly, because a computer can store a huge amount of data, the use of computer systems allows inspectors to access information, such as previous inspection data, inspection manuals, pictures, sketches, etc. when and where they need it without carrying this information to the site in paper form. Secondly, the use of computer systems in the field can eliminate the process of reentering the data collected using the paper-based approach. In other words, the data entered in the field, can be integrated into the database immediately without the need for reentering the information. Eliminating the data reentry process also helps to improve the accuracy of the collected information because mistakes can be made during the reentry of the data due to human errors.

In the following subsections, several types of mobile devices with relevant technologies are introduced, available implementations using mobile computing in BMS are discussed, and the location-based concept is provided.

2.4.1 Mobile and wearable computers

Wearable computers have a main unit that includes the CPU, memory, hard disk, etc., and that can be attached to the body of the user. The display unit may be a Head-Mounted Display (HMD) or a flat panel display that is handheld, arm-worn, or attached to the chest of the user. The flat panel display is usually much smaller than those of laptop computers and should be readable in outdoor lighting conditions. HMDs can have high resolution comparable to laptops (full color 800x600 pixels) and can be binocular or monocular. Input devices include touch screens, noise-canceling microphones, pen-based systems and wearable keyboards. Examples of wearable computers are Xybernaut (2005) (Figure 2.5). There are several groups developing new wearable computer systems, such as the WearableGroup at Carnegie Mellon University (2005) and the MIT Wearable Computing Group (2005).

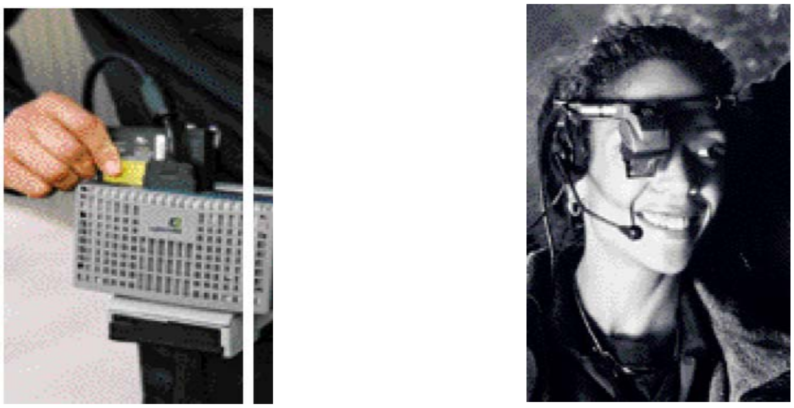
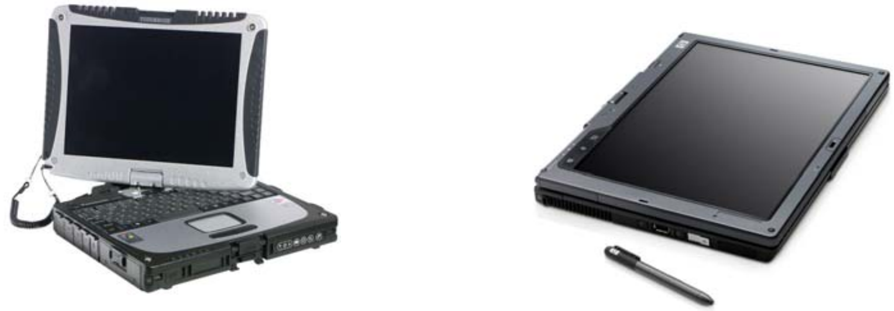


Figure 2.5 Examples of wearable computers: Xybernaut mobile assistant -V
(Source: Xybernaut, 2005)

Pen-based tablet computers have displays that are comparatively large and their operation requires both hands, e.g., the Panasonic Toughbook CF-18 (Panasonic, 2005) and HP Compaq tc4200 (HP, 2005) (Figure 2.6). The advantages of the large display are that they allow for easier handwriting text input and better graphical display. In addition, new software offered at the operation system level provides several new features for interaction such as using the tablet computer digital pen to "gesture", e.g., scratching out digital ink on the display with a digital pen to erase the digital ink.



(a) Panasonic Toughbook CF-18 Tablet PC (Source: Panasonic, 2005)

(b) HP Compaq tc4200 Tablet PC (Source: HP, 2005)

Figure 2.6 Examples of Tablet PC

Personal Digital Assistants (PDAs) are cheaper and lighter than wearable computers and their computing power is limited, e.g., HP iPAQ (HP, 2005). PDAs equipped with a wireless communications card can be used as thin clients for applications running on a remote server. For example, Microsoft Windows Terminal Server allows users to run applications remotely on a thin client PDA (Microsoft, 2005). However, running desktop applications on a thin client in the field is not always a workable solution. Figure 2.7 shows several types of PDAs.

(a) Compaq iPaq PDA
(Source: Compaq, 2005)

(b) Handspring Treo 700
(Source: Handspring, 2005)

(c) Siemens SX66 PDA Phone
(Source: Siemens, 2005)

Figure 2.7 Examples of thin mobile clients

2.4.2 Technologies used in mobile computers for data collection

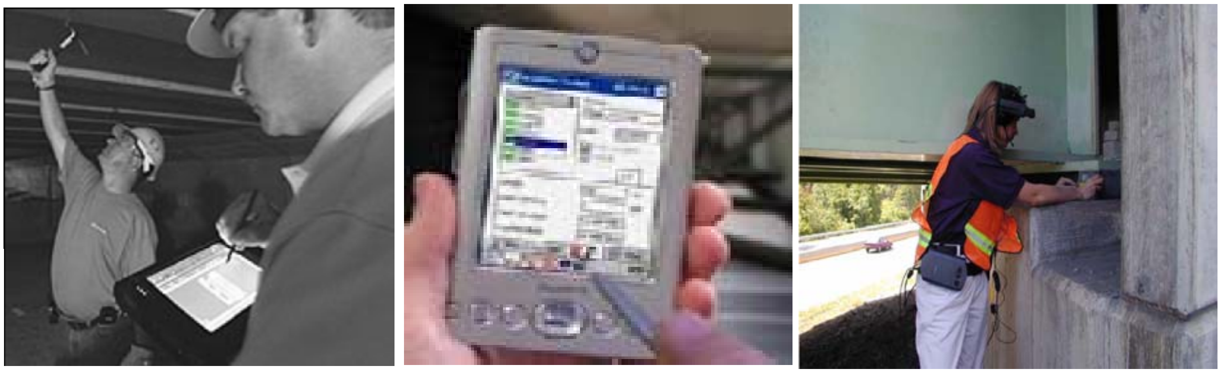
Many technologies are available to support the data collection functionality of the field inspection process. The conventional data collected from these technologies can be used in two forms: raw data about an item, e.g. picture, comment, video record, and identification data used as an index for retrieving more information about a specific item from a database, e.g., barcode (Sunkpho, 2001).

Recently, some advanced technologies that can be used on mobile computers are sketching, handwriting recognition, and speech recognition. Computer sketching is a widely used graphic technique which can arbitrarily mark symbol on pictures, drawings and maps using line, polygon, and digital ink (Rhonda, 2005). Using CAD software (e.g., AutoCAD), sketching can provide a convenient editing tool similar to traditional paper sketching. Meanwhile, after the edition, all the changed contents will be saved in a digital format. With respect to handwriting recognition, the users can write words on the writing pad like they do on paper using electronic stylus; then the system will recognize the handwriting and convert it into text elements. If recognition confidence is low, a dialog box automatically appears that allows text to be verified and corrected. Speech recognition provides a mean for collecting the data and controlling the application for a task that requires the use of both hands (hand-free operation). The user must have about 20 minutes voice recognition training of the system before using it. The user can use dictation and voice commands input. The advantages of speech interfaces over other interaction methods, such as a keyboard, handwriting, or a pointing device, include the naturalness of interaction and the elimination of the use of hands (Sunkpho et al., 2000; Sunkpho, 2002). However, speech-based interaction may not work in noisy environments. In addition, Gordon et al. (2005) discussed a data collection system that delivers some functionalities to support construction inspection tasks by enabling the visualization and manipulation of the data captured using embedded sensing and modeling technologies. Furthermore, research about mobile model-based data collection has been undertaken to simulate real inspection scenarios (Hammad et al., 2004b). However, model-based inspection of bridges has not been studied in depth so far.

2.4.3 Available mobile computing applications in BMS

On-site inspection requires inspectors to be hands-free most of the time because they need to move continuously while taking measurements and notes. For this purpose, research in this field aims to use mobile and wearable computing techniques to increase the efficiency and safety of field workers under severe working and environmental conditions (Beadle et al., 1997). These techniques are expected to lead to cost savings in terms of personnel, inspection time, and data processing.

Recent BMSs are introducing new information technologies to integrate multimedia information and to facilitate mobile data collection and manipulation using pen-based tablet PCs, PDAs, and wearable computers (Figure 2.8). For example, a system developed by the University of Central Florida for the Florida Department of Transportation (Kuo et al., 1994) consists of both a field and office components with a pen-based notebook computer used to collect all field inspection data. Rojas (1995) discussed the concept of re-engineering of the construction inspection process and developed the Field Inspection Reporting System (FIRS) based on pen-based computers, digital cameras, and web-centric process to support inspection. The Digital Hardhat (DHH) is a pen-based computer with special multimedia reporting system software that allows the field worker to save multimedia information, such as text, sound, video and images, into a database. DHH technology enables dispersed inspectors to communicate information and to collaboratively solve problems using shared multimedia data (Stumpf et al., 1998). ScanPrint (Advitam, 2005) operating on pen-based tablet computer is being used in the U.S. by bridge inspection consultants for data collection and bridge management (Figure 2.9). The Massachusetts Highway Department is using a system called IBIIS to store and manage all of their bridge documents (Leung, 1996). As part of this system, inspectors are equipped with a video camcorder to take video and still photographs and a notebook computer to enter the rating data and commentary for each bridge. A more recent, PDA-based field data collection system for bridge inspection is Inspection On Hand (IOH) (Trilon, 2004). IOH helps inspectors capture all rating information, commentary and sketches using hand-held, pen-based PDAs, and share data with Pontis BMS. Garrett et al. (2002) discussed the issues in delivering mobile and wearable computer-aided inspection systems for field users. Sunkpho et al. (2002) developed the Mobile Inspection Assistant (MIA) that runs on a wearable computer and delivers a voice recognition-based user interface.



(a) Pen-based computer (Source: Advitam, 2005)

(b) Handheld computer (Source: Trilon, 2004)

(c) Wearable computer (Source: FHWA, 2000)

Figure 2.8 Example applications of mobile computing in bridge inspection

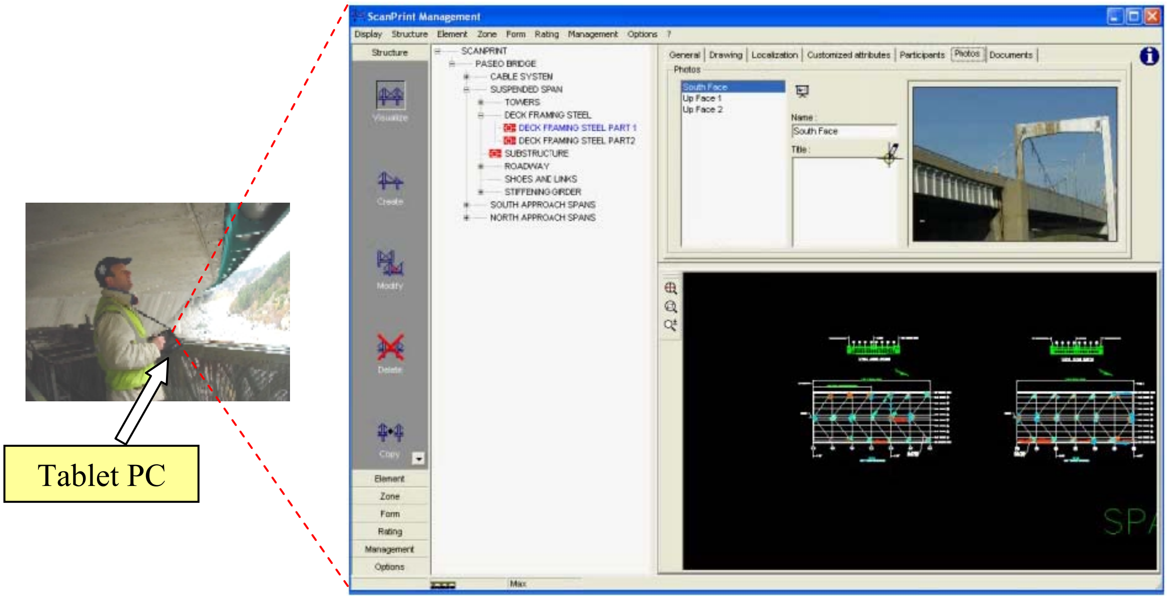


Figure 2.9 Bridge inspection software: Scanprint (Source: Advitam, 2005)

2.4.4 Location-based computing (LBC) for inspection field tasks

LBC is an emerging discipline focusing on integrating geoinformatics, telecommunications, and mobile computing technologies. LBC utilizes geoinformatics technologies, such as Geographic Information Systems (GISs) and the Global Positioning System (GPS) in a distributed real-time mobile computing environment. LBC is paving the way for a large number of location-based services and is expected to become pervasive technology that people will use in daily activities, such as mobile commerce, as well as in critical systems, such as emergency response systems. In LBC, elements and events involved in a specific task are registered according to their locations in a spatial database, and the activities supported by the mobile and wearable computers are aware of these locations using suitable positioning devices. For example, an inspection system based on LBC would allow the bridge inspector to accurately locate the defects on a predefined 3D model of the bridge in real time without the need for any post-processing of the data.

Hammad et al. (2004a) discussed the concept of a mobile data collection system for engineering field tasks based on LBC called *__LBC__ for __Infra__structure field tasks* or LBC-Infra. The concept of LBC-Infra is to integrate spatial databases, mobile computing, tracking technologies and wireless communications in a computer system that allows infrastructure field workers using mobile and wearable computers to interact with georeferenced spatial models of the infrastructure and to automatically retrieve the necessary information in real time based on their location, orientation, and specific task context. Using LBC-Infra, field workers will be able to access and update information related to their tasks in the field with minimum efforts spent on the interaction with the system, which results in increasing their efficiency and reducing the cost of infrastructure inspection. In addition, the wirelessly distributed nature of LBC will allow field workers to share the collected information and communicate with each other and with personnel at a remote site (office). This feature is of a great value especially in emergency cases. Figure 2.10 shows the concept of LBC-Infra (Hammad et al. 2004a). In this figure, a bridge inspector, equipped with a wearable computer, is inspecting a large highway bridge searching for damages, such as cracks. The inspector is equipped with a mobile or wearable computer that has a wireless communications card and is connected to tracking devices. Based on the location and orientation of the inspector and the task to be achieved, the system may display information about the parts of interest within his or her focus or navigation arrows to the locations where cracks are most likely to be found or the locations found in previous inspections. The inspector compares the changes in conditions by wirelessly accessing and viewing any of the previous inspection reports stored in the office database using spatial queries based on his location and orientation. The spatial database of the bridge and the surrounding environment, and the tracking devices attached to the inspector, make it possible to locate structural elements and detected problems and provide navigation guidance to these objects. In addition, all newly collected information is tagged in space. For example, using a pointing device equipped with a laser range finder, the inspector can point at the location of cracks on the bridge. The exact location can be calculated based on the location and direction of the pointer, the distance to the crack, and the registered 3D model of the bridge.

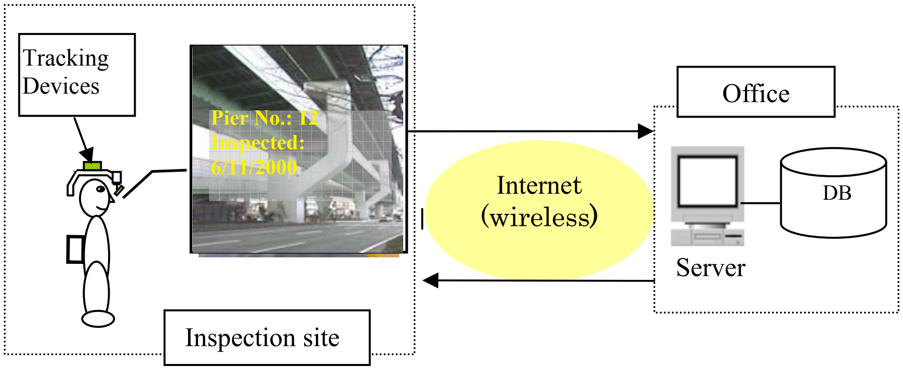
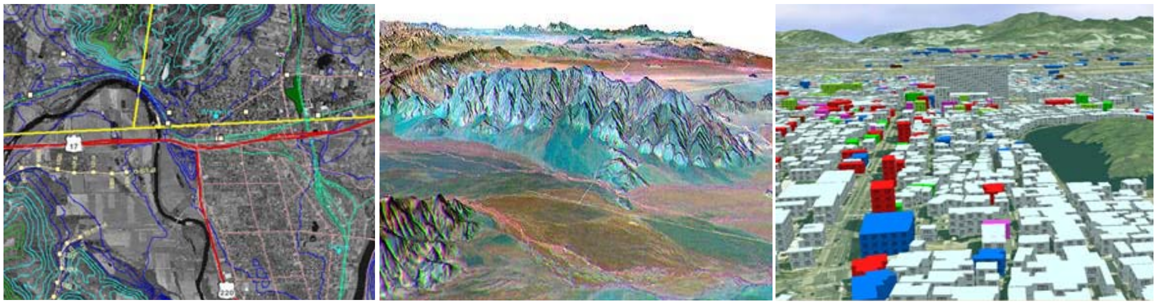


Figure 2.10 Concept of LBC-Infra (Source: Hammad et al. 2004a)

Techniques behind LBC include a combination of GIS, GPS, and wireless communications. These techniques are briefly discussed in the following.

2.4.4.1 Geographic Information Systems (GISs)

Spatial databases are an important component of LBC-Infra where a GIS is used for managing spatial databases, visualization, and spatial modeling and analysis. Current vector GISs can be categorized into one or more of the following: 2D GISs, 2.5D GISs, or 3D GISs (Figure 2.11). 2D GIS databases contain only the X and Y coordinates of the objects stored in them (points, lines, and polygons). When a GIS database contains the Z coordinate as an attribute of the planar points, the GIS is considered to be a 2.5D GIS. Digital Elevation Models (DEMs) are examples of 2.5D GIS models that can be represented using contour lines or a Triangular Irregular Network (TIN). 3D GIS databases contain 3D data structure representing both the geometry and topology of the 3D shapes, and allowing 3D spatial analysis.



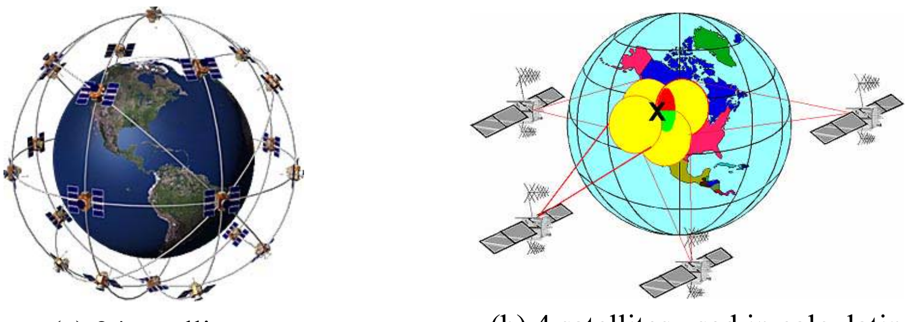
(a) 2D GIS (b) 2.5D GIS (c) 3D GIS

Figure 2.11 Different GIS categories

A spatial database must contain two types of information about the represented objects: geometric data and topological data. Geometric data contain information about the shape of the objects, whereas topological data include the mathematically explicit rules defining the connectivity between spatial objects (Laurini and Thompson, 1992). Through such topological models, GISs can answer spatial queries about infrastructure objects. Researchers from the GIS, computer graphics, and CAD communities have been investigating spatial data structures and models that can be used as the base of 3D GISs for the past several years (ESRI, 2005; ESRI, 2006).

2.4.4.2 Global Positioning System (GPS)

GPS is one positioning technology which is available anywhere within certain conditions and it measures the horizontal and vertical positions of the receiver from the GPS satellites. Because of this availability and the relatively good accuracy and low cost of GPS, it is widely used for mobile mapping and other data collection tasks. The GPS consists of 24 earth-orbiting satellites so that it can guarantee that there are at least 4 of them above the horizon for any point on earth at any time (Figure 2.12). The factors that affect GPS accuracy include ionospheric and tropospheric distortion of the radio signals from the satellites, orbital alignment and clock errors of the satellites, and signal multi-path errors (reflections and bouncing of the signal near buildings). In addition, GPS is easily blocked in urban areas, near hills, or under highway bridges. The accuracy of a position is also a function of the geometry of the GPS constellation visible at that moment in time, i.e., when the visible satellites are well separated in the sky, GPS receivers compute positions more accurately. One method to increase the accuracy of GPS is by using Differential GPS (DGPS). DGPS is based on correcting the effects of the pseudo-range errors caused by the ionosphere, troposphere, and satellite orbital and clock errors by placing a GPS receiver at a precisely known location. The pseudo-range errors are considered common to all GPS receivers within some range. Multi-path errors and receiver noise differ from one GPS receiver to another and cannot be removed using differential corrections. DGPS has a typical 3D accuracy of better than 3 m and an update rate of 0.1-1 Hz. The DGPS corrections can be sent to the mobile GPS receivers in real time, or added later by post-processing of the collected data. Real-time kinematic GPS receivers with carrier-phase ambiguity resolution can achieve accuracies better than 10 cm, but are computationally very intensive to run in real time (Kaplan, 1996).



(a) 24 satellites

(b) 4 satellites used in calculating the location of the receiver

Figure 2.12 Satellites used for GPS

2.5 DECISION-SUPPORT SYSTEM (DSS) FOR BRIDGE CONDITION EVALUATION

The use of computer systems can make knowledge-based assistance in the field possible. For example, when a bridge component is found to be cracked, the crack found may be considered critical, i.e., it can lead to catastrophic failure of the entire bridge. However, the evaluation of whether a crack is critical is sometimes a complex process, e.g., different inspection data from different sources are required for analysis. Therefore, it can be difficult to do such complex analysis at the inspection site manually. A computer system could possibly be used as a DSS to help the inspector in making complex decision so that the appropriate action can be performed more efficiently and effectively.

The reported condition of bridge element or component is an evaluation of its current physical state compared to what it was on the day it was built. Accurate assignment of condition ratings is dependent upon the bridge inspector’s ability to identify the bridge components and their elements. Bridge components are the major parts of a bridge including the deck, superstructure, substructure, and foundations. Bridge elements are individual members comprised of basic shapes and materials connected together to form bridge components. The overall condition rating of bridge components is directly related to the physical deficiencies of bridge elements. In order to evaluate bridge deterioration, a qualified inspector must have professional training and possess sufficient experience. As the trained engineers retire, a significant experience gap is created. Therefore, it is of great interest to develop DSSs for bridge inspection.

2.5.1 Condition rating definition

During an inspection, an attempt is made to determine the condition of an element based on the subjective opinion of qualified experts following inspection manuals. It is usually in the form of a number. No calculations are performed, but the numbers are later used in conjunction with other factors to calculate the health index (Pontis, 2005; Thompson and Shepard, 2000) which is then further used to decide on the maintenance strategy. The definition of condition rating is different from one agency to another. This is because the factors that influence condition deterioration rely on location, climate, traffic condition, etc. For instance, Table 2.2 shows an example about the condition rating definition of painted steel element used in Oregon Bridge Inspection Manual (ODT, 2003), and Table 2.3 describes element Performance Condition Rating (PCR) used by the MTQ (MTQ, 2004). Obviously, the values of condition rating involve the judgment of inspectors.

Table 2.2 Painted steel element condition rating definition (Source: ODT, 2003)

Condition Rating	Condition State Descriptions
1	There is no evidence of corrosion and any paint systems are sound and functioning as intended to protect the metal surface.
2	There is little or no active corrosion. Surface or freckled rust may have formed or is forming. The paint system may be chalking, peeling, curling, or showing other early evidence of paint system distress, but there is no exposure of metal.
3	Surface or freckled rust is moderate to heavy. There may be exposed metal, but there is no measurable loss of section.
4	The paint system has failed. Corrosion may be present but any section loss due to active corrosion does not yet warrant a structural analysis of the element or the bridge.
5	Corrosion has caused section loss and is sufficient to warrant structural analysis to ascertain the impact on the ultimate strength and/or serviceability of either the element or the bridge.

Table 2.3 Performance condition rating (PCR) of elements (Source: MTQ, 2004)

PCR		Percentage of decrease of the ability of an element		
Code	Description	Primary	Secondary	Auxiliary
6	Excellent	0-1%	0-2%	0-5%
5	Good	1-5%	2-10%	5-20%
4	Acceptable	5-10%	10-20%	20-40%
3	Inferior	10-15%	20-30%	40-60%
2	Weak	15-20%	30-40%	60-80%
1	Critical	>20%	>40%	>80%

2.5.2 Methodologies used for DSSs in BMS

Stochastic models have been used by experienced transportation agencies that have a relatively large bridge population with a recorded history of bridge conditions. These techniques have treated the two basic problems of deterministic models by capturing the uncertainty of the deterioration process and considering the current bridge condition in predicting the future condition. However, the drawback of using probability distribution functions is that they have not been tested practically. Although Markovian models have been employed by many advanced BMSs, such as Pontis and BRIDGIT, and have achieved great advances in modeling bridge deterioration, they have some assumptions and limitations that can be summarized as follows:

(1) Markovian models assume the presence of state independence that means past conditions have no effect on the predicted ones (Madanat et al., 1995).

(2) Markovian models assume discrete transition time intervals, constant bridge population, and stationary transition probabilities.

(3) Markovian models are incapable of considering the effect of major maintenance treatments on the deterioration process.

(4) It is quite difficult to consider the interactive effects among deterioration mechanisms of different bridge components.

(5) Transition probabilities are estimated based on subjective engineering judgment and require frequent update when new data are obtained.

Recently, several Artificial Intelligence (AI) models have been proposed to support inspectors during on-site bridge inspection, such as rule-based Expert Systems (ESs), Artificial Neural Networks (ANNs), Case-Based Reasoning (CBR), and Bayesian Networks (BNs). ESs (Friedman-Hill, 2003) use a relatively simple model written in the form of *if-then* rules that can be adapted to various problems (e.g., fuzzy rules for uncertainty problems) without having large amount of data. However, rule-based ESs are difficult to update their rule bases when new problems are encountered; and they require knowledge acquisition from domain experts, which is an inconvenient and time-consuming process. In addition, the output information is generally fixed. ANNs (Arditi and Tokdemir, 1999) emulate the learning capabilities of the human brains and they have been recognized as a powerful tool in solving engineering problems. Pattern recognition, classification, diagnosis, estimation, optimization, and prediction are the most successful applications. The performance of an ANN depends mainly on training cases; this performance is evaluated by using a set of unseen cases. However, ANNs have some limitations in terms of their behavior and applicability; they deal only with problems whose inputs and outputs are numeric or converted symbolic attributes, this may lead to losing the information contained in the original representation; they cannot handle efficiently problems that have incomplete data or variable data structure; and they have to be re-trained and re-tested when new data are available, which is a time-consuming task. CBR provides a technique for utilizing past experiences and existing corporate resources to guide diagnosis and fault finding when solving similar ones (Rivard et al. 1998; Morcous, 2000). It relies on experience with similar problems that were encountered before. CBR systems also use a rule library to match the query case and find solutions. The advantages of CBR are its ability to create adapted solutions and automatically case library updating. The limitation of this approach is that the retrieval time increases linearly with the number of cases. Therefore this approach is more effective when the case base is relatively small. BNs (Jensen, 1996) are relatively new AI approach that overcomes most of the drawbacks of ANNs, CBR, and ESs by reusing the specific knowledge encapsulated in the previously experienced problems and is more flexible to represent probabilistic rules and uncertainty.

Research for developing DSS using BNs has been proposed in different areas. McCabe et al. (1998) introduced BNs in a probabilistic ES to evaluate the performance measures, determine the most likely causes of poor performance, and recommend changes to the simulation model. Sloth et al. (2004) developed a software named BridgeAnalyzer to analyze bridge defects' causes based on BNs. The software poses questions in a specific sequence in order to find the most probable cause for a given observation of a defect. Attoh-Okine and Bowers (2004) investigated the use of BNs for modeling bridge deterioration based on expert knowledge.

2.5.3 Theoretic background of Artificial Intelligence (AI) methods

Four AI methods are introduced in the following subsections, which are ANNs, CBR, Rule-based ESs, and BNs. Because ANNs and CBR are not used in this research, only the basic concepts are described and the detailed theories can be found in Arditi and Tokdemir (1999) and Rivard et al. (1998), respectively. With respect to rule-based ESs and BNs, both of them are used for evaluating bridge elements condition rating as will be explained in Chapters 4 and 5. Therefore, the theories of both rule-based ESs and BNs are discussed in detail.

2.5.3.1 Artificial Neural Networks (ANNs)

ANNs are information processing paradigm that is inspired by the way biological nervous systems, such as the brain. An ANN is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. The ANN performs pattern recognition or data classification through a learning process, which consists of three layers of nodes: input-nodes, hidden layers, and output-nodes (Arditi and Tokdemir, 1999) (Figure 2.13). The weights and the thresholds achieved by training determine the behavior of the network.

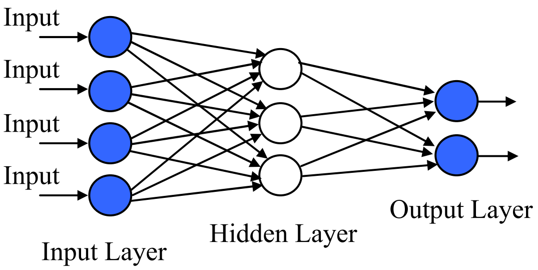


Figure 2.13 Layers structure of ANNs (Source: Arditi and Tokdemir, 1999)

2.5.3.2 Case-Based Reasoning (CBR)

In CBR, problem solving is based in a collection of past cases rather than being encoded in generic rules or other knowledge descriptions. Each case typically contains a description of a problem and a solution. In order to solve a current problem it is matched against the cases in the case base, and similar cases are retrieved. The retrieved cases are used to suggest a solution which is reused and tested for success. The solution may be revised if necessary. The current problem and the final solution are stored as part of a new case. CBR systems, in general, have four main aspects: case representation, case accumulation, case retrieval, and case adaptation. The process of each main aspect is shown in Figure 2.14 (Morcous, 2000).

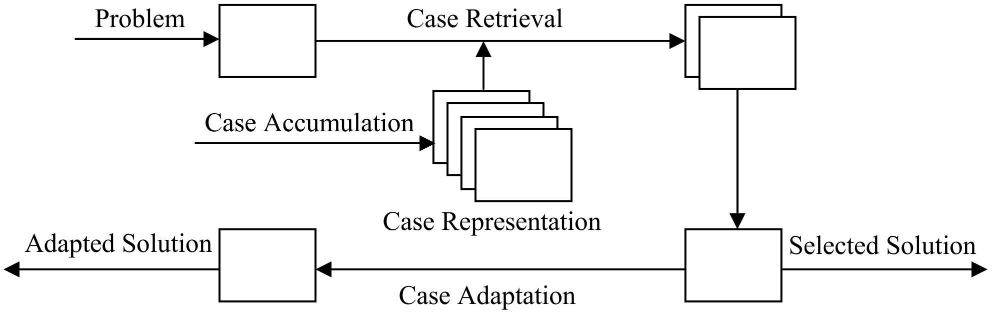


Figure 2.14 CBR process and its main aspects (Source: Morcous, 2000)

2.5.3.3 Rule-based Expert Systems (ESs)

A rule-based ES is based on developing rules. These rules reflect essential relationships within the domain. When specific information about the domain comes in, the rules are used to draw conclusions and to point out appropriate actions. This inference takes place as a kind of chain reaction. The architecture of a rule-based ES contains a knowledge base and an inference engine. The basic units represented in rule-based ES are the rules.

Rules can be fired through matching the asserted facts from the knowledge base using the inference engine (Friedman-Hill, 2003) (Figure 2.15).

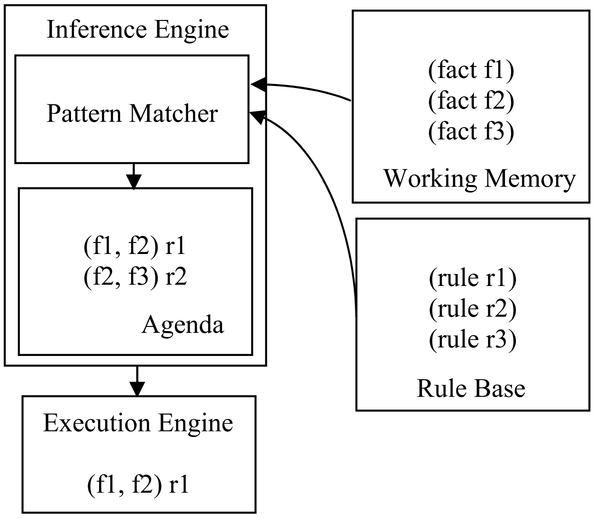


Figure 2.15 Architecture of rule-based ESs (Source: Friedman-Hill, 2003)

Two types of inferences are common in rule-based ES: forward and backward chaining. Forward chaining starts from the known facts to trigger all the rules whose premises are satisfied. Then all the results from the fired rules are added to the known facts. The process is repeated until no new facts are added. The control flow of a backward-chaining system is more complex than that of a forward-chaining system. Backward-chaining systems try to satisfy the goals in the goal stack. They do this by finding rules that can conclude the information needed by the goal, and trying to make the *if* parts of those rules satisfied. The system terminates with success when the goal stack is empty. It terminates with failure if the system runs out of rules to try. The backward chaining inference engine can be embedded in forward chaining framework to solve a problem. In addition, fuzzy logic (fuzzy rules) can also be used in rule-based ESs to describe linguistic information that are imprecise and ambiguous (e.g., the defect level can be described as severe, moderate or minor). An effort to realize rule-based ESs used in bridge inspection systems is to write the rules based on inspection manuals and expert knowledge.

2.5.3.4 Bayes' rule and Bayesian Networks (BNs) learning

A BN consists of a set of nodes and a set of directed edges between the nodes (Figure 2.16). Edges reflect cause-effect relations within the domain. These effects are normally not completely deterministic. The strength of an effect is modeled as a probability (Jensen, 1996).

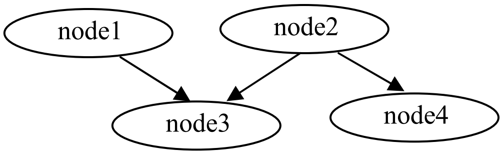


Figure 2.16 Structure of BNs (Source: Jensen, 1996)

BNs provide a method to represent relationships between variables even if the relationships involve uncertainty, unpredictability or imprecision. BNs are directed acyclic graphical models combined with probabilities that follow the rules of the probability theory. The probability theory establishes a set of cause-effect relationships where the nodes are connected by directional arcs, ensuring that the system as a whole is consistent, and providing ways to interface models to data. The nodes of the network represent random variables; the states of the node represent the values taken by a variable; and the relationships between nodes represent probabilistic dependencies between variables. These dependencies are quantified through a set of Conditional Probability Tables (CPTs). Each variable is assigned a CPT of the variables acting as its parents. For variables without parents, this is an unconditional distribution. The basic concept in BNs is using Bayes' rule for conditional probabilistic inference (Jensen, 1996). Equation 2.1 gives a basic description of the Bayes' rule. If $B_1, B_2, \cdots B_j \cdots, B_k$ are the possible scenarios with an effect on the event A, then:

$$P(B_j \mid A) = \frac{P(B_j \cap A)}{P(A)} = \frac{P(A \mid B_j) \cdot P(B_j)}{\sum_{i=1}^{k} P(A \mid B_i) \cdot P(B_i)} \qquad \text{Equation (2.1)}$$

Where $P\ (B_j \mid A)$ is the conditional probability of B_j given that all we know is A. In the present paper, a BN is used to create a DSS for defect cause-effect probabilistic analysis in bridge inspection. The deduction procedure of Bayes' rule and a calculation example can be found in Appendix A.

A BN may be created by an expert, learned automatically from data files, or developed by a combination of the above two methods. The network captures knowledge in a modular form using chance variables that can be transmitted from one situation to another and allow a clear visualization of the relationships involved. By identifying the decision variables to be controlled and/or the utility variables to be optimized, an influence diagram (also known as a decision network) can be formed. Furthermore, BNs combining decision and utility variables can be used in BMSs to find optimal decisions of maintenance activities and to control the allocation of limited funding.

Eliciting BNs from experts can be a laborious and expensive process. Thus, in recent years, there has been a growing interest in learning-based on available data. BNs allow conditional probabilities to be defined and learned from a collection of cases. If the collection of cases is a sample from the population, then we can use the frequency information included in these cases as an approximation of the unknown probabilities. The learned conditional probabilities can be used in the network to make predictions for new cases. BNs support two kinds of learning: structural and parametric learning. Structural learning is the process where the system learns the dependencies between the variables which lie in the data. Parametric learning is the process where the system fills in the parameters describing the strength of the dependencies in the learned structure. In this research, parametric learning using the Expectation-Maximization (EM) algorithm from incomplete data is used for the calculation of the probabilities of the network.

The EM algorithm (Bilmes, 1998; Russell and Norvig, 2003) is a broadly applicable approach to the iterative computation of maximum likelihood estimates used in a variety of incomplete-data problems. In each iteration there are two steps, the Expectation step (E-step) and the Maximization step (M-step). In the E-step, the EM algorithm performs inference in the network for each of the datum in the database. This allows the information from the data to be used, and various necessary statistics to be calculated from the resulting posterior probabilities. Then in the M-step, parameters are chosen to maximize the log posterior given these statistics are fixed. The statistics collected in the E-step are no longer accurate for the new set of parameters. Hence, the E-step must be repeated, then the M-step and so on. At each stage, the EM algorithm guarantees that the posterior probability must increase. Therefore, the posterior probability eventually converges to a local maximum of the log posterior. The EM algorithm is discussed in detail in Appendix B.

2.5.4 Comparison of AI methods

The AI technologies mentioned in Section 2.5.2 can be categorized into two kinds of knowledge representation systems.

(1) Sub-symbolic knowledge representation encodes knowledge in patterns of data structures or use probabilistic information. Sub-symbolic knowledge representation is typically used in uncertain information tasks where the meaning is uncertain or probabilistic, such as ANNs and BNs.

(2) Symbolic knowledge representation assigns understandable meaning to any portion of the data structures used to represent knowledge, such as rule-based ESs and CBR.

The comparisons of AI menthods mentioned in Section 2.5.2 are given below to illustrate their different features and the summary is shown in Table 2.4.

(1) BNs and ANNs

The difference between the two types of networks is that the hidden layer of ANN does not in itself have an interpretation of the domain of the system, whereas all the nodes of a BN represent concepts that are well defined with respect to the domain. Therefore, BNs require detailed knowledge of the domain in question. Probabilities of BNs can be assessed using a combination of theoretical insight, empirical studies, training and various more or less subjective estimates. ANNs can only be learned by training, so it is unable to utilize the knowledge that might have in advance. In addition, ANNs have an even less flexible input-output structure in that an entirely new network is required if any variables are added and deleted because the additional data requires retraining of the network.

(2) BNs and CBR

BNs have the capability to model the relationships between items of interests, both in a quantitative and a qualitative way, and the utilization of knowledge stored in the form of cases to modify the structure and strength of these dependencies according to user's input. In recent years, there has been a significant effort to develop methods and algorithms for inducing BNs directly from data. CBR is a problem-solving paradigm that is able to utilize the specific knowledge of previously experienced, concrete cases. Basically it solves a new problem by remembering a previous similar situation and by reusing information and knowledge of that situation.

(3) BNs and rule-based ESs

BNs have the ability to adjust variables to be input or output at any point without redesigning the system and affecting the rest of the network. However, rule-based ESs require knowledge acquisition from domain expert and permit evidence to be entered only at specific points. The output information is generally fixed. ESs fail to give any solution to a problem whenever no rule is available, and it is difficult to update the rule library when new problems are encountered. An essential difference between rule-based ESs and systems based on BNs is that rule-based ESs try to model the experts' way of reasoning, while with BNs try to model dependencies in the domain itself.

Table 2.4 Comparison of different AI methods

	AI knowledge representation systems			
	Sub-Symbolic		Symbolic	
	ANNs	CBR	Rule-based ESs	BNs
Uncertain/probabilistic	●	○	○	●
Learning/training	●	●	N/A	●
Rules library	N/A	●	●	N/A
Incomplete data	N/A	●	N/A	●
Expert opinion	○	○	●	○

● Major function ○ Minor function

2.6 SUMMARY AND CONCLUSIONS

In this chapter, literature about the current situation and future trends of BMSs was reviewed to find the possibility for new information technology integration. Although mobile devices have been used for bridge inspection activities, the practical applications are still based on 2D and rarely consider the interactions with a 3D model. The location of inspection targets and defects cannot be effectively addressed as is in a real inspection scenario. Moreover, the AI methods used for inspection support are still at the early stage of development. Therefore, based on the literature review, two main proposed approaches, LBC for data collection and DSS for condition evaluation, are explored as our potential research objectives as explained in Chapter 1.

The detailed approaches of mobile LBC integrating several interactive techniques with a 3D bridge model and the DSS using rule-based ESs and BNs will be discussed in Chapters 3 and 4, respectively.

CHAPTER 3 MOBILE LOCATION-BASED BRIDGE INSPECTION

3.1 INTRODUCTION

Mobility is a basic characteristic of field tasks. In a bridge inspection scenario, the inspector has to move most of the time in order to do the job at hand. The inspector walks over, under or around the bridge or in some cases climbs the bridge. Knowing the exact location of the inspector with respect to the inspected elements can greatly facilitate the task of data collection by automatically identifying the elements, and potentially specifying the locations of defects on the elements. Approximate location information can still help in focusing the scope of data collection by identifying the objects that are of interest. Present methods of capturing location information using paper or digital maps, pictures, drawings and textual description can lead to ambiguity and errors in interpreting the collected data.

LBC is an emerging discipline focused on integrating geoinformatics, telecommunications, and mobile computing technologies (Beadle et al., 1997; Karimi and Hammad, 2004). Using LBC combined with GIS and a 3D model, bridge elements which are registered according to their positions in a spatial database can be located using suitable tracking devices, and defects on specific elements can be recorded more efficiently and accurately. Hammad et al. (2004a; 2004b) discussed the concept and requirements of a mobile data collection system for engineering field tasks called *LBC for Infrastructure field tasks* (LBC-Infra) and identified its system architecture based on available technologies and the modes of interaction. The present research further develops LBC-Infra by building on the experience gained from this system.

In this chapter, a new approach integrating a BMS with GIS and 3D detailed models of bridges is discussed. The integrated inspection system based on this approach navigates the inspector to the locations he/she needs to inspect, displays information about the results of previous inspections as augmentations of the 3D model, and identifies the inspection sequence and the defect locations with navigation guidance. Furthermore, the new approach allows the inspector to add defect inspection information, including the location of defects, simply by picking an element in the model at the location where the defect has been found and then by selecting the type and level of the defect using the user interface of the mobile system.

The proposed approach for mobile location-based bridge inspection systems requires re-engineering the inspection process, integrating different software and hardware components, and using the location of the inspector to facilitate inspection tasks. These aspects of the proposed approach are discussed in the following sections.

3.2 RE-ENGINEERING INSPECTION PROCESS THROUGH LBC TECHNOLOGY

As discussed in Chapter 2, a bridge inspection is a multi-task process that normally includes three stages (Thomas and Ronald, 2004): (1) inspection management, (2) inspection and condition assessment, and (3) report generation. In stage 1, the inspection plan, including data gathering about bridge locations, schedule, inspection team, and equipment, is made based on inventory data before starting the field inspection. In stage 2, the actual on-site inspection and condition assessment occurs following the sequence of validating targeted bridges, visual inspection, and condition evaluation. In stage 3, all the current data from on-site inspection are consolidated with relevant CAD drawings to generate a final report. However, conventional bridge inspection relies on marking up paper records in the field and then returning to the office to create semi-manual reports. The disadvantages of using this method for data collection are low efficiency and the inaccuracy of data input. To compensate for such disadvantages, mobile and LBC technologies can assist in automating the process and can eventually become an integral part of the entire inspection process to enhance bridge inspection (Figure 3.1). In Figure 3.1, the preparations of inspection management at stage 1 are accomplished with the help of computers. At stage 2, on-site inspection and condition assessment are processed based on LBC technologies. Finally, inspection results can be automatically saved into a database and the corresponding reports can be generated upon request. The potential advantages of using mobile location-based technology are: (1) Facilitating acquisition, transfer, and querying of data; (2) Improving the efficiency and accuracy during bridge inspection by considering location information based on integrating GIS, GPS and 3D models to help inspectors in updating inspection data; and (3) Supporting communication and collaboration between inspectors at remote field locations.

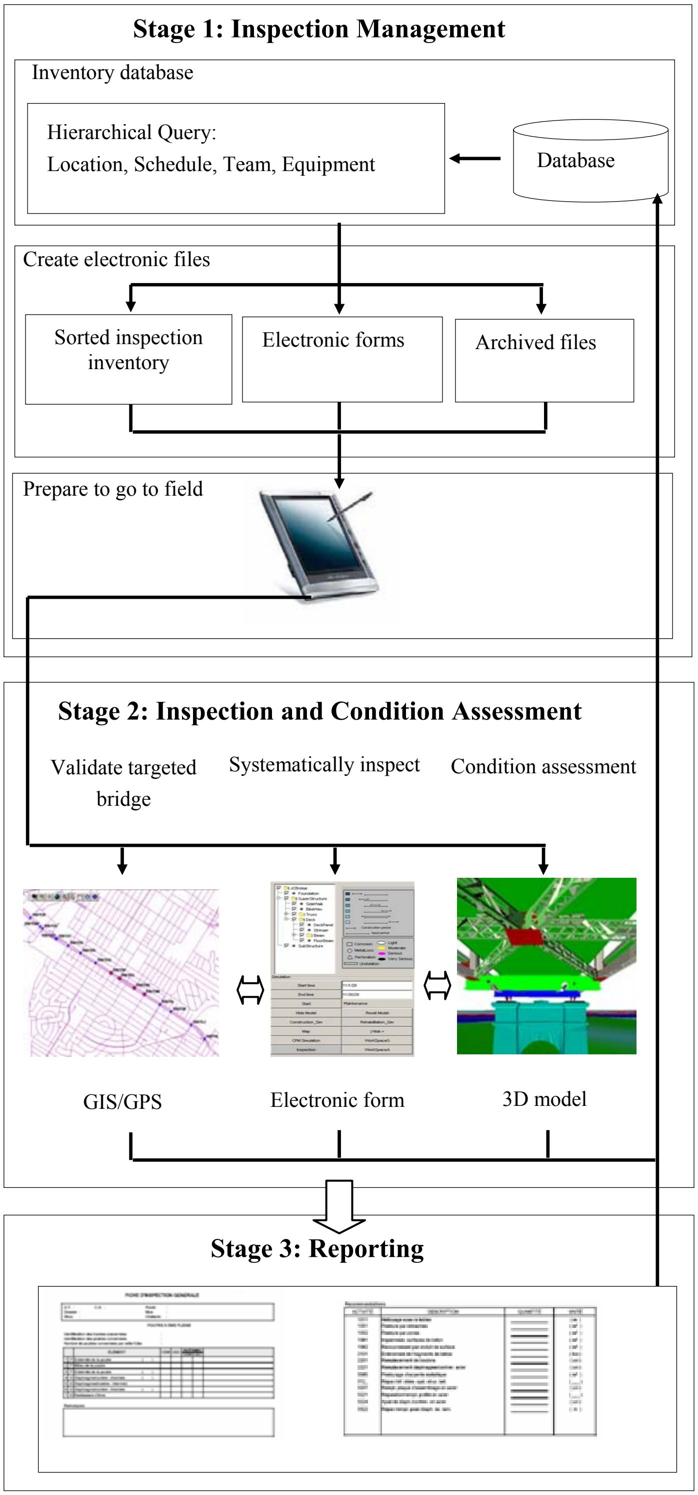
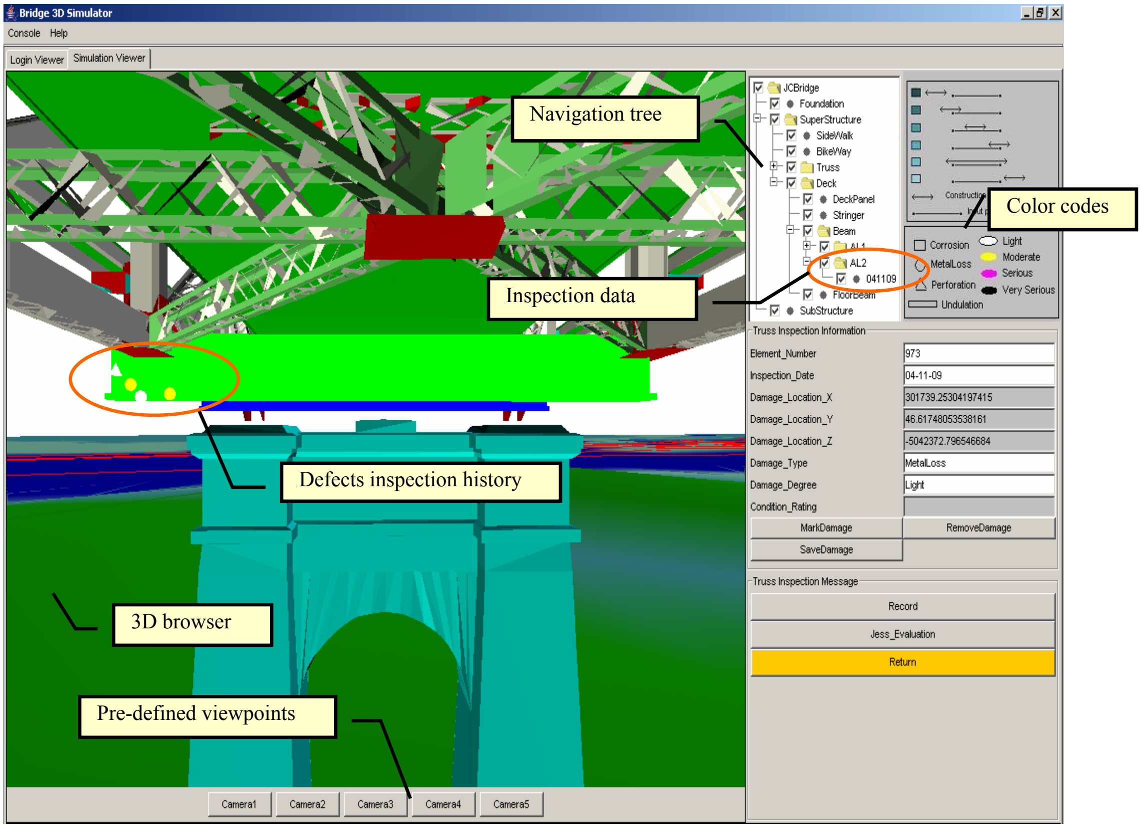


Figure 3.1 Visual bridge inspection re-engineering process

3.3 GENERAL USER INTERFACE DESIGN

One of the major steps in realizing the re-engineering process of bridge inspection is to adequately consider the human-computer interaction patterns in stage 2 above (on-site inspection and condition assessment). For that purpose, a well designed Graphical User Interface (GUI) is essential to satisfy the mobile location-based data collection. A mobile bridge inspection GUI should facilitate the interaction with 2D GIS and 3D virtual environments in real time and provide consistent feedback to the user. The design of the GUI should consider the logic of the inspection process and meet relevant inspection guidelines as well. In addition, this design needs to improve the efficiency of data entry, reduce input errors, and provide automatic access to information that can support inspectors. A well designed GUI will effectively guarantee the reliability of data collection thereby ultimately improving the bridge structure evaluation and maintenance decisions.

In order to show the 3D model on the small screen of a tablet PC used in a mobile situation, a simple but efficient user interface should be designed focusing more on navigation and interaction functions. The main area of the interface should be used to show the 3D browser. Navigation is facilitated by adding a structure tree of the bridge to guide the user to find certain elements in the 3D model. The data stored in the database about the structure of the bridge can be used to automatically generate the logical tree. Through querying the database, the root of the tree is found and the root node is created. Then, queries are applied recursively to find other element nodes based on the data stored in the table. Each tree node has a check box, which facilitates showing or not showing that element in the 3D model. In order to help the user move within the 3D environment without losing his/her way, different viewpoints, such as the side view and top view of the bridge, are created to navigate the user to some specific locations. In addition, a clear symbol and color coding is needed to explain the different types of the defects (e.g., *corrosion, metaloss, perforation,* and *undulation*) and their deterioration levels (e.g., *light, moderate, serious,* and *very serious*), respectively. Using these symbols and colors, the defects historic records (e.g., Figure 5.2 (c)) can be automatically retrieved from the database when selecting a specific member and inspection year from the navigation tree and be displayed on the 3D model. The design of the interface is shown in Figure 3.2.



Truss Inspection Information

Element_Number	973
Inspection_Date	04-11-09
Damage_Location_X	301739.25304197415
Damage_Location_Y	46.61748053538161
Damage_Location_Z	-5042372.796546684
Damage_Type	MetalLoss
Damage_Degree	Light
Condition_Rating	

Truss Inspection Message

Figure 3.2 Design of the user interface of the prototype system

3.4 GENERAL STRUCTURE OF MOBILE LOCATION-BASED BRIDGE INSPECTION SYSTEM

The application of mobile computing techniques can facilitate the inspector's activities by allowing him/her to concentrate on the details of inspection tasks. Figure 3.3 shows the proposed general structure of components and techniques used in mobile location-based bridge inspection systems. In these systems, there are two types of components, which are device components and software components. The device components include portable PCs, PDAs, wearable computers, GPS receivers, HMDs, digital cameras, and wireless communications. Figure 3.4 shows an example of bridge inspection using mobile devices integrated in our proposed prototype system (more details are given in Chapter 5). The devices in this system include a tablet PC with an electronic stylus, an HMD, a Real-Time Kinematic (RTK) GPS receiver, and a digital camera. The software components support three stages: *process control*, *data collection*, and *data analysis*. Each stage interacts with various functions. The *Process control* navigates the inspector to locations of inspection targets following available requirements. *Data collection* is based on 2D and 3D models of bridges with different Levels of Details (LoDs). *Data analysis* aims to rate the conditions, diagnose the causes of defects, and make suitable decisions related to maintenance activities.

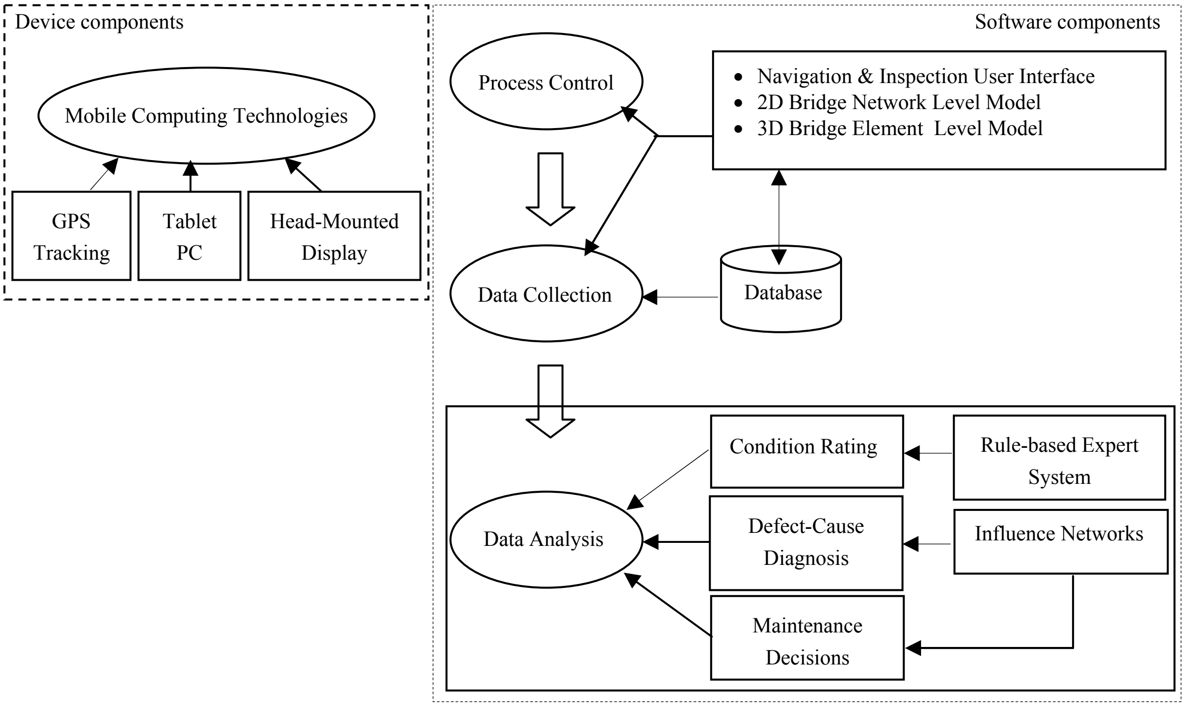


Figure 3.3 General structure of mobile location-based bridge inspection system

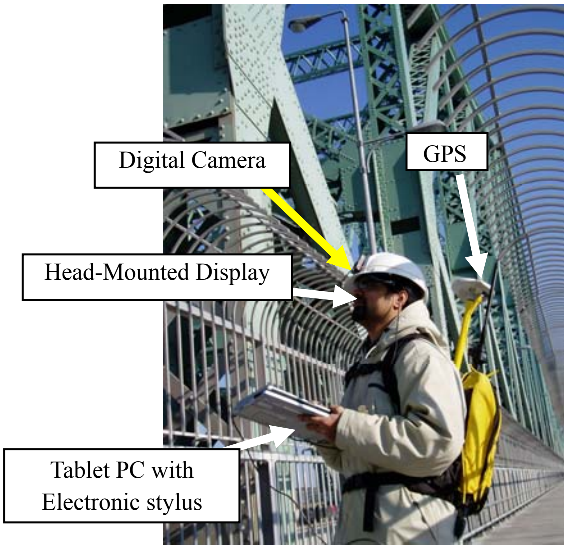
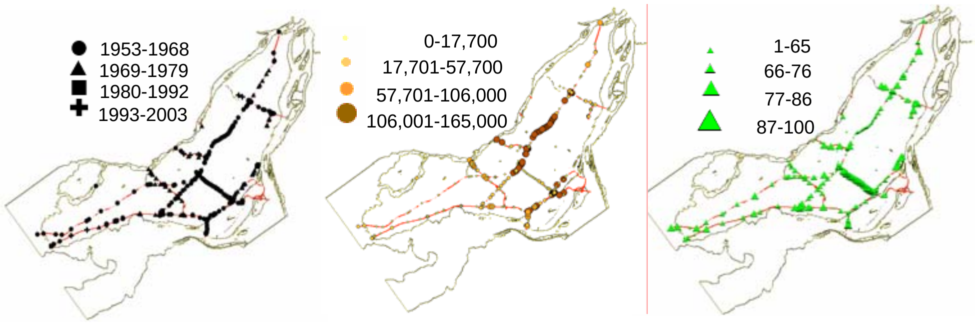


Figure 3.4 Mobile devices used in the prototype system

3.5 INTEGRATEING 2D AND 3D MODELS

Integrating 2D GIS maps with the BMS database helps in visualizing basic bridge information, such as bridge age, traffic volume, and condition rating. GIS map layers allow the visualization of data based on categories, quantities, and attributes. From Figures 3.5(a-c), the bridge inspector can easily refer to the distributions of the ages of bridges, traffic volumes, and bridge state indices using a point symbol for each bridge. In addition to the point representation, lines or polygons can also be used to represent bridges in a GIS layer (Figure 3.6). These different representations have different LoDs about the bridges. For example, the polygon representation of a bridge can be considered as an approximate representation of the bridge deck and can be used to calculate the deck area for the purpose of inspection and maintenance activities. The locations of bridges can be defined from available road maps or aerial photos.

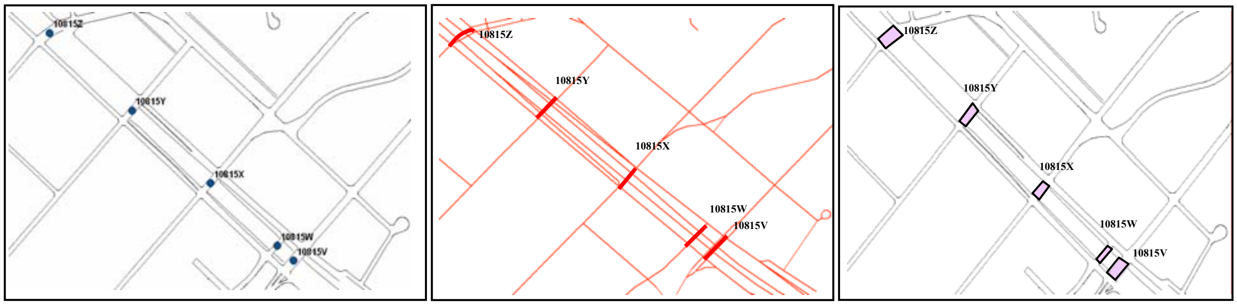


(a) Construction year

(b) Annual average daily traffic volume (vehicles/day)

(c) Combined index of structure (percentage)

Figure 3.5 Distribution of bridge attributes in Montreal

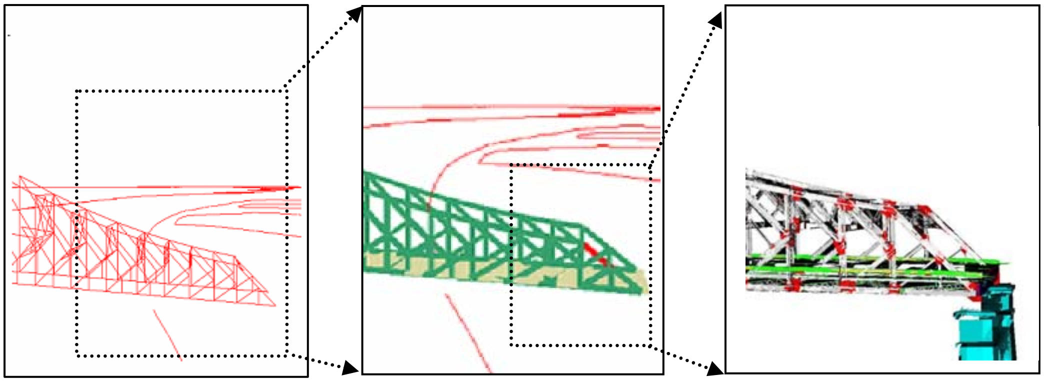


(a) Points (b) Lines (c) Polygons

Figure 3.6 GIS LoDs of bridge representation

Not only 2D GIS maps, but also 3D models of bridges can also be integrated with a BMS. The LoDs in 3D models can be used to create simpler versions of a bridge to meet different precision needs and to improve the image-rendering performance. When the viewer is far from the bridge, a simplified model can be used to speed up the rendering. Due to the distance, the simplified version looks approximately the same as the more detailed version.

Figure 3.7 shows an example of three LoDs for a bridge according to the distance between the viewpoint and the model: wire frame, prismatic elements and detailed 3D model. The more detailed model is shown when the user is nearer to the bridge. The 3D representations provide a powerful model-based interface for retrieving and updating inspection data and other information. However, this method is expensive and can be justified only in the case of major bridges where a 3D CAD model is usually available. With the integration of the 2D maps and 3D models, several tools developed for supporting location-based bridge inspection have been explained in Section 2.4.



(a) Wire frame (b) Prismatic elements (c) Detailed model

Figure 3.7 3D LoDs of bridge representation

3.6 LOCATION-BASED AUTOMATIC BRIDGE SELECTION AND NAVIGATION

While in the field, a bridge inspector may inspect a number of small bridges in a short time (e.g., short span bridges crossing over a highway in an urban area) or a single large bridge that may take several days. In both cases, the inspector usually has an approximate plan of the sequence of tasks to be achieved and the route to follow between the different locations that should be visited. The optimization of the inspection route can be used to improve the efficiency of inspection (Mizuno et al., 2002). However, following a rigid plan may not be practical because of the difficulty of anticipating all the factors that could cause a change in the sequence of inspection tasks. For example, the inspector may discover an unexpected problem in a part of the bridge that triggers the need to visit other locations to check other bridges or bridge elements that have not been considered in the plan. In this case, the inspector would benefit from being able to automatically retrieve those bridges or bridge elements based on their relative locations with respect to his/her present position.

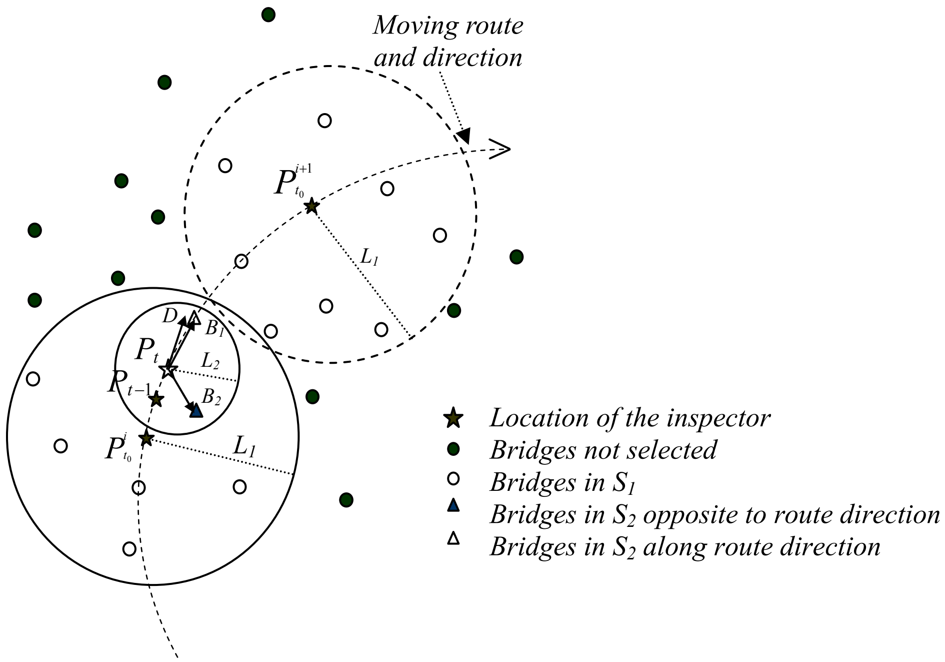

Moving route and direction
$P_{t_0}^{i+1}$
L_1
D
B_1
P_t
L_2
P_{t-1}
B_2
$P_{t_0}^{i}$
L_1
★ Location of the inspector
● Bridges not selected
○ Bridges in S_1
▲ Bridges in S_2 opposite to route direction
△ Bridges in S_2 along route direction

Figure 3.8 General concept of selecting the nearest bridge using GIS and tracking

Retrieving bridge information from the BMS database in real time may not be efficient because of the large number of bridges and bridge elements. In this section, as a first step towards facilitating the automatic retrieval of relevant inspection information, a location-based automatic bridge selection algorithm integrating GIS and GPS is developed based on the distance between the user and a set of bridges represented by their center points as shown in Figure 3.8. This algorithm can be extended in the future to the more general case of retrieving information about bridge elements based on a 3D spatial model.

Figure 3.8 shows a conceptual diagram of how the bridges are selected based on the distance between the inspector and a set of bridges. As the inspector moves from one location to another, the set of bridges is selected in three steps: (1) A larger set of bridges (S_1) is periodically selected (every Δt $=T$) within a distance L_1 from the inspector's position (P_{t_0}) at initial time t_0; (2) A smaller set of bridges (S_2) is continuously selected from S_1 within a smaller distance L_2 from the inspector's position (P_t) at current time t; and (3) A final set of bridges (S_3) is selected from S_2 by choosing only those bridges that are within the field of view of the inspector. Figure 3.9 shows the flowchart of the process of selecting the nearest bridges using GPS and GIS. In order to avoid excessive and unnecessary updates of the position based on GPS data, the required frequency (f) for updating the GIS map as well as the minimum distance (d_{min}) between two successive positions can be set. In addition, the quality of the position information is considered based on the Dilution of Precision (*DOP*). *DOP* is the geometric effect caused when satellites visible to the GPS receiver are too close to each other. When the satellites are further apart, the position obtained from the GPS is more accurate. The *DOP* can range from an ideal value of 1 to the least accurate value of 50. The maximum allowed *DOP* value (DOP_{max}) required for updating the position can be set to assure the reliability and accuracy of the position data. The values of L_1, L_2 and T are specified depending on the speed of movement of the inspector. Assuming the inspector is moving at an average walking speed, the following values could be used: T = *10 minutes*, L_1=*1000 m* and L_2= *100 m*. The initial time t_0 is set to time t which is the current time at the end of every period of length T. A set of bridges S_1 is selected within a distance L_1 from P_{t_0} . A new position (P_t) read from the GPS is considered only if the *DOP* is less than DOP_{max}. The location P_t is then converted from Latitude/Longitude (*LL*) coordinates to the map projection coordinates. If P_t satisfies the conditions of f and d_{min}, the current position is set at P_t. If the time difference ($\Delta t = t\text{-}t_0$) is greater than T, t_0 is reset to the current time and a new set of bridges S_1 within distance L_1 from the new initial position $P_{t_0}^{i+1}$ is selected. Otherwise, another set of bridges S_2 within L_2 from P_t is selected from $S_{1.}$ The third step in the selection is to select only those bridges (S_3) that are in the semicircle corresponding to the direction of movement of the inspector ($\vec{D}$). These bridges can easily be found by noticing that the dot product of the vector $\vec{D}$ and a vector $\vec{B}$ connecting the present position P_t and the location of a bridge satisfying the above condition will have a positive value (Equation 3.1).

$$\vec{D} \bullet \vec{B} > 0 \qquad \text{Equation (3.1)}$$

As an example of this selection, bridge B_1 in Figure 3.8 will be selected in S_3, while bridge B_2 will be eliminated from S_3. The bridges in S_3 are listed in the user interface in the order of increasing distance from P_t and the information regarding these bridges is retrieved from the BMS database. This process is repeated until the tracking is stopped.

After selecting the next target of inspection, navigation is used to help inspectors understand their upcoming tasks. Two types of navigation are proposed in this research: logical navigation and graphical navigation (Reinhardt et al., 2004). As shown in Figure 3.10, the logical navigation is represented by a hierarchical tree, which includes the structure of the bridge as extracted from the database. The graphical navigation is achieved by inserting an arrow to guide the inspector to the approximate locations where a selected type of defect usually occurs or to the locations of the defects found during previous inspection. This navigation support helps novice inspectors to detect the potential defects in a short time.

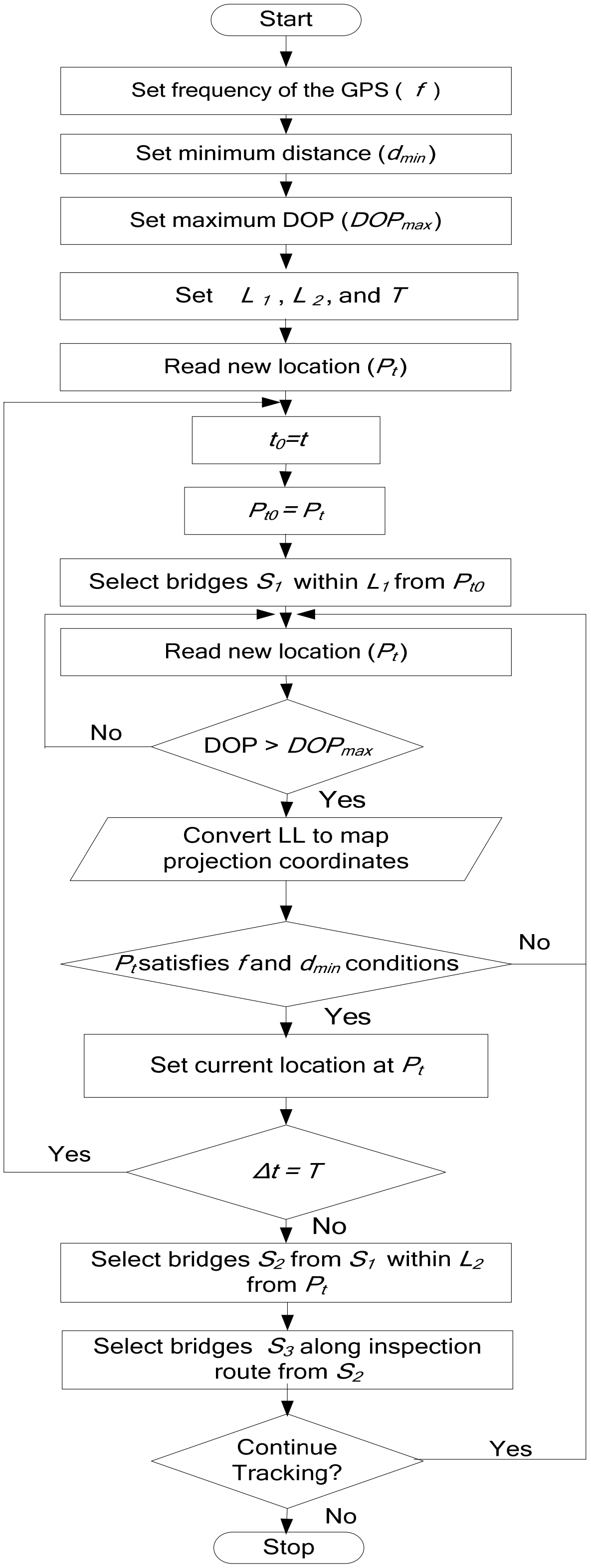

Start
Set frequency of the GPS (f)
Set minimum distance (d_{min})
Set maximum DOP (DOP_{max})
Set L_1, L_2, and T
Read new location (P_t)
$t_0=t$
$P_{t0} = P_t$
Select bridges S_1 within L_1 from P_{t0}
Read new location (P_t)
No
DOP > DOP_{max}
Yes
Convert LL to map projection coordinates
P_t satisfies f and d_{min} conditions
No
Yes
Set current location at P_t
Yes
$\Delta t = T$
No
Select bridges S_2 from S_1 within L_2 from P_t
Select bridges S_3 along inspection route from S_2
Continue Tracking?
Yes
No
Stop

Figure 3.9 Flowchart of selecting the nearest bridge using GIS and tracking

3.7 PICKING AND MARKING FOR LOCATING DEFCTS ON 3D MODEL

Interaction with the 3D model is mainly facilitated by selecting an element of the 3D bridge model. The selection is done by picking the element with a picking device, such as a mouse or an electronic stylus. Using the picking behavior defined in the user interface, the data collected during on-site inspection can be added directly to the 3D model eliminating the need to draw sketches as is usually required in present inspection practice. Picking is the process of selecting shapes in the 3D virtual world using the 2D coordinates of the picking device. A pick shape is selected as the picking tool. The pick shape could be a ray, segment, cone, or cylinder. The pick shape extends from the viewpoint location, through the picking device location and into the virtual world. When a pick is requested, pickable shapes that intersect with the pick shape (e.g., pick ray) are computed. The pick returns a list of objects, from which the nearest object can be computed. Each element in the 3D environment has a predefined ID, which is related with the data stored in the database. Upon selection, the element will be highlighted and a query is activated to retrieve the matching information in the database.

In order to interactively retrieve or update information related to the picked element, it is important to know the location and the orientation of that element in the 3D environment of the virtual model. During inspection, the system allows the inspector to mark the defect on the 3D model by directly adding a symbolic 3D shape onto the surface of the inspected element in the model after picking the approximate location of the observed defect. The location of the defect is represented by the point of the picking. Different defects can be represented with different shapes and the level of the defect can be represented with different colors. Figure 3.10 illustrates how to locate defects on 3D floor-beam model using picking and marking functions.

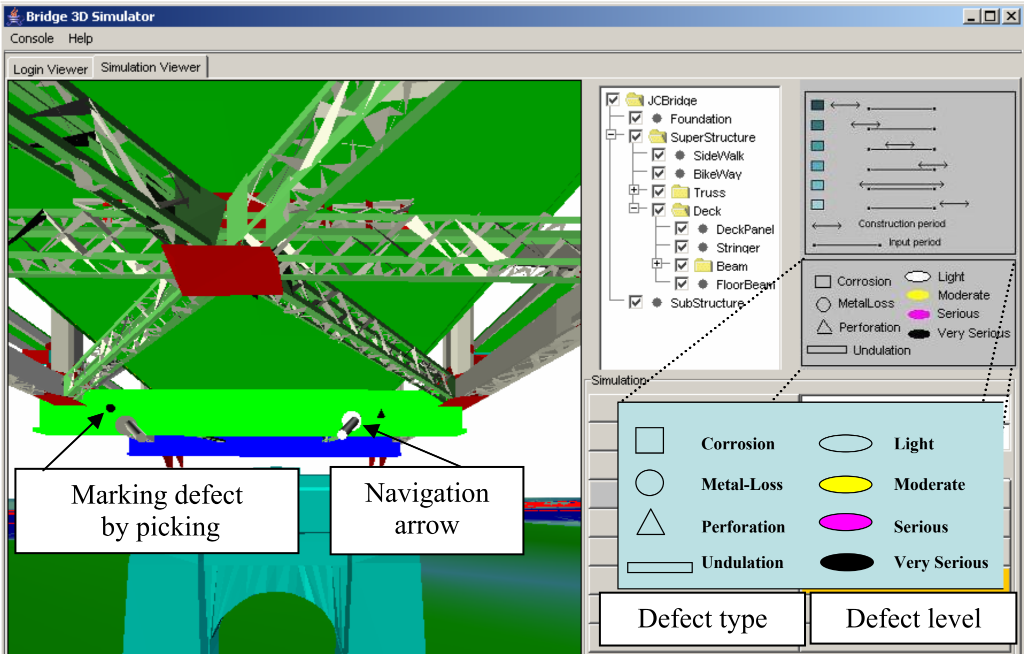


Figure 3.10 Design of location-based bridge inspection using 3D model

3.8 SUMMARY AND CONCLUSIONS

In this chapter, we have proposed a new location-based mobile computing approach to support the data collection activities during bridge inspection. The following conclusions about the proposed approach can be stated: (1) A new mobile location-based computing approach to re-engineer data collection of bridge inspection was designed; (2) A well designed GUI used in mobile situations to facilitate on-site data collection was developed; (3) The integration of GIS and a 3D model with suitable LoDs was discussed; (4) A new algorithm for the automatic selection of bridges using GPS tracking was developed; and (5) New interaction techniques for navigation, picking, and marking defects on the 3D bridge model were investigated.

Chapter 4 will discuss how to use the collected inspection data to accurately and effectively evaluate bridge condition. For this purpose, a DSS is developed integrating the proposed system with a rule-based ES and a learning-based BN.

CHAPTER 4 BRIDGE INSPECTION DECISION-SUPPORT SYSTEM

4.1 INTRODUCTION

In Chapter 3, mobile location-based techniques have been discussed for improving bridge inspection data collection. This chapter discusses how to use the collected data to evaluate elements condition rating. During bridge inspection, an attempt is made to determine the condition of an element based on the subjective opinion of the inspector and to understand the most probable causes of the observed defects in order to identify the appropriate measures to prevent further deterioration. However, the inspection evaluation results may vary depending on the level of experience of the inspector. In order to make up for the experience gap between inspectors, it is of great interest to develop a DSS to effectively support inspectors.

In this chapter, two methods, a rule-based ES and a learning-based BN, are considered to support inspectors when evaluating the condition rating of bridge concrete decks.

4.2 RULE-BASED ESs

Figure 4.1 illustrates how rules are defined for bridge inspection evaluation. Inspection codes and manuals and relevant experiences have been converted into rules and saved in the knowledge base. Specific rules have been customized to be activated when asserted facts match the Left Hand Side (LHS) of a rule, and the Right Hand Side (RHS) of this rule will output the results or generate new asserted facts. For example, after completing the inspection input of a floor-beam, condition rating of the beam can be obtained from the ES. Through extracting relative data from the database, existing facts can be asserted into the ES. Then, predefined rules in the ES are fired when their LHS are satisfied. The condition rating acquired from the ES is displayed and saved into the database.

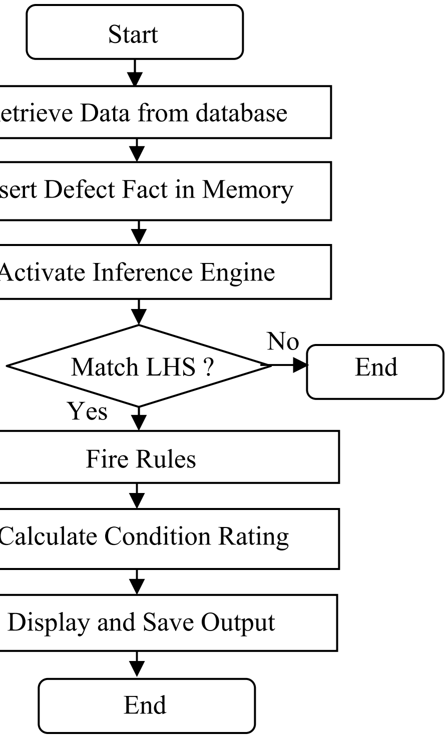

Start
Retrieve Data from database
Insert Defect Fact in Memory
Activate Inference Engine
Match LHS ?
No
End
Yes
Fire Rules
Calculate Condition Rating
Display and Save Output
End

Figure 4.1 Bridge element condition rating evaluations

4.3 CONCRETE DECK DIAGNOSIS

For the simple condition rating evaluation based on a small set of rules, a rule-based ES works well. However, if inspection data having multiple relationships and incomplete values, developing the rule-based ES used for element condition evaluation becomes more complex, time consuming and less accurate. Therefore, a new approach, learning-based BN, is discussed to analyze bridge concrete deck deterioration and to support inspectors making decision on-site. Using the EM algorithm for case learning, the results of the BN are tested with additional cases to spot weaknesses in the network and to improve performance by collecting additional data for learning. To demonstrate the feasibility and usefulness of the learning-based BN for diagnosing concrete bridge deck deterioration, the procedure of creating BN based on the EM algorithm is developed.

Poole et al. (1998) outlined the necessary steps for the development of a well-designed BN: (1) Define the relevant variables; (2) Define the states of the variables; (3) Establish the relationships between the variables; and (4) Calculate CPTs of the nodes. The proposed approach for learning-based BNs requires extracting data from inspection records, defining nodes and states, generating case file, establishing relationships between variables, calculating CPTs using an EM algorithm, and testing the network with new cases. These steps of the proposed approach are shown in Figure 4.2.

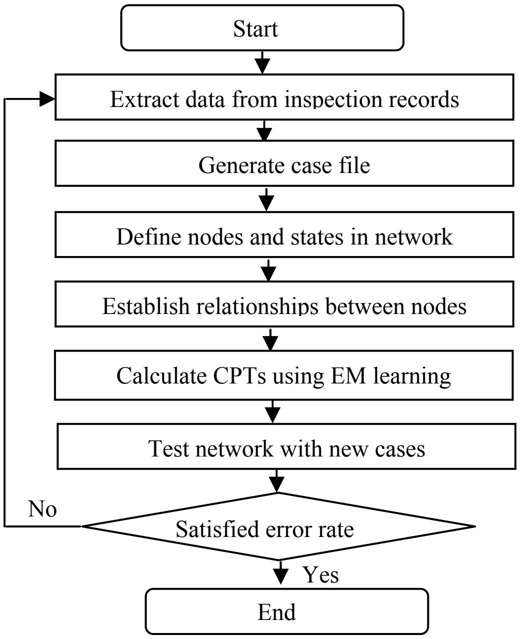
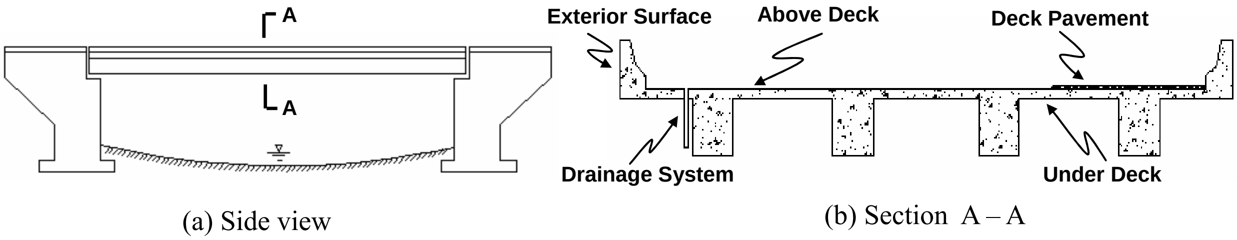

Start
Extract data from inspection records
Generate case file
Define nodes and states in network
Establish relationships between nodes
Calculate CPTs using EM learning
Test network with new cases
No
Satisfied error rate
Yes
End

Figure 4.2 Flowchart of creating a BN for bridge deck diagnosis based on learning

As shown in Figure 4.2, before creating a BN, variables and states should be extracted and defined from inspection records based on inspection manuals to generate a case file. In general, variables defined in the case file involve major causes that influence deck deterioration, defects types, condition evaluation, and maintenance activities. States of a variable are the range of the values taken by that variable. After reading the data from the case file, nodes and states are defined in the BN. The relationships between the nodes need to be established manually depending on engineering knowledge. The relationships between the nodes are categorized into four layers: cause layer, effect layer, condition evaluation layer, and maintenance layer. The nodes in the upper layers are the causes of the nodes in the lower layers. The CPTs of the network are calculated using the EM learning algorithm based on the defined states. In order to verify the accuracy of prediction of the network, an error rate test for each node is undertaken using a set of new cases. The test allows the user to find the nodes where the predictions are less accurate. The user can reexamine the CPTs of these nodes or supply additional data for learning. The detailed process of each step is described in the following subsections.

4.3.1 Definition of inspection data

In order to define the data to be extracted from inspection records, it is important to understand the structure of the inspection database. In this research, the inspection data obtained from the MTQ is used as an example. The MTQ data contains general information of each bridge, such as *Age of the Deck*, *Annual Average Daily Traffic (AADT)*, *Average Daily Truck Traffic (ADTT)*, and *Bridge Class*. The database also includes detailed description of defects and deterioration evaluation for different components of bridge decks. Based on the inspection manual used in Quebec (MTQ, 2004), the main types of defects of concrete decks are *delamination*, *cracks*, *spalls*, and *reinforcement steel corrosion*. In addition, the deterioration degree (e.g., *light, medium, important,* and *severe*) and percentage of these defects vary depending on the specific deck area. Therefore, five deck areas are defined to evaluate the conditions of the deck, and an overall deck condition is calculated based on them. The five areas are: *Deck Pavement (DP)*, *Drainage System (DS)*, *Exterior Surface (ES)*, *Under Deck (UD)*, and *Above Deck (AD)* (Figure 4.3) (MTQ, 2004). Each area is evaluated using two values: Material Condition Rating (*MCR*) and Performance Condition Rating (*PCR*) with the range for each value from 1 to 6 (a higher value represents a better condition). The *MCR* is an indication of the condition of a component based on the severity and extent of the observed defects in its material. The *PCR* is an indication of the condition of a component based upon its ability to perform its intended function in the structure. According to the values of *MCR* and *PCR*, maintenance activities are suggested for the specific deck area.



(a) Side view

(b) Section A – A

Figure 4.3 Concrete deck evaluation areas (Source: MTQ, 2004)

4.3.2 Case file generation and nodes and states definition

In this study, the case file is generated from the MTQ inspection database for one inspection period (2000-2003). The database contains inspection records of 10335 bridges. The data used for the case generation are for bridges in Montreal (297 bridges). Variables and states extracted from the inspection information are represented in the case file in a certain format (Table 4.1). The first row includes variables that are converted into nodes in the network. The values in each column under a variable are converted into states of the corresponding node.

The states of some variables with known values (e.g., Age and Class) can be directly obtained from inspection records. However, the states of other variables have to be defined manually by the user based on the description of the defect (e.g., the dimensions of a crack can be used to define the deterioration level). The states defined by the user should match the deterioration levels specified in inspection manuals (MTQ, 2004; OSIMS, 2000). For example, the cracks description of deck exterior surface extracted from MTQ inspection data is given as following: "*Longitudinal cracks 2-3 mm on 20% of deck exterior surface"*. Based on cracks classification in Table 4.2, the severity of these cracks can be defined as *Very Severe*. Using Figure 4.4 and combining the information about the severity level and the percentage of loss in the exterior surface (20%) for a primary component (i.e., exterior surface), the cross point of the dash lines falls in zone 1, which means that the value of MCR of the exterior surface is 1 based on the observed cracks.

Table 4.1 Sample case file

Variables

Age	AADT	ADTT	Class	Deterioration_DP	Blockage_DS	Cracks_ES	PCR_DP	MCR_DP
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
A40	T200	T50	C3	Severe	Partial	Thread	P2	M1
A40	T200	T10	C1	*	Partial	Wide	P5	M5
A40	T200	T10	C1	*	Partial	Narrow	P5	M5
A40	T100	*	C3	*	Partial	*	P1	M2
A40	T200	T10	C1	*	No	*	P2	*
A40	T100	T5	C3	Severe	Partial	Narrow	P1	M1
A40	T100	*	C3	*	No	*	*	M4
A40	T200	T50	C3	Severe	Partial	*	P2	M3

Missing data

States of one variable

Table 4.2 Cracks classification in MTQ (Source: MTQ, 2004)

Severity	Dimension (mm)
Thread (Light)	0-1.125
Narrow (Medium)	0.125-0.7
Average (Severe)	0.7-1.375
Wide (Very Severe)	>1.375

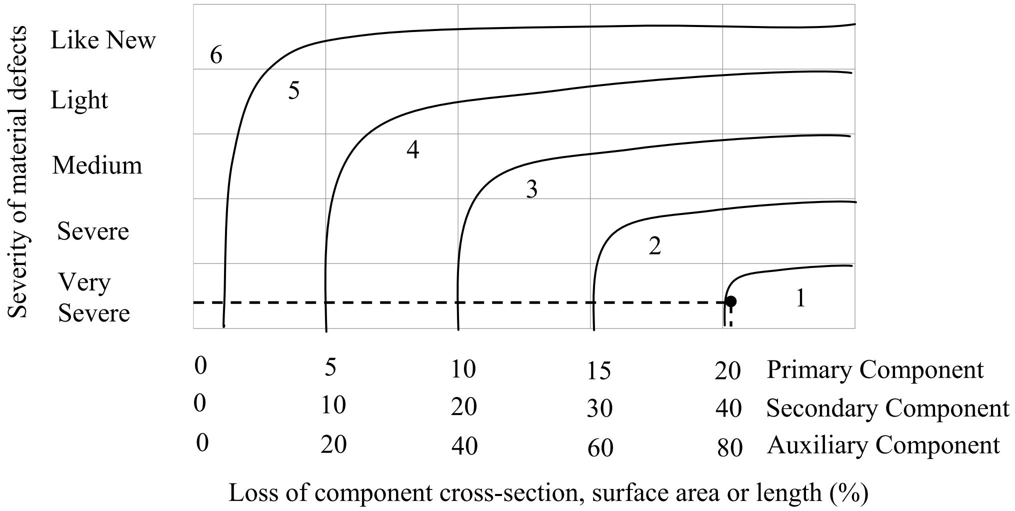


Figure 4.4 Material condition rating (MCR) of element in MTQ (Source: MTQ, 2004)

The percentage of defect in certain deck area can be obtained from an expert opinion or through calculation. Figure 4.5 gives an example about how to calculate the percentage of delamination on a beam to evaluate the MCR of under deck.

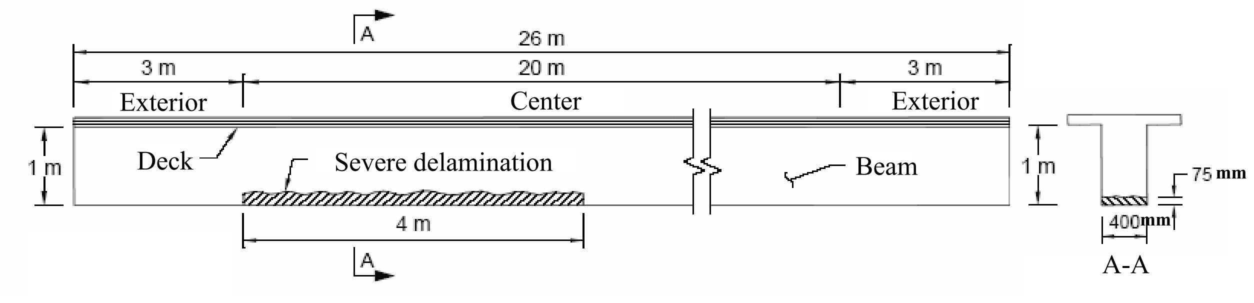


Figure 4.5 Delamination on bridge beam (Source: MTQ, 2004)

Assuming the severity of material defect is *Severe*, and then delamination of the MCR for the middle of the beam (Primary Component) is calculated as follow:

Section transverse:	
(75*400)/ (1,000*400) = 7.5%	MCR = 4 (From Figure 4.4)
Surface:	
(400+2*75) *4,000/ (2,400*20,000) = 4.5%	MCR = 5 (From Figure 4.4)
Length:	
4,000 / 20,000 = 20%	MCR = 2 (From Figure 4.4)

MCR 2 is retained for the middle of the beam

The evaluation of PCR of element component can be gained through Table 4.3.

Table 4.3 Performance Condition Rating of element in MTQ (Source: MTQ, 2004)

Element component evaluation (PCR)				
Component		Percentage of decrease of the capacity of an element		
Code	Description	Primary	Secondary	Auxiliary
6	Excellent	0-1%	0-2%	0-5%
5	Good	1-5%	2-10%	5-20%
4	Acceptable	5-10%	10-20%	20-40%
3	Inferior	10-15%	20-30%	40-60%
2	Weak	15-20%	30-40%	60-80%
1	Critical	>20%	>40%	>80%

Not all states can be obtained from inspection records and missing data are represented by an asterisk (*). Missing data mainly result from the different details in the textual description of the defects. In our case, there are at least three types of defects' descriptions by different inspectors. Some descriptions record defects with very detailed information (e.g., location, dimensions, and percentage), while others have only few words even if the deterioration level is severe. The values used for network learning are derived from those inspection records. Variables in Table 4.4 are the nodes defined in the cause layer, such as *Deck Age*, *AADT*, *ADTT*, and *Bridge Class*. The states in *Deck Age* are classified as ranges of 10 years each (Table 4.5 (a)). For example, *A50* represents deck age between 41-50 years. Figure 4.6 shows a distribution of construction year of bridge deck in Montreal.

Table 4.4 States classification defined in the cause layer

	Variables			
	Deck Age (Year) (1)	AADT (1000 veh./day) (2)	ADTT (1000 veh./day) (3)	Bridge Class (4)
States	A10	T100	T1	C1
	A20	T200	T5	C2
	A30	T300	T10	C3
	A40	T400	T100	
	A50	T500	T150	
		…	T200	
		T1700	T300	

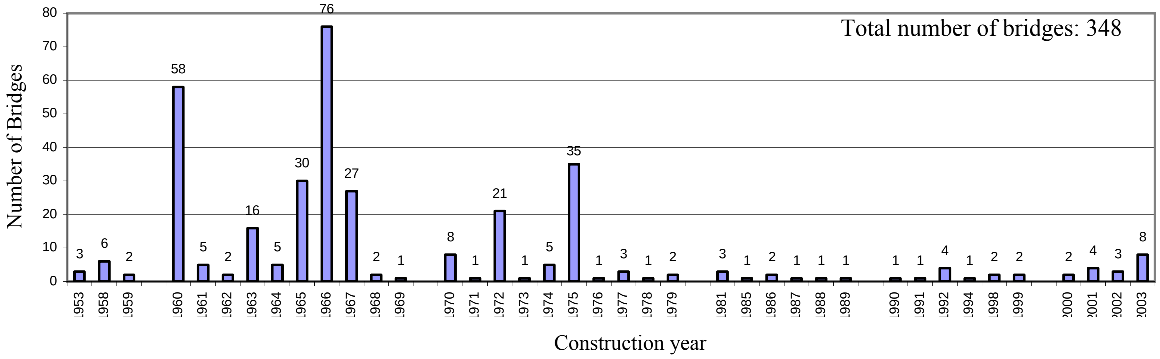


Figure 4.6 Construction year of bridge deck in Montreal

The states *AADT* and *ADTT* are classified in several ranges. For instance, *T100* in *AADT* represents 0-100,000 vehicles per day (Table 4.5 (b)) and *T100* in ADTT represents 5,001-10,000 heavy vehicles per day (Table 4.5 (c)). In the MTQ data, *ADTT* normally accounts for 7-13% of *AADT*. The states in *Bridge Class* are classified into three types: *C1*, *C2*, and *C3*. *C1* means the bridge has the highest priority.

Table 4.5 States classification for Age, AADT, and ADTT

(a) Age

Construction year	State	Range
1953	A50	41-50
1963	A40	31-40
1973	A30	21-30
1983	A20	11-20
1993	A10	0 -10

(b) AADT (Vehicles/per day)

State	Range
T100	0-10,000
T200	10,001-20,000
T300	20,001-30,000
T400	30,001-40,000
T500	40,001-50,000
T600	50,001-60,000
T700	60,001-70,000
T800	70,001-80,000
T900	80,001-90,000
T1000	90,001-100,000
T1100	100,001-110,000
T1200	110,001-120,000
T1300	120,001-130,000
T1400	130,001-140,000
T1500	140,001-150,000
T1600	150,001-160,000
T1700	160,001-170,000

(c) ADTT (Vehicles/per day)

State	Range
T1	1-100
T5	101-500
T10	501-1,000
T50	1,001-5,000
T100	5,001-10,000
T150	10,001-15,000
T200	15,001-20,000
T300	20,001-30,000

Variables in Table 4.6 are the nodes in the effect layer, including the major defect types occurring in concrete decks, and the states in each variable represent the severity of defects. Table 4.7 lists two types of condition evaluation variables of decks: *MCR* and *PCR*, which are the nodes in the condition evaluation layer with the values in the range from 1 to 6. Variables in Table 4.8 show the five deck areas that need to be maintained. The measures of maintenance are defined as states for the corresponding variables. Tables 4.6-4.8 list all the rest variables and states used in creating the network.

Table 4.6 States classification defined in the effect layer

	Variables					
	Deterioration	Blockage	Delamination	Cracks	Spalls	Steel Corrosion
States	Light	No	Light	Thread	Light	No
	Medium	Partial	Medium	Narrow	Medium	Light (> -200mv)
	Important		Important	Average	Important	Uncertain (-200mv~ -350mv)
	Severe		Severe	Wide	Severe	Activity (< -350mv)

Table 4.7 States classification defined in the condition evaluation layer

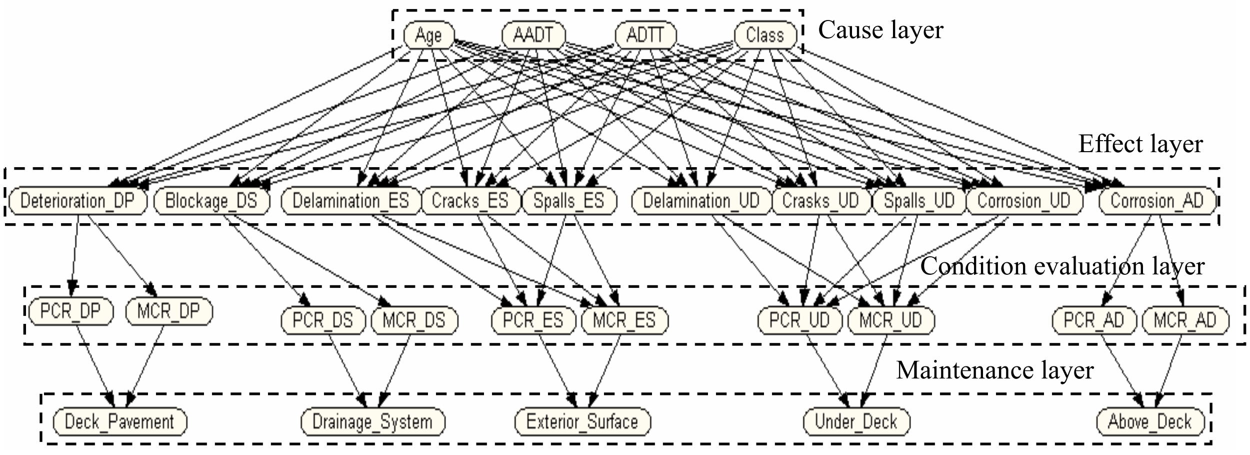
	Variables	
	MCR	PCR
States	M6	P6
	M5	P5
	M4	P4
	M3	P3
	M2	P2
	M1	P1

Table 4.8 States classification defined in the maintenance layer

	Variables				
	Deck Pavement	Drainage System	Under Deck	Exterior Surface	Above Deck
States	Patching	Cleaning	No	No	No
	Seal crack	Modification	Yes	Yes	Yes
	Resurface	Repair			
	Repair				

4.3.3 Relationship establishment and case learning

After generating the case file including 150 cases, nodes and states in the network are created automatically by reading the case file. Causal arrows are added manually based on cause-effect relationships and engineering knowledge. As mentioned in Section 4.3.1, the relationships between the nodes are categorized into four layers (Figure 4.7): cause layer, effect layer, condition evaluation layer, and maintenance layer. Each layer consists of several nodes that directly affect the nodes in the adjacent lower layer. The nodes in the cause layer are specified as the general factors that result in deck deterioration, such as *deck age*, *AADT*, *ADTT*, and *bridge class*. The nodes in effect layer include defect types occurring at the five areas of the deck, such as *delamination, cracks, spalls*, and *reinforcement steel corrosion*. Based on the types and deterioration levels of the defects, condition ratings of each deck area (*PCR* and *MCR*) are assessed in the condition evaluation layer. Finally, maintenance measures for individual deck areas can be planned based on the *PCR* and *MCR*, as shown in the maintenance layer. The case study discussed in Section 5.4 focuses on the application of the effect layer and the condition evaluation layer. The nodes in the cause layer do not cover all the factors that can affect concrete deck deterioration, such as *de-icing chemical*, *protective system*, and *maintenance actions*, because of the large set of missing data in the inspection records.



AADT: Annual Average Daily Traffic
ADTT: Average Daily Truck Traffic
PCR: Performance Condition Rating
MCR: Material Condition Rating

DS: Drainage System
ES: Exterior Surface
UD: Under Deck
AD: Above Deck
DP: Deck Pavement

Figure 4.7 Structure of the BN for concrete deck deterioration

Once the network structure is ready, learning can be undertaken using the predefined cases to generate CPTs of each node (Figure 4.8). This step is done automatically by the system using the parametric EM algorithm.

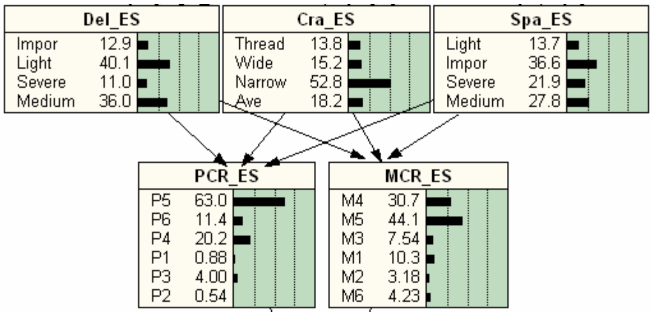


Figure 4.8 CPT generation of each node

4.3.4 Network testing

In order to test the usability and accuracy of the network obtained from the case learning, it is necessary to grade the network using a set of new cases to see how well the predictions or diagnosis of the network match the actual cases. The test allows the user to find the nodes where the predictions are less accurate. The user can reexamine the CPTs by supplying additional data for learning. The basic idea of this test is to divide the net's nodes into two classes: observed and unobserved. The observed nodes will have their values read from the case file and be used to predict the values of the unobserved nodes by using BN updating. This process is repeated for each case in the case file. For each such case, the predicted values for the unobserved nodes are compared with those that are actually observed in the case file. All successes and failures are recorded. These statistics are gathered up and presented in a final report that describes the performance of each unobserved node, i.e., how often the predictions are accurate.

For instance, node *MCR_UD* (MCR for the under deck area in Figure 4.7) is selected as unobserved node. The possible states of *MCR_UD* are *M1, M2, M3, M4, M5,* and *M6*. For each case processed (total 100 new cases with 10% missing data), the BN generates beliefs for each of these states from observed cases. The most likely state is chosen as its prediction for the value of *MCR_UD*. Then the results are compared with the actual value of *MCR_UD* of that case. In the sample report shown in Table 4.9, the error rate of node *MCR_UD* calculated using Equation 4.1 is 10.11%.

Error Rate = (Number of wrongly predicted values) / (Total number of values) x 100%

= 9 / 89 x 100% = 10.11% Equation (4.1)

Table 4.9 MCR_UD testing with cases

Actual Value	*Number of Predicted Values*						*Total*
	M1	M2	M3	M4	M5	M6	
M1	**5**	0	0	0	0	0	5
M2	0	**4**	0	0	0	0	4
M3	0	0	**6**	0	0	0	6
M4	0	0	1	**12**	2	0	15
M5	1	0	0	1	**49**	3	54
M6	0	0	0	0	1	**4**	5
							89

Figure 4.9 shows the different error rates of the nodes in the network (except those in the maintenance layer) based on 50, 100 and 150 cases used for learning. In each group, the error rate for the selected unobserved node is tested using the same new 100 cases with average missing data of 10%. Obviously, the higher number of cases used for learning, the more accurate are the prediction results of the BN. The average error rate of the nodes in the BN using 150 cases learning is about 10%, which is considered satisfactory for the purpose of bridge inspection.

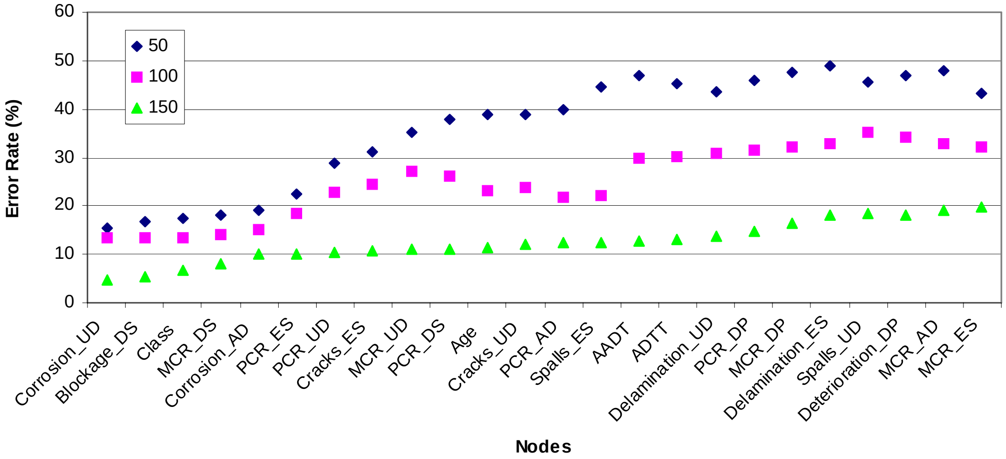


Figure 4.9 Comparison of error rates from different numbers of cases used in learning

It is useful to know how much the belief in a particular node is influenced by other nodes, and how sensitive they are. BNs can efficiently measure this influence using sensitivity analysis. Therefore, the major causes can be found from the nodes with high degree of influence. When calculating the CPTs from a set of cases, the results of the CPTs consist of unexpected probabilities from missing data. The major factor that generates these probabilities is the high percentage of missing data for the node itself or its parent nodes. For example, when the deterioration level of deck pavement in the *effect layer* is set to *Severe*, the states of *PCR* of deck pavement (*PCR_DP*) in the *condition evaluation layer* are distributed in states *P1, P2,* and *P5*. However, according to the actual cases, this *PCR* should have states of only *P1* and *P2*. The appearance of the value *P5* is because of the large amounts of missing data of deck pavement (58% data missing). In addition, the percentage of missing data affects also the error rates of the nodes. For example, the error rates of *MCR_AD* and *MCR_ES* (Figure 4.9) are higher than those of other nodes because the percentages of missing data in these nodes are 62.7% and 65.3%, respectively.

4.4 SUMMARY AND CONCLUSIONS

In this chapter, rule-based ES was discussed and tested for the application of element condition rating evaluation. Due to some limitations in rule-based ES, a new DSS based on BNs using the EM algorithm for case learning was proposed to support inspectors in the diagnosis of bridge concrete deck deterioration. The following conclusions about the proposed approach can be drawn: (1) A case file can be generated using previous inspection records and relevant manuals for network learning; (2) A deck deterioration BN can be built using case learning; (3) The learned BN was tested with a set of new cases to verify the accuracy of the prediction (average error rate of 10% when using 150 cases); and (4) The developed user interface of the BN was found useful in investigating the cause-effect probabilities related to deck deterioration.

In Chapter 5, the location-based mobile bridge inspection DSS, implemented in Java, is demonstrated using several case studies to support inspectors for data collection and condition assessment during bridge inspection.

CHAPTER 5 IMPLEMENTATION AND CASE STUDIES

5.1 INTRODUCTION

To demonstrate the feasibility and usefulness of the proposed methodology, a prototype system is developed and discussed in detail in this chapter. The prototype system is built using Java language and integrating a 3D bridge model, an object-relational database, an ES, a GIS, a GPS interface, an inspection GUI, and a multimedia interface. Due to the different inspection data (the MTQ data and the Jacques Cartier Bridge data) acquired from different bridge management agencies, two case studies for LBC and one case study for DSS are given to demonstrate different parts in the prototype system.

5.2 DATA ACQUISITION AND PROTOTYPE SYSTEM STRUCTURE

The MTQ data used in the case studies are about the 300 bridges in Montreal. These data cover general inspection descriptions of bridge and are suitable for the network-level analysis. The detailed 3D bridge model of the Jacques Cartier Bridge is used in the element-level bridge inspection case study.

5.2.1 MTQ database

The data used in developing the case studies were obtained from the MTQ during one inspection period (2000-2004). The database consists of 10335 province-owned structures that are categorized into eight categories of structures: culverts, slab bridges, beam bridges, box-girder bridges, truss bridges, arch bridges, cabled bridges, and other structures. The percentages of these structure types and the number of structures in each category are presented in Figure 5.1 and Appendix C, respectively. The MTQ database including three types of data (e.g. inventory data, inspection data, and maintenance data) for each of the bridges is recorded in 11 tables (Table 5.1). The Structure Inspection Manual and Damage Evaluation Inspection Manual are used to explain the contents of the database (MTQ, 2004).

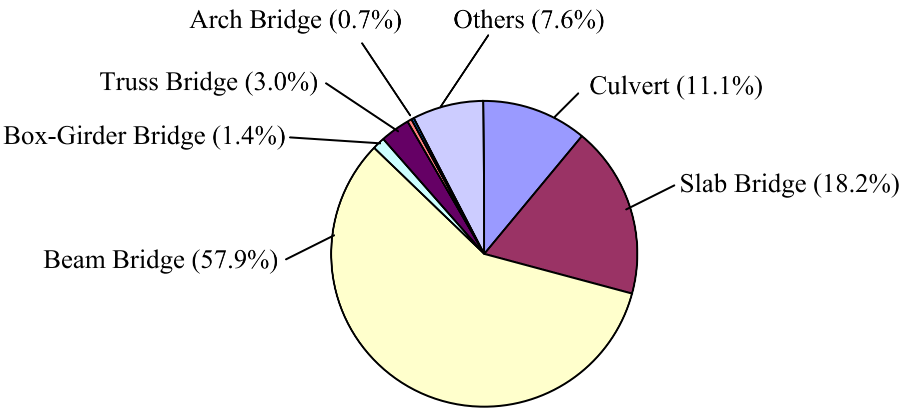


Figure 5.1 Structure types and numbers in Quebec

Table 5.1 Inspection tables used by MTQ (Source: MTQ, 2004)

Type	Table name	Description
Inventory	SGSD010P	General information about structure
Inspection	SGSD400P	Obstacles in the section inventory
	SGSD410P	The elements of foundation in the inventory
	SGSD420P	The structural systems of the inventory
	SGSD700P	Inspection form (type A - Z)
	SGSD710P	Details about inspection form
	SGSD720P	The inspection evaluation
	SGSD730P	The inspection comment
	SGSD740P	The inspection summary
Maintenance	SGSD750P	Maintenance activities
	SGSD770P	Maintenance cost

Inventory data consist of approximately 220 data items that can be categorized into administrative, technical, and descriptive data. Administrative data include data items for bridge identification, location, importance, classification, and jurisdiction. Technical data include data items about the traffic over and under the bridge and postings on the bridge. Descriptive data include data items that describe the geometry, material, and structural system of different bridge components. They also include data about the various utilities on the bridge. More information about these data items is available in MTQ (2004).

Inspection data consist of detailed visual inspection of 135 bridge elements collected every 3 years. These data are collected using 21 different inspection forms (Table 5.2), each of which corresponds to a group of correlated bridge elements, such as foundation elements, truss elements, and deck elements, etc. An example of the bridge deck inspection form is presented in Table 5.3. This table is also the basis for deck condition evaluation. In column *Type*, the letters P, S, and A represent primary element, secondary element, and auxiliary element, respectively. Each inspection form includes the Material Condition Ratings (MCR) and Performance Condition Ratings (PCR) of the inspected element in addition to some remarks and recommendations regarding the maintenance of these elements (MTQ, 2004).

Maintenance data represent the major maintenance actions that are expected to be made in the future. The cost and the recommended period of each action are roughly estimated. Although the developed system is designed to consider the effort of previous maintenance actions on the predicted condition, the data of these actions will not be included in testing the system because of its unavailability.

Table 5.2 Inspection forms used in Quebec (MTQ, 2004)

Inspection Form	Form Description	No. of Inspected Elements
A	Signalistion	4
B	Water stream, approaches backfilling, and slope protection	5
C	Abutment and wing wall	7
D	Foundation and substructure	6
E	Deck	7
F	Solid beam	7
G	Box girder	8
H	Truss beam	5
I	Arch beam	6
J	Arch with spanderal wall	9
K	Structure floor	7
L	Wind bracings	5
M	Deck Joints	7
N	Curbs and side walks	3
O	Barriers	9
P	Approaches	7
Q	Retaining walls	6
R	Culverts	6
S	Covered bridge	11
U	Suspension bridge	7
V	Cable-Stayed bridge	3
		135

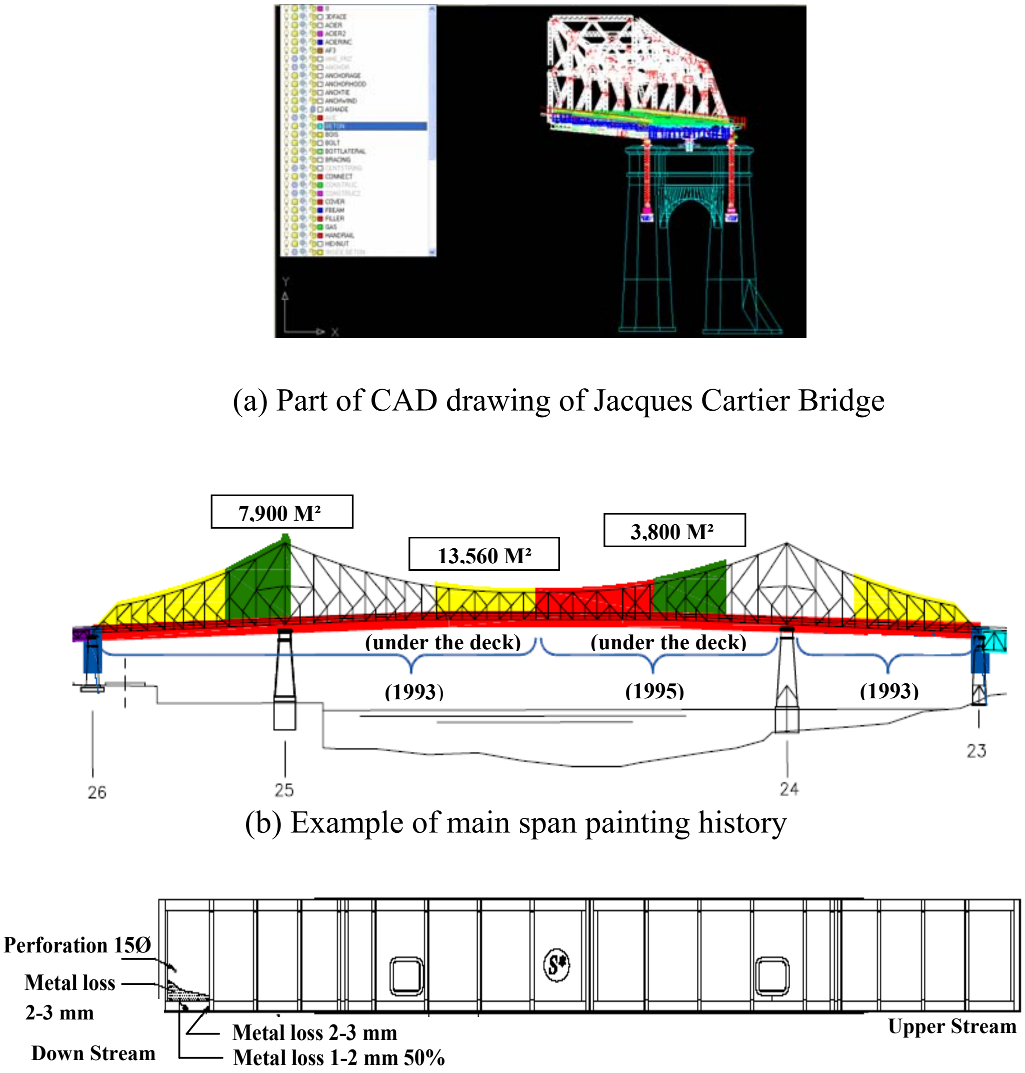
Table 5.3 Inspection Form E for deck evaluation (MTQ, 2004)

No.	Type	Element	MCR	PCR	Description
1	S	Wearing surface			
2	A	Drainage system			
3	P	Exterior surface			
4	P	Exterior surface			
5	P	End soffit (under deck)			
6	P	Middle soffit (under deck)			
7	P	End soffit (under deck)			
8	P	Above concrete deck			

5.2.2 Jacques Cartier Bridge database

The Jacques Cartier Bridge data were acquired from the bridge management authority (The Jacques Cartier and Champlain Bridges Incorporated) (PJCCI, 2004; Zaki and

Mailhot, 2003). The data include AutoCAD drawings, deck rehabilitation schedules and inspection and maintenance records. Figure 5.2(a) shows a CAD drawing of a part of the Jacques Cartier Bridge. Figure 5.2(b) shows an example of the inspection data of a floor-beam including metal loss and perforation. Figure 5.2(c) shows main span painting history of the bridge. These data have been used in the development of the prototype system.



(a) Part of CAD drawing of Jacques Cartier Bridge

(b) Example of main span painting history

(c) Example of floor-beam inspection information

Figure 5.2 Examples of Jacques Cartier Bridge data
(Source: PJCCI, 2004; Zaki and Mailhot, 2003)

5.2.3 Prototype system structure

The prototype system is developed based on the above data. In order to easily group the data that are related, the related classes and interfaces are organized in different packages in the system. A package is a collection of related classes and interfaces providing access protection and namespace management. The general structure of the packages developed in the prototype system is illustrated in Figure 5.3. The *bridge3d.core* package creates the 3D bridge shape. The *bridge3d.model* package includes the interfaces to the database. The *bridge3d.ui* package has the main GUI classes. Figure 5.4 shows the relationships in the inspection database. The database consists of six tables (i.e., *Inspector, Truss, Date and Time, Instrument, Damage,* and *Task*) with detailed inspection contents in each table, and the tables are linked by several primary keys for retrieving the relevant information.

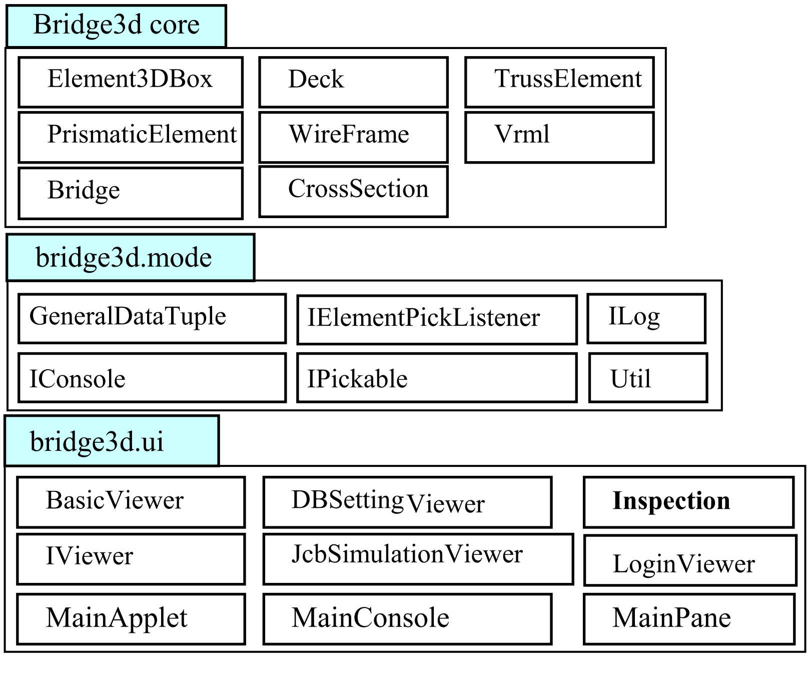


Figure 5.3 Classes and packages defined in the prototype system

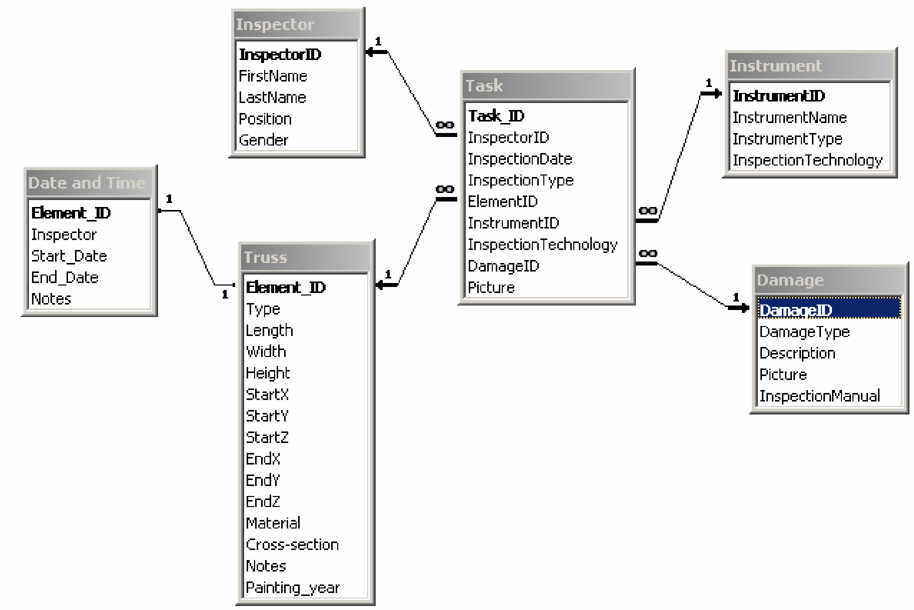


Figure 5.4 Relationships in the inspection database

5.3 CASE STUDIES OF LBC

The hardware used in the prototype system includes a tablet PC with electronic stylus (Fujitsu T Series (Fujitsu, 2005), Toshiba Tecra M4 Tablet PC (Toshiba, 2005), and Panasonic Toughbook-18 (Panasonic, 2005)), an HMD (Micro Optical SV-6 PC Viewer (MicroOptical, 2005) and Microvision Nomad ND2000 (Microvision, 2005)), an RTK GPS (Trimble 5700 (Trimble, 2005)), and a digital camera (Logitech-3000). The equipment photos and the specifications of each device can be found in Appendix D. The LBC prototype system is built using Java language and integrating a 3D bridge model, an object-relational database, an ES, a GIS, a GPS interface, an inspection GUI, a multimedia interface (JMF, 2004), and a hyperlink function. The database is designed with Microsoft Access XP and is accessed using Java Database Connectivity (JDBC). The data can be retrieved and updated using Structured Query Language (SQL).

With respect to the LBC applications, the nearest inspection target can be found based on a large set of positions of bridges using the bridge selection algorithm integrating GIS and GPS. A 3D model of the Jacques Cartier Bridge is created using Java3D, which is an API for developing portable applications and applets that can run on multiple platforms (Walesh and Gehringer, 2001). Based on the bridge 3D model, functions such as navigation, picking, and marking are developed. In the following subsections, two case studies for LBC application are discussed using the MTQ data and the Jacques Cartier Bridge data, respectively, to demonstrate the feasibility of the prototype system.

5.3.1 Case study of bridge selection algorithm

Taking advantage of the bridge selection algorithm explained in Section 3.6, the inspector equipped with a GPS receiver can use his/her present position and the locations of the bridges in the GIS system to select the nearby bridges and order the inspection tasks for the selected bridges according to their distances along the inspection route (Figure 5.5). Once a bridge is selected, the related information is automatically retrieved from the BMS database (Figure 5.6). In this example, four bridges are selected within *100* m from the position of the inspector, and the sequence of inspection is determined by the distance along the inspection route. In addition, the specific inspection information about the nearest bridge (10815M) is retrieved from the database.

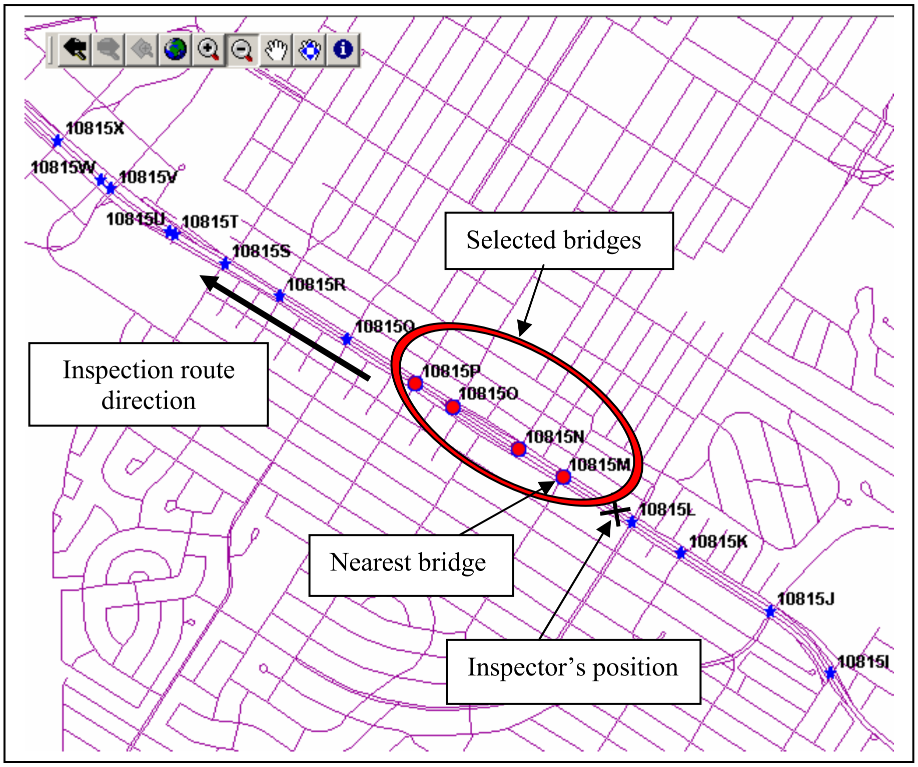
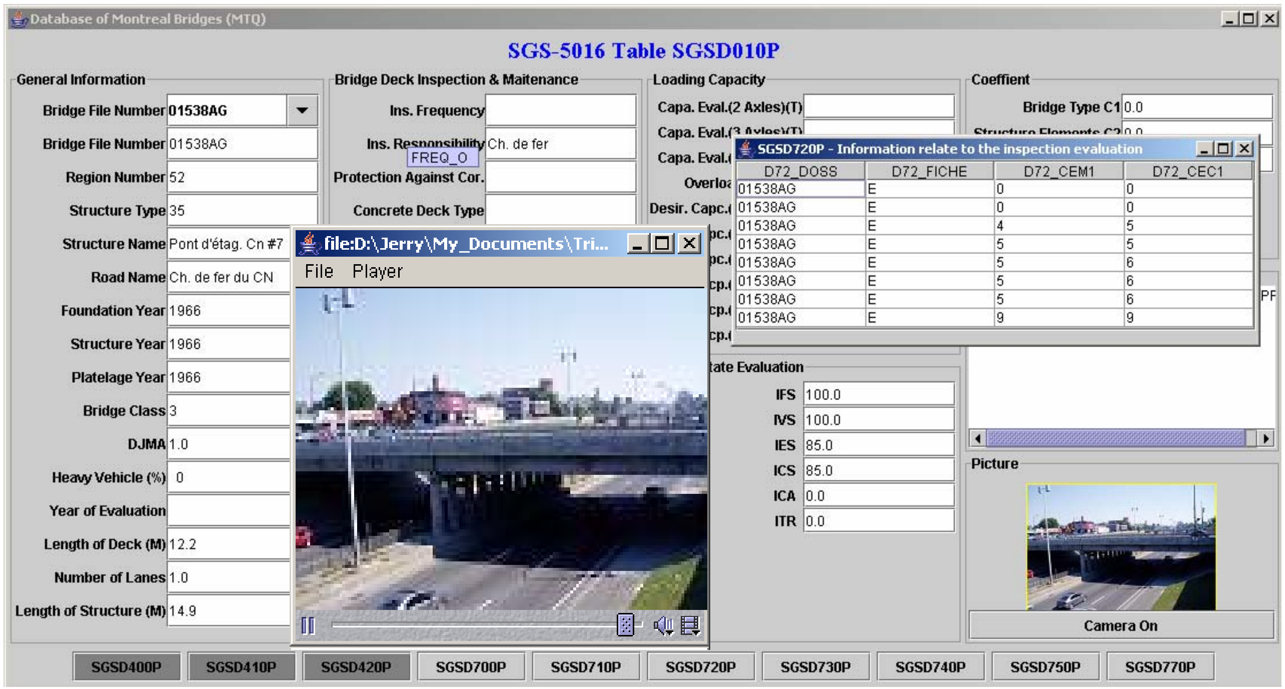


Figure 5.5 Finding the nearest bridges in GIS along inspection route



SGS-5016 Table SGSD010P

General Information

Bridge File Number 01538AG
Bridge File Number 01538AG
Region Number 52
Structure Type 35
Structure Name Pont d'étag. Cn #7
Road Name Ch. de fer du CN
Foundation Year 1966
Structure Year 1966
Platelage Year 1966
Bridge Class 3
DJMA 1.0
Heavy Vehicle (%) 0
Year of Evaluation
Length of Deck (M) 12.2
Number of Lanes 1.0
Length of Structure (M) 14.9

Bridge Deck Inspection & Maitenance

Ins. Frequency
Ins. Responsibility Ch. de fer
FREQ_O
Protection Against Cor.
Concrete Deck Type

Loading Capacity

Capa. Eval.(2 Axles)(T)
Capa. Eval.(3 Axles)(T)
Capa. Eval.(
Overloa
Desir. Capc.(

Coeffient

Bridge Type C1 0.0
Structure Elements C2 0.0

SGSD720P - Information relate to the inspection evaluation

D72_DOSS	D72_FICHE	D72_CEM1	D72_CEC1
01538AG	E	0	0
01538AG	E	0	0
01538AG	E	4	5
01538AG	E	5	5
01538AG	E	5	6
01538AG	E	5	6
01538AG	E	5	6
01538AG	E	9	9

tate Evaluation

IFS 100.0
IVS 100.0
IES 85.0
ICS 85.0
ICA 0.0
ITR 0.0

Picture

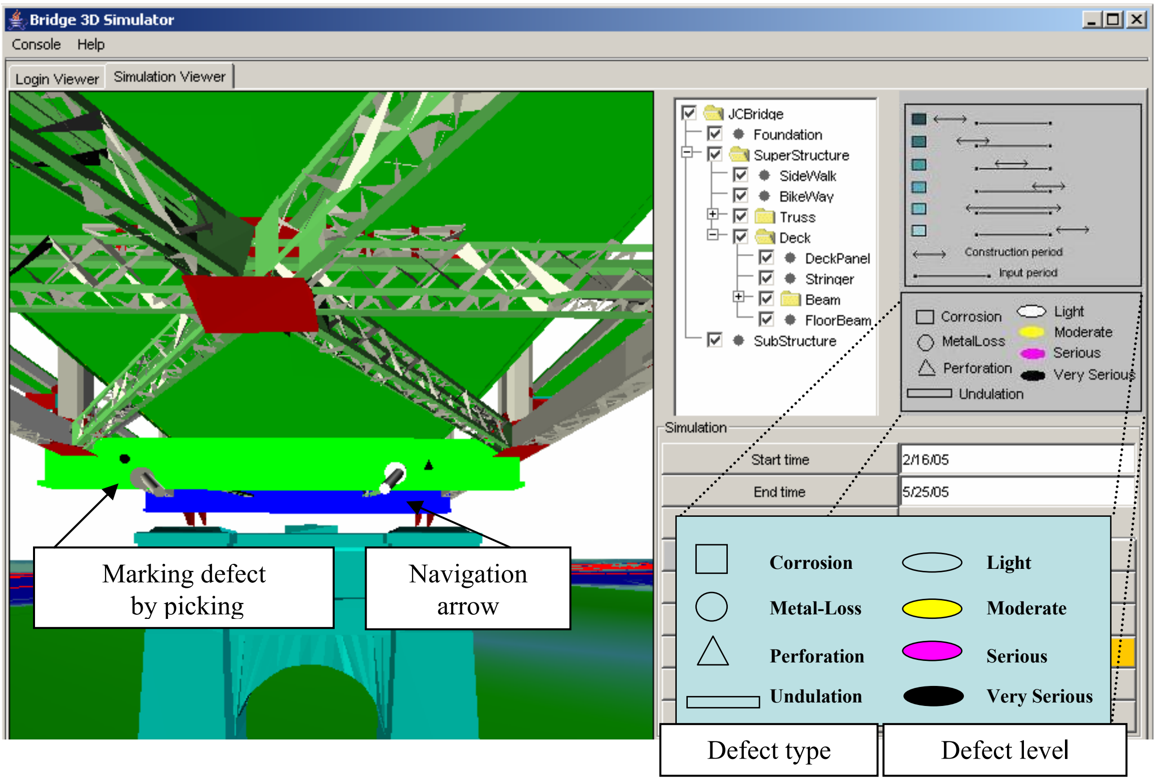
Figure 5.6 Bridge database interface

The interface comprises general information of the bridge, deck inspection and maintenance, loading capacity, coefficient factors, structure state index, comment, and photo capture function. In the current prototype system, the inspector can take photos of each inspected bridge and save them into the database automatically, and the photos can be updated following the latest inspection after the bridge is chosen.

5.3.2 Case study of location-based bridge inspection using 3D model

The main user interface of the system includes a time input interface that allows the user to query the database about events that took place during a specific period (e.g., what are the defects found during the previous inspection? What is the sequence of replacing the deck panels in 2001?). The start and finish dates of a period can be input using a calendar interface or sliding bars, and the 3D model will reflect the corresponding elements with different colors representing the progress ratio. A hierarchical tree of the bridge structure is shown on the right side. Each tree node has a check box, which facilitates showing or not showing that element in the 3D model. In addition, the user can navigate the 3D bridge model and select an element of the bridge by picking that element. Upon selection, the element is highlighted and the related information about the element is displayed. Alternatively, the user can select an element from the database interface and the element is highlighted in the model.

Figure 5.7 shows the location-based visual inspection process with the navigation and picking functions. At the beginning of the inspection activities, virtual arrows automatically guide the inspector with a predefined inspection order according to the inspection plan. Following this step, the inspector is asked to select the specific defect type. The possible locations of the selected defect type are indicated on the inspected element using animated arrows. The arrows are created dynamically and inserted into the scene graph. The trajectory of the arrow is computed based on the present position of the inspector (obtained from tracking) and the location of the defect. Figure 5.7 shows an example of picking a floor-beam in the 3D model to input the locations of several defects. The defects are automatically marked on the 3D model of the floor beam using specific shapes and colors, which are defined based on the defect type and deterioration degree, respectively. For instance, in Figure 5.7, the black sphere represents very serious metal loss.

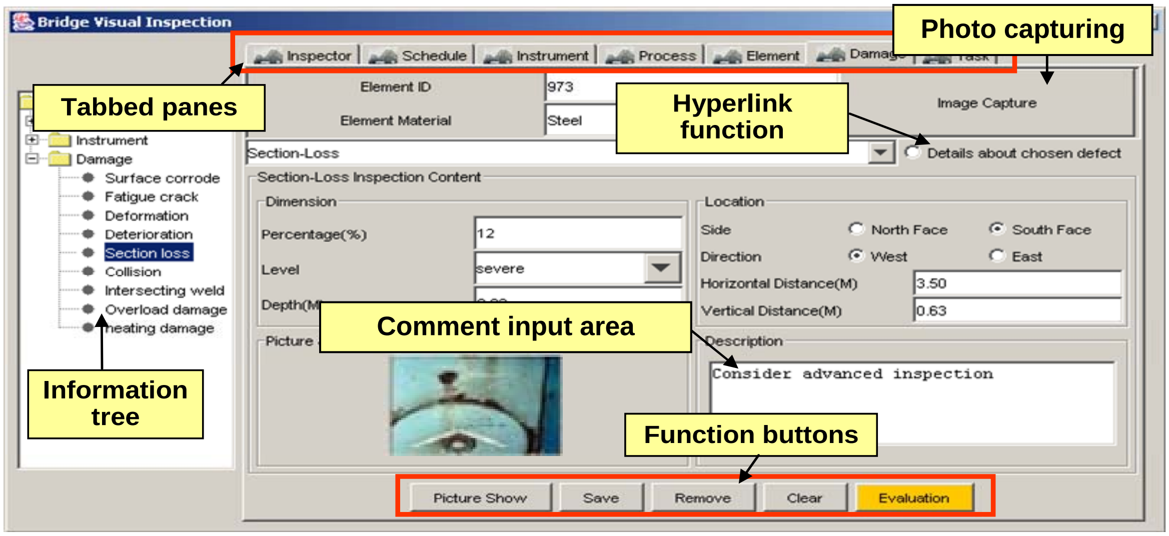


Start time 2/16/05

End time 5/25/05

Figure 5.7 Implementation of location-based bridge inspection using 3D model

After picking the approximate position of the defect based on visual observation, the inspector can complete the data input and save these data with the help of the defect inspection input pane (Figure 5.8). In this interface, an inspector can apply inspection procedures by using a number of ordered tabbed panes. The panes are *Inspector, Schedule, Instrument, Process, Element, Damage,* and *Task*. In the first two tabbed panes, some general inspection information needs to be input about the inspector and the schedule. The user can find, add, and update the bridge inspection data by querying the database. In the *Instrument* pane, a suitable inspection tool can be selected depending on the type of the defect. The *Process* pane is used to guide the inspector to find the elements that need to be inspected based on the schedule or the previous defect locations. In the *Element* pane, the inspector can choose the element to inspect according to a customized inspection scheme by picking the element on the 3D model at the approximate location. In this pane, basic information about the chosen element, such as type, material, and dimensions, will be retrieved from the database and added to the pane. The *Damage* pane is the core part of the bridge inspection interface. Video/image capture functionality has been implemented using JMF API to collect and save defect image information. The hyperlink functionality links to a Hypertext Markup Language (HTML) format file according to the defect type to allow the inspector to access inspection manuals. The link is context-sensitive and extracts only the relevant information. The last pane, *Task*, summarizes the previous inspection information for future assessment.



Element ID: 973

Element Material: Steel

Section-Loss

○ Details about chosen defect

Section-Loss Inspection Content

Dimension

Percentage(%): 12

Level: severe

Depth(M)

Location

Side: ○ North Face ● South Face

Direction: ● West ○ East

Horizontal Distance(M): 3.50

Vertical Distance(M): 0.63

Picture

Description: Consider advanced inspection

Figure 5.8 Defect inspection input pane

5.4 CASE STUDY OF INSPECTION DECISION-SUPPORT SYSTEM

In the proposed approach, Java Expert System Shell (JESS) is used as the rule-based inference engine (Friedman-Hill, 2003). Because JESS is a Java-based Application Programming Interface (API), it is possible to call Java functions from JESS, to extend JESS by writing Java code, and to embed JESS in Java applications. Figure 5.9 illustrates how to achieve interaction between a Java applet and JESS inference engine. In this figure facts are asserted from the applet interface into the working memory to activate the inference engine combined to the predefined rules. Using *Rete* algorithm to process rules in inference engine, JESS provides an efficient mechanism for solving the difficult many-to-many facts matching problem. The output is the element condition rating result after comparing all the available facts with the rules. An example of the interaction between Java and JESS can be found in Appendix E.

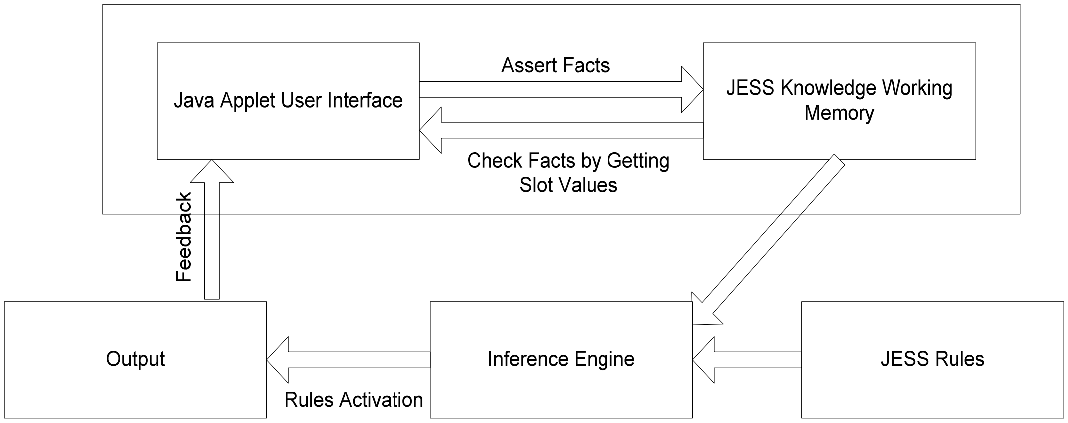

Java Applet User Interface
Assert Facts
JESS Knowledge Working Memory
Check Facts by Getting Slot Values
Feedback
Output
Inference Engine
JESS Rules
Rules Activation

Figure 5.9 Interactions between Java and JESS

Another application for the prototype system is to develop a DSS for bridge deck deterioration evaluation. This application used a learning-based BN based on the MTQ database. BNs have the ability to adjust variables to be input or output at any point without redesigning the system. Therefore, on one hand, causes input based on the defect records during an inspection can be used to calculate the probability of the evaluation of a concrete deck, which could effectively support inspectors when evaluating deck deterioration. On the other hand, understanding the causes of the deterioration is of utmost important. This understanding permits the inspector to spend more time evaluating the trouble areas of the bridge deck. A GUI is designed based on the learned BN (Figure 5.10) to effectively analyze the cause or effect probabilities for deck deterioration. The GUI (Figure 5.11) is developed using the Java API of a BN software (Netica, 2005). Using the GUI, inspectors are required to select one of the condition ratings of the specified deck component or a sequence of predefined defect descriptions which are used as new evidences in the network, and then the probabilities of an expected cause state or effect state are calculated and displayed. This application attempts to simulate human experience and provide decision support for on site visual inspection.

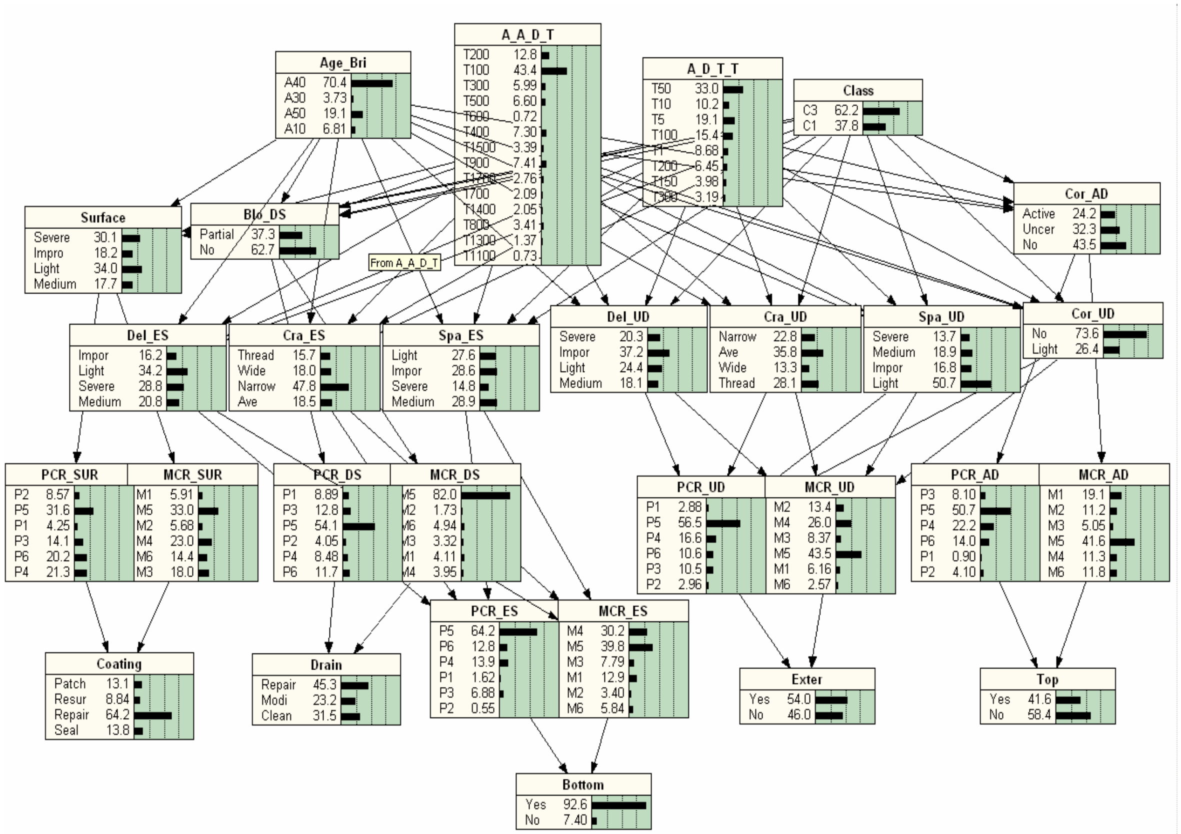
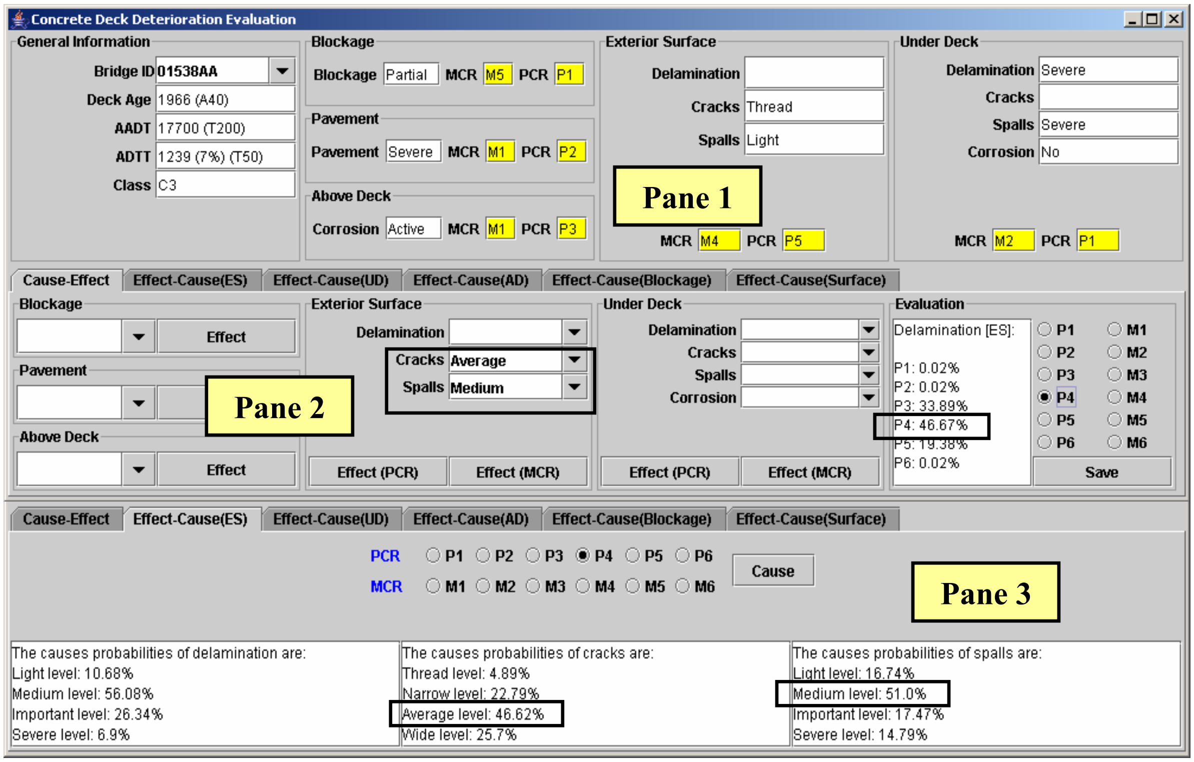


Figure 5.10 A learning-base BN for deck deterioration evaluation



General Information

Bridge ID 01538AA

Deck Age 1966 (A40)

AADT 17700 (T200)

ADTT 1239 (7%) (T50)

Class C3

Blockage

Blockage Partial MCR M5 PCR P1

Pavement

Pavement Severe MCR M1 PCR P2

Above Deck

Corrosion Active MCR M1 PCR P3

Exterior Surface

Delamination

Cracks Thread

Spalls Light

MCR M4 PCR P5

Under Deck

Delamination Severe

Cracks

Spalls Severe

Corrosion No

MCR M2 PCR P1

Cause-Effect | Effect-Cause(ES) | Effect-Cause(UD) | Effect-Cause(AD) | Effect-Cause(Blockage) | Effect-Cause(Surface)

Blockage — Effect

Pavement

Above Deck — Effect

Exterior Surface

Delamination

Cracks Average

Spalls Medium

Effect (PCR) Effect (MCR)

Under Deck

Delamination

Cracks

Spalls

Corrosion

Effect (PCR) Effect (MCR)

Evaluation

Delamination [ES]:

P1: 0.02%
P2: 0.02%
P3: 33.89%
P4: 46.67%
P5: 19.38%
P6: 0.02%

○ P1 ○ M1
○ P2 ○ M2
○ P3 ○ M3
● P4 ○ M4
○ P5 ○ M5
○ P6 ○ M6

Save

Cause-Effect | Effect-Cause(ES) | Effect-Cause(UD) | Effect-Cause(AD) | Effect-Cause(Blockage) | Effect-Cause(Surface)

PCR ○ P1 ○ P2 ○ P3 ● P4 ○ P5 ○ P6

MCR ○ M1 ○ M2 ○ M3 ○ M4 ○ M5 ○ M6

Cause

The causes probabilities of delamination are:
Light level: 10.68%
Medium level: 56.08%
Important level: 26.34%
Severe level: 6.9%

The causes probabilities of cracks are:
Thread level: 4.89%
Narrow level: 22.79%
Average level: 46.62%
Wide level: 25.7%

The causes probabilities of spalls are:
Light level: 16.74%
Medium level: 51.0%
Important level: 17.47%
Severe level: 14.79%

Figure 5.11 User interface for concrete deck deterioration cause-effect analysis

Figure 5.11 shows the user interface for the DSS. *Pane 1* includes general deck information, defects description, deterioration level, and condition evaluation of the five different deck areas. All the data can be retrieved from inspection records and can be displayed automatically in the interface when a bridge ID is selected by the inspector. Pane 2 shows an example about cause-effect relationship (i.e., from defect details to deterioration evaluation). In *Pane 2*, *Average Cracks* and *Medium Spalls* are discovered on the exterior surface of a deck and are considered as two defect causes. Based on the learned cases, the probabilities of the deterioration levels on the exterior surface can be found from the network. The probability of P4 (46.67%), as shown in *Pane 2*, is the maximum value. Thus, 4 would be the most likely value that the inspector should select for the PCR of the exterior surface of the deck. *Pane 3* shows another example of the effect-cause relationship (i.e., from deterioration evaluation to defect details). The selection of the evaluation level, such as P4, is given to summarize the current condition of the exterior surface. The inspector can acquire the probabilities of different defect types located at certain area based on the selected evaluation level. For instance, for the current condition of deck exterior surface equal to P4, the most likely causes are: Medium- level delamination (56.08%), Average-level cracks (46.62%), or Medium-level spalls (51.0%). With the help of this information, maintenance activities can be planned effectively.

5.5 VALIDATION OF THE MOBILE TRACKING

In order to test the proposed approach, the prototype system was used on August 27, 2005, in Montreal to perform two groups of inspection tasks: (1) Inspection of four small bridges crossing Highway 15 North (Figure 5.5); and (2) Inspection of a major bridge (The Jacques Cartier Bridge). In the first group, the four inspected bridges, from south to north, were the following: *Jean-Talon (10815M)*, *des Jockeys (10815N)*, *Paré (10815O)*, and *Ferrier (10815P)*. The distance between two sequential bridges ranges from *150* to *250 m*. The test covered the individual functionalities and the overall performance of the system in supporting inspection. The integration of GPS and GIS in the system provided accurate help in finding the inspected targets and in automatically retrieving the relevant inspection information from the database (e.g., previous bridge conditions). The bridge selection algorithm described in Section 3.6 as well as the GPS functionality were tested according to the example explained in Figure 5.5. One of our main concerns was about the availability of the GPS signals because it is well known that urban canopies affect this availability. However, the GPS signals were available in RTK mode throughout the duration of the test when walking on the bridges. The visual inspection tasks were simulated using the GUI of the system on the tablet PC, and some pictures were taken and automatically added to the database using the digital camera attached to the hardhat of the inspector.

In the second group of inspection tasks on the Jacques Cartier Bridge, the inspection data collection was done based on the 3D model. Compared with the traditional manual data input, using the prototype system improved the efficiency of data collection with the help of directly marking defects on the bridge 3D model as explained in Figure 5.7. However, the GPS tracking was not available at certain locations when walking under the bridge or even on the top of the bridge because of the obstruction caused by the truss elements.

Other problems were identified regarding the difficulty of reading the HMD or the display of the tablet PC because of the ambient light conditions and the small size of the characters in the GUI. Two types of HMDs were tested: Microvision Nomad ND2000 and MicroOptical SV-6. Microvision Nomad ND2000 has a rugged, monochrome red display (32 gray levels) readable in all lighting conditions with automatic brightness adjustment. MicroOptical SV-6 is smaller, less rugged and has color display. Both displays support resolution of 800x600 pixels. Appendix D shows the basic specifications of both displays. After testing these displays under different conditions, it was found that the MicroOptical SV-6 is more suitable for LBCInfra because of its overall superior visibility using colors. As for the size of the text displayed with the above resolution, it was found that a minimum font size of 25 points is required for comfortable readability. The typical displays found on most laptop and tablet computers are designed for use under the optimal conditions found within an office or home environment. Therefore, in outdoor test for readability and visibility of our three tablet PCs, the results showed that the displays cannot effectively prevent glare and reflectivity in high ambient light and sun-lit environments.

Further development and testing of the prototype system are needed to improve the usability of the system and identify and test the optimal hardware architecture suitable for the prototype system.

5.6 SUMMARY AND CONCLUSIONS

This chapter described the implementation of the proposed approaches and methods in Chapters 3 and 4. A prototype system, implemented in Java, is developed and several case studies about bridges in Montreal were used to demonstrate the feasibility of these approaches and methods. The software development tools were selected to integrate several information technologies in the prototype system. The system was demonstrated to engineers responsible of the bridge management and they gave positive evaluation. Furthermore, the preliminary testing of the system and its user interface showed that it has good potential for realizing future BMSs because it was carefully designed and implemented to satisfy the specific requirements of these systems.

In summary, mobile location-based DSS is useful in helping the bridge inspector easily collect inspection data on-site, reducing errors during inspection process, visualizing inspection scenario, and making decision about condition rating.

CHAPTER 6 SUMMARY, CONCLUSIONS, CONTRIBUTIONS, AND FUTURE WORK

6.1 SUMMARY

The core part of a BMS is a database that is built up of information obtained from regular inspection and maintenance activities. Among the various tasks of bridge management, field inspection is essential for evaluating the current condition of a bridge. Effective bridge management is thus heavily dependent on field inspectors to collect detailed condition information for all of the individual elements of a bridge and to evaluate bridge condition based on that information. Therefore, this research focused on developing a bridge inspection system that provides the BMS with accurate and effective on-site data collection and reliable condition evaluation.

The literature review of bridge inspection systems has revealed some improved implementations in this field, such as mobile computing technology for facilitating data collection and AI methods for supporting condition evaluation. However, on the one side, the systems used by mobile computing technology do not consider the interactions with 3D model. It is difficult to find the locations of objects and to simulate a real inspection scenario. On the other side, the application of AI methods in BMS is still limited. Rule-based ESs still involve the subjective judgment of the inspector and lack of flexibility when implemented in practice. Therefore, the mobile location-based bridge inspection DSS, which integrates the above mentioned approaches, is introduced to provide improved BMSs. In addition to the integration of several technologies, the proposed system tries to eliminate the shortcomings of previous research. For example, a well-designed GUI with various interaction functions implemented for mobile devices is more suitable for on-site visual bridge inspection activities. The 3D bridge model with advanced interaction functions (e.g., LoDs, navigation, picking, and marking) has been integrated with the mobile computing system to easily find and address the locations of defects. GIS and GPS are also used to accurately locate the positions of the inspector and inspection targets so that inspection quality and efficiency can be improved. In addition, BNs have been developed in a DSS for the probabilistic condition evaluation of bridge elements using case learning mechanism. BNs can be assessed using a combination of empiric studies and learning methods, which make probabilistic analysis more reliable and flexible.

Two proposed approaches, LBC and DSS, were discussed in this research with three case studies to demonstrate their applications. The data used in the case studies, including approximately 300 bridges in Montreal, were stored in the database. The prototype system is built using Java language and integrating a 3D bridge model, an object-relational database, an ES, a GIS, a GPS interface, an inspection GUI, and a multimedia interface. The 3D model is created using Java3D API. The database is designed with Microsoft Access XP and is accessed using Java Database Connectivity (JDBC). The data can be retrieved and updated using Structured Query Language (SQL). The DSS is developed using a BN software and its Java API (Netica, 2005). The preliminary testing of both approaches of LBC and DSS showed good potential to integrate them in future BMSs.

6.2 CONCLUSIONS

There are two groups of conclusions in this research. The first group is about mobile location-based bridge inspection. In this group, the following conclusions about the proposed approach can be stated: (1) A new mobile location-based computing approach to re-engineer data collection of bridge inspection was designed; (2) A well designed GUI used in mobile situations to facilitate on-site data collection was developed; (3) The integration of GIS and a 3D model with suitable LoDs was discussed; (4) A new algorithm for the automatic selection of bridges using GPS tracking was developed; and (5) New interaction techniques for navigation, picking, and marking defects on the 3D bridge model were investigated.

The following second group of conclusions about bridge inspection decision-support system can be drawn: (1) A case file can be generated using previous inspection records and relevant manuals for network learning; (2) A deck deterioration BN can be built using case learning; (3) The learned BN was tested with a set of new cases to verify the accuracy of the prediction (average error rate of 10% when using 150 cases); and (4) The developed user interface of the BN was found useful in investigating the cause-effect probabilities related to deck deterioration.

A prototype system, implemented in Java, was developed and several case studies about bridges in Montreal were used to demonstrate the feasibility of these approaches and methods.

6.3 CONTRIBUTIONS

The contributions of this research are grouped into the following areas:

(1) A framework of bridge inspection was developed including LBC technologies integration and a DSS development. Several computational issues necessary to realize the framework were also discussed considering both the hardware and software. The hardware comprised mobile devices, such as a tablet PC, a GPS, an HMD, and a digital camera. The software consists of the GUI design, GIS application, database management, 3D interaction, and element condition evaluation. This approach makes the first attempt to integrate 3D bridge models and intelligence expert system with a BMS making the information accessible to mobile on-site situation.

(2) Data collection using mobile computing technology for on-site visual bridge inspection has been developed and improved. In the prototype system, the concept of LBC applied to automatically retrieve the necessary information in real time based on the location, orientation, and specific task context of the inspector can result in the following advantages: (a) On-site data collection becomes more accurate and efficient using mobile devices; (b) Defects can be directly added to and retrieved from the 3D model eliminating the need to draw sketches as is usually required in present inspection practice; and (c) The nearest bridge inspection target can be automatically retrieved along the predefined inspection route based on GPS and GIS. The following conclusions about the LBC approach can be stated: (a) A new mobile location-based computing approach to re-engineer data collection of bridge inspection was designed;

(b) A well designed GUI used in mobile situations to facilitate on-site data collection was developed; (c) The integration of GIS and a 3D model with suitable LoDs was discussed; (d) A new algorithm for the automatic selection of bridges using GPS tracking was developed; and (e) New interaction techniques for navigation, picking, and marking defects on the 3D bridge model were investigated.

(3) With respect to modeling bridge deck deterioration, a new DSS approach that employs BNs to evaluate the current element condition was introduced. BNs are one of the AI approaches that can be used to solve uncertainty and imprecise issues involved in incomplete data through case learning method. This approach has eliminated some of the shortcomings of rule-based ESs and ANNs presented in Chapter 2. A learning-based BN applied in the prototype system is designed based on the collection of comprehensive engineering knowledge and tested with different number of cases for achieving a satisfactory error rate. The practical implementation of the net shows that it can effectively support non-experienced inspector to analyze the most likely damage related cause-effect relationships based on current inspection data. The following conclusions about the DSS approach can be drawn: (a) A case file can be generated using previous inspection records and relevant manuals for network learning; (b) A deck deterioration BN can be built using case learning; (c) The learned BN was tested with a set of new cases to verify the accuracy of the prediction (average error rate of 10% when using 150 cases); and (d) The developed user interface of the BN was found useful in investigating the cause-effect probabilities related to deck deterioration.

(4) A prototype system, implemented in Java, was developed and several case studies about bridges in Montreal were used to demonstrate the feasibility of the above mentioned approaches and methods. The system was demonstrated to engineers responsible of the bridge management and they gave positive evaluation. Furthermore, the preliminary testing of the system and its user interface showed that it has good potential for realizing future BMSs because it was carefully designed and implemented to satisfy the specific requirements of these systems. In summary, mobile location-based DSS was found useful in helping the bridge inspector easily collect inspection data on-site, reducing errors during inspection process, visualizing inspection scenario, and making decision about condition rating.

6.4 FUTURE WORK

While pursuing this research, several limitations have been identified related to the requirements and the performance of the developed methods and techniques.

(1) The usability of the prototype system for BMSs needs more testing. Further development and testing in practical situations are necessary to improve the functionalities and usability of the system.

(2) Because of the intervention of the bridge truss, the GPS receiver could not receive a stable signal when testing on the Jacques Cartier Bridge.

(3) The light conditions could influence the display effect on the tablet PC screen, which makes reading the screen display difficult.

(4) The BN for bridge deck deterioration evaluation used only 150 cases for learning and 150 cases for testing. More inspection cases can be added to improve the accuracy of the learned net. Furthermore, case file generation for the BN is time consuming because the information extracted from inspection description needs to be looked up manually. In some inspection records, defect descriptions are only few words even though the deterioration level of an element is severe. These incomplete data result from non-standard inspection input. In order to overcome this problem, our future work is to utilize current inspection input interface and database management to automatically generate the case file, which can be used for BN learning.

In order to enhance the capabilities of the mobile location-based bridge inspection DSS, the previous points need to be explored in the future research.

REFERENCES

Advitam-Group (2005). Web site: <http://www.advitam-group.com/>.

American Association of State Highway and Transportation Officials (AASHTO) (1993). "Guidelines For Bridge Management System", Washington, D.C.

American Association of State Highway and Transportation Officials (AASHTO) (1999). "Pontis: The Complete Bridge Management System", accessed Webpage: <http://aashtoware.camsys.com/html/PontisReport.html.>

Arditi, D. and Tokdemir, O. (1999). "Comparison of Cased-Based Reasoning and Artificial Neural Networks", Journal of Computer in Civil Engineering, ASCE, Vol.13, No.3, pp.162-169.

Attoh-Okine, N. and Bowers, S. (2004). "Graphical Probability Models for Bridge Management", NSF-REU, University of Delaware.

Beadle, H.W.P., Harper, B., Maguire, G.Q. and Judge, J. (1997). "Location Aware Mobile Computing", Proc. IEEE/IEE International Conference on Telecommunications, (ICT'97), Melbourne, April, pp.1319-1324.

Beyh, S. and Kagioglou, M. (2004). "Construction Sites Communications Towards The Integration of IP Telephony", ITCon Vol.9 (2004), pp.325-344.

Bilmes, J.A. (1998). "A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models", International Computer Science Institute, Berkeley CA, TR-97-021, pp. 1-5.

Bürgy, C. (2002). "An Interaction Constraints Model for Mobile and Wearable Computer-Aided Engineering Systems in Industrial Applications", Submitted in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy, Department of

Civil and Environment Engineering, Carnegie Mellon University, Pittsburgh, Pennsylvania.

Czepiel, E. (1995). “Bridge Management Systems, Literature Review and Search”, <http://www.iti.northwestern.edu/publications/technical_reports/tr11.html>.

ESRI (2005). “ArcGIS9-Using ArcGIS 3D Analyst”, ESRI tutorial.

ESRI ArcView 3D Analyst (2006).
Web site: < http://www.esri.com/software/arcview/extensions/3danalyst/ >.

Federal Highway Administration (FHWA) (2000). “Bridge Inspector's Reference Manual”, Publication No. FHWA NHI 03-001.

Friedman-Hill, E. (2003). “Jess in Action: Java Rule-based Systems”, ISBN 1930110898.

Fujitsu (2005). Web site: <http://www.fujitsupc.com>.

Garrett, J.H. Jr., Bürgy, C., Reinhardt, J. and Sunkpho, J. (2002). “An Overview of the Research in Mobile/Wearable Computer-Aided Engineering Systems in the Advanced Infrastructure Systems Laboratory at Carnegie Mellon University”, Bauen mit Computern, Bonn, Germany. VDI Verlag GmbH, Duesseldorf, Germany.

Gordon, C., Akinci, B., Boukamp, F. and Huber, D. (2005). “Assessment of Visualization Software for Support of Construction Site Inspection Tasks Using Data Collected From Reality Capture Technologies”, Proceedings of the 2005 ASCE International Conference on Computing in Civil Engineering, July 12–15, 2005, Cancun, Mexico.

Handspring (2005). Web site: < http://web.palm.com/ >.

Hammad, A., Garrett, J.H. and Karimi, H. (2004a). “Location-Based Computing for Infrastructure Field Tasks”, In Karimi, H. and Hammad, A. (editors), “Telegeoinformatics: Location-Based Computing and Services,” CRC Press.

Hammad, A., Zhang, C., Hu,Y. and Mozaffari, E. (2004b). “Mobile Model-Based Bridge Lifecycle Management Systems”, Proc.,Conference on Construction Applications of Virtual Reality.ADETTI/ISCTE, Lisbon, pp. 109-120.

HP (2005). Web site: <http://www.hp.com/hpinfo/newsroom/press_kits/2005/mobility/products.html>.

Java Media Framework (JMF) website (2004). <http://java.sun.com/products/java-media/jmf/index.jsp>.

Jensen, F. (1996). “An Introduction to Bayesian Networks”, ISBN: 0-387-91502-8.

Kaplan, E.D. (1996). “Understanding GPS: Principles and Applications”, Artech House.

Karimi, H. and Hammad, A. (editors) (2004). “Telegeoinformatics: Location-Based Computing and Services” , CRC Press.

Kuladinithi, K., Timm-Giel, A. and Görg, C. (2004). “Mobile Ad-hoc Communications in AEC Industry”, ITcon Vol. 9, pp. 313-323.

Kuo, S.S., Clark, D.A. and Kerr, R. (1994). “Complete Package for Computer-Automation Bridge Inspection Process”, Transportation Research Record, No. 1442, pp. 115-127.

Laurini, R. and Thompson, A.D. (1992). “Fundamentals of Spatial Information Systems”, A.P.I.C. Series, Academic Press, New York, NY.

Leung, A. (1996). “Perfecting Bridge Inspecting”, Civil Engineering Magazine, pp.59-61.

Liu, D., Cheng, J., Law, K.H. and Wiederhold, G. (2002). “An Engineering Information Service Infrastructure for Ubiquitous Computing”, CIFE Technical Report #141, Stanford University.

Madanat, S., Mishalani, R. and Ibrahim, W.H.W. (1995). "Estimation of Infrastructure Transition Probabilities from Condition Rating Data", Journal of Infrastructure Systems, ASCE, Vol.1, No.2, pp. 120-125.

McCabe, B., AbouRizk. M.S. and Geobel, R. (1998). "Belief Network for Construction Performance Diagnostics", Journal of Computing in Civil Engineering, Vol. 12, No. 2, pp. 93-100.

MicroOptical web site (2005). <http://www.microopticalcorp.com/>.

Microsoft Windows XP Tablet PC Edition web site (2003). <http://www.microsoft.com/windowsxp/tabletpc/>.

Microvision web site (2005). <http://www.microvision.com/>.

Mills, T.H. and Wakefield, R.R. (2004). "Modernizing Bridge Safety Inspection With Process Improvement and Digital Assistance", Final Contract Report, Virginia Polytechnic Institute & State University.

Ministry of Transportation of Quebec (MTQ) (2004). "Manual de l'Usage du Système de Gestion des Structure SGS-5016", Bibliothèque Nationale du Québec, Gouvernerment du Québec.

MIT Wearable Computing Group (2005). Website <www.media.mit.edu/wearables>.

Mizuno, Y., Abe, M., Fujino, Y. and Abe, M. (2002). "Development of Interactive Support System for Visual Inspection of Bridges", International Forum on Infrastructure Maintenance, Tokyo, Japan, pp. 1-15.

Morcous, G. (2000). "Case-Based Reasoning for Modeling Bridge Deterioration", Doctoral thesis, Department of Building, Civil, and Environment Engineering, Concordia University, Montreal, Quebec, Canada.

National Bridge Inspection Standards (NBIS) (1996). “Code of federal regulations, No. 23 CFR650”, U.S. Government Printing Office, Washington, D.C., pp.238–240.

National Engineering Technology (NET) Corporation (1994), “Bridgit Bridge Management System: Technical Manual—Version 1.00.” NCHRP Project 12-28(2) A, National Cooperative Highway Research Program, Transportation Research Board, Washington, DC.

Netica website (2005). < http://www.norsys.com/>.

Oregon Department of Transportation (ODT) (2003).“Bridge Inspection Pocket Coding Guide”.

Ontario Structure Inspection Management System (OSIMS) (2000). “Ontario Structure Inspection Manual”, ISBN: 0-7794-0431-9.

Panasonic web site (2005). <http://www.ixbt.com/portopc/panas-cf-18.shtml>.

Peña-Mora, F. and Dwivedi, G.H., (2002). “Multiple Device Collaborative and Real time Analysis System for Project Management in Civil Engineering”, ASCE Journal of Computing in Civil Engineering, Vol.16, No.1, pp.23-38.

PJCCI (2004). The official website of Jacques Cartier and Champlain Bridges Incorporated. <http://www.pjcci.ca/>.

Pontis Bridge Management (2005). “Pontis Release 4.4 User’s Manual”, AASHTO, Washington, D.C. 20001.

Poole, D.L., Mackworth, A., and Goebel, R.G. (1998). “Computational intelligence: a logical introduction”, Oxford University Press, New York, N.Y.

Reinhardt, J., Akinci, B. and Garrett, J.H. (2004). “Navigational Models for Computer Supported Project Management Tasks on Construction Sites”, Journal of Computing in Civil Engineering, Vol.18, No.4, October 1, 2004, pp. 281-290.

Rhonda, M. (2005). “Investigating Insurance Claims Using ArcGIS 9 and a Tablet PC”, ArcUser, The magazine for ESRI Software Users. January-March, 2005.

Rivard, H., Fenves, S.J., and Gomez, N. (1998). “Case-Based Reasoning for Conceptual Building Design”, 1st International Conference on New Information technologies for Decision making in Civil Engineering, University of Quebec, Montreal, Quebec, October, pp. 355-366.

Rojas, E. (1995). “Reengineering the construction inspection process through information technology”, M.S. Thesis. University of Colorado at Boulder, Colorado.

Russell, S.J. and Norvig, P. (2003). “Artificial Intelligence: A Modern Approach”, (Second edition), ISBN: 0-13-790395-2.

Ryall, M.J. (2001). “Brdige Management”, ISBN 0 7506 5077X.

Siemens web site (2005). < http://www.siemens.ca/index.jsp >.

Sloth, M., Johnsen, T.H., Larsen, E.S., Rasmussen, E. and Skaanning, C. (2003). “Use of Probabilistic Approach as Decision Support during Condition Evaluation”, Second Gulf Conference on Roads, Abu Dhabi, pp. 2-16.

Stumpf, A., Liu, L.Y., Kim, C.S. and Chin, S. (1998). “Delivery and Test of the Digital Hardhat System at U.S. Army Corps of Engineers Fort Worth District Office”, USACERL ADP Report 99/16.

Sunkpho, J. (2001). “A Framework for Developing Field Inspection Support Systems”, Submitted in Partial Fulfillment of the Requirements for the Degree of Doctor of

Philosophy, Department of Civil and Environment Engineering, Carnegie Mellon University, Pittsburgh, Pennsylvania.

Sunkpho, J., Garrett, J., and McNeil, S. (2002). "A Framework for Field Inspection Support Systems Applied to Bridge Inspection", Proceedings of the 7th International Conference on the Applications of Advanced Technologies in Transportation, Cambridge MA, August 5-7, pp. 417-424.

Sunkpho, J., Garrett, J., and Smailagic, A. (2000). "Opportunities to Use Speech Recognition for Bridge Inspection", In the Proceedings of the 2000 ASCE Construction Congress, pp. 184-193, Orlando, FL.

Thompson, P.D., Ellis, R.D. and Merlo, T. (1998). "Implementation of Ontario Bridge Management System", Transportation Research Circular E-C049: 9th International Bridge Management Conference, IBMC03-028, pp. 112-127.

Thompson, P.D. and Shepard, R.W. (2000), "AASHTO Commonly-Recognized Bridge Elements", Report prepared for National Workshop on Commonly Recognized Measures for Maintenance, pp. 2-13.

Transport Canada (1998). "Transportation In Canada", Annual Report.

Trilon, Inc. (2004). Web site: <http://www.trilon.com/>.

Trimble (2005). Web site: <http://www.trimble.com/>.

Toshiba (2005). Web site: <http://www.toshiba.ca/>.

Via (2005). Web site: <http://www.via.com.tw>.

Walesh, A. and Gehringer, D. (2001). Java 3D, API Jump-start, Prentice Hall PTR.

WearableGroup at Carnegie Mellon University (2005).

Web site < http://www.wearablegroup.org/ >.

Xybernaut (2005). Web site <http://www.xybernaut.com>.

Zaki, A.R. and Mailhot, G. (2003). "Deck Reconstruction of Jacques Cartier Bridge Using Pre-cast Pre-stressed High Performance Concrete Panels", PCI Journal, September-October, pp. 2-15.

APPENDIX A: Bayes' rule and a BN calculation example

Basic Axioms of Probability:

The theory of probability we discuss below is based on certain fundamental assumptions or axioms. For every event *E* in a sample space *S*, there is a probability
$P(E) \geq 0$
Secondly, the probability of the certain event *S* is
$P(S) = 1.0$
Thirdly, for two events E_1 and E_2 that are mutually exclusive,
$P(E_1 \cup E_2) = P(E_1) + P(E_2)$

Two definitions:

Mutually exclusive events: If the occurrence of one event precludes the occurrence of another event, then the two events are mutually exclusive.
Collectively exhaustive events: Two or more events are collectively exhaustive if the union of all these events constitutes the underlying sample space.

Condition Probability:

The probability of an event may depend on the occurrence (or non-occurrence) of another event. If this dependence exists, the associated probability is a condition probability.

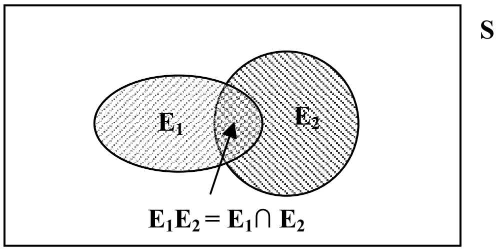


Figure A.1 Condition probabilities in E_1 and E_2

In the Figure A.1, the condition probability of E_1 assuming E_2 has occurred, denote $P(E_1|E_2)$. Hence, we obtain the condition probability of E_1 given E_2 as:

$$P(E_1|E_2) = \frac{P(E_1E_2)}{P(E_1)} \qquad \textit{Equation (A.1)}$$

E_2 given E_1 as

$$P(E_2|E_1) = \frac{P(E_2E_1)}{P(E_1)} \qquad \textit{Equation (A.2)}$$

From Equation A.1 and A.2, the probability of the joint event E_1E_2 is

$$P(E_1E_2) = P(E_1|E_2)P(E_2) = P(E_2|E_1)P(E_1) \qquad \textit{Equation (A.3)}$$

Total Probability Theorem:

Assume that the probability of an event *A* is always accompanied by the occurrence of other events E_n, $n = 1, 2, \ldots, i$ (Figure A.2). That means the probability of *A* will depend on which of the events E_n has occurred. Therefore, the probability of *A* will be an expected probability.

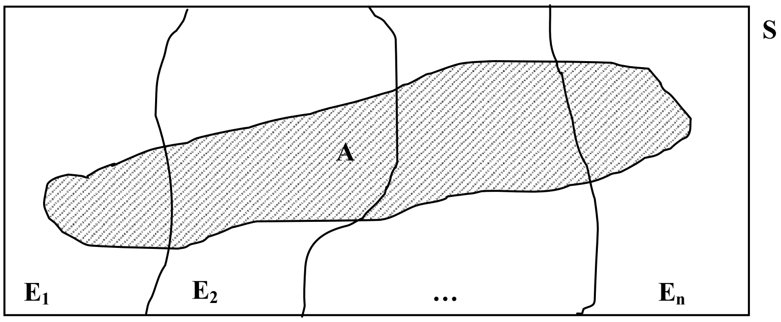


Figure A.2 Diagram with events A and $E_1, E_2, \ldots, E_n$

Consider *n* mutually exclusive and collectively exhaustive events $E_1, E_2, \ldots, E_n$; that is, $E_1 \cup E_2 \cup \ldots \cup E_n = S$. Then if *A* is an event also in the sample space, we have

$A = AS = A\ (E_1 \cup E_2 \cup \ldots \cup E_i) = AE_1 \cup AE_2 \cup \ldots \cup AE_n$

Where $AE_1, AE_2 \ldots AEn$ are also mutually exclusive, then

$P\ (A) = P(AE_1) + P(AE_2) + \ldots + P(AE_n)$ *From Equation A.3*

$P\ (A) = P(A|E_1)\ P(E_1) + P(A|E_2)\ P(E_2) + \ldots + P(A|E_n)\ P(E_n)$ *Equation (A.4)*

Bayes' Theorem:

Based on total probability theorem, if the event *A* occurred, what is the probability that a particular E_n also occurred (reverse probability)?
Apply Equation A.3 to the joint event AE_n, we have,

$P(A|E_n)\ P(E_n) = P(E_n|A)\ P(A)$; *therefore*

$P(E_n|A) = \dfrac{P(A \mid E_n)P(E_n)}{P(A)}$; *(Bayes' rule)*

If P(A) is expanded using the total probability theorem (Equation A.4), Bayes' rule becomes

$$P(E_n|A) = \frac{P(A \mid E_n)P(E_n)}{\sum_{j=1}^{i} P(A \mid E_j)P(E_j)} \qquad \textit{Equation (A.5)}$$

A BN calculation example:

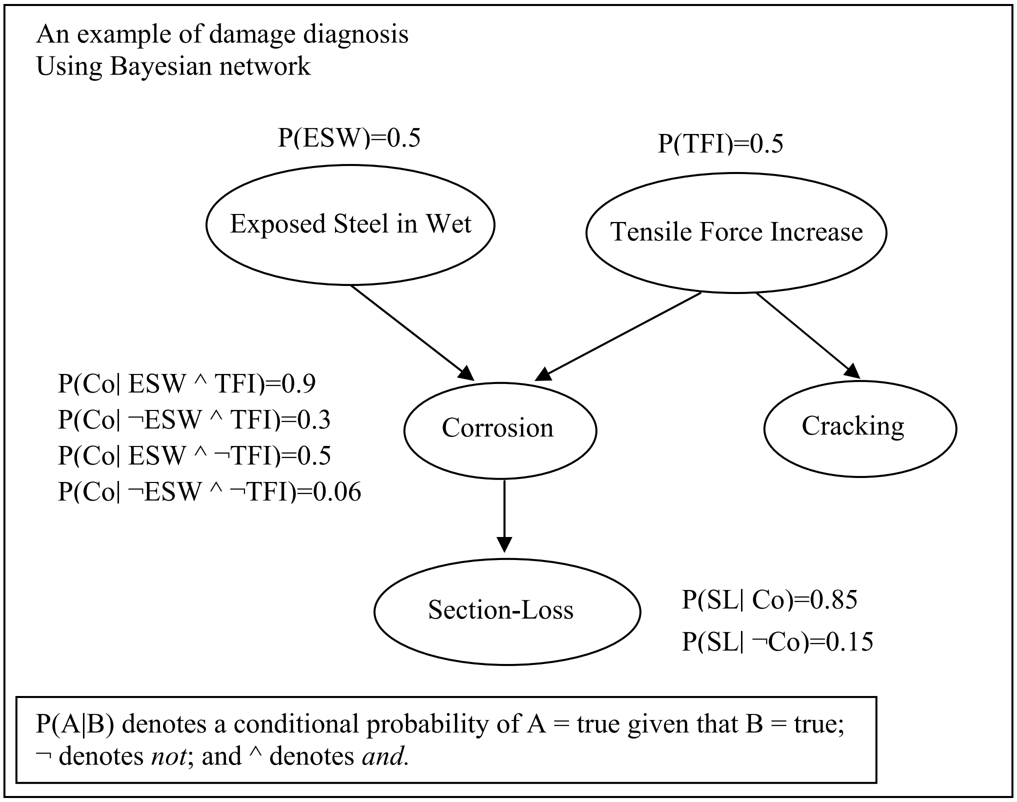


Figure A.3 An example of damage diagnosis using BN

Corrosion (*Co*), Section-Loss (*SL*), and Cracking (*Cr*) are the major defects which are usually occurred on bridge steel components. In the simple example (Figure A.3), Exposed Steel in Wet (*ESW*) and Tensile Force Increase (*TFI*) are considered as the critical causes that will result in *Co* and *Cr*, and corrosion will be developed to *SL* when the deterioration degree increasing. In the figure, partial probabilities are available based on previous data statistic and experience acquirement. Assume the inspector wants to know the probability of *TFI* under the given condition of *SL, P (TFI | SL) =0.622*. The calculations depending on Bayes' rule (Equation A.5) will give the answer:

$$\textbf{\textit{P (TFI| SL)}} = \frac{P(SL \mid TFI)P(TFI)}{P(SL)} \; ; \qquad \textit{Equation (A.6)}$$

Where in Equation A.6,

*P (SL|TFI) =P (SL|TFI ^ Co)*P (Co|TFI) + P (SL|TFI ^ ¬Co)*P (¬Co|TFI); Equation (A.7)*

Because SL and TFL are d-separate by Co, So Equation A.7 will be changed into

*P (SL|TFI) =P (SL|Co)*P (Co|TFI) + P (SL|¬Co)*P (¬Co|TFI); Equation (A.8)*

*In (A.8), P (Co|TFI) =P (Co|TFI ^ ESW)*P (ESW) + P (Co|TFI ^ ¬ESW)*P (¬ESW);*

*P (SL) = P (SL|Co)*P (Co) + P (SL|¬Co)*P (¬Co) ; Equation (A.9)*

*In (A.9), P (Co) = P (Co|ESW ^ TFI)*P (ESW)*P (TFI) +*

*P (Co|¬ESW ^ TFI)*P (¬ESW)*P (TFI) +*

*P (Co|ESW ^ ¬TFI)*P (ESW)*P (¬TFI) +*

*P (Co|¬ESW ^ ¬TFI)*P (¬ESW)*P (¬TFI);*

Using the available values, we can get:

P (TFI) = 0.5;

*P (Co|TFI) = 0.9*0.5 + 0.3*0.5 = 0.60;*

*P (SL|TFI) = 0.85*0.60 + 0.15*0.40 = 0.57;*

P (Co) = 0.225+0.075+0.125+0.015=0.44;

*P (SL) = 0.85*0.44 + 0.15*0.56 = 0.458;*

***P (TFI|SL**) = 0.57*0.5/0.458 =0.622.*

Therefore, the inspector may conclude that the probability of Tensile Force Increase is 62.2% when Section-Loss was detected.
The same method can be used to calculate the likelihood of TFI under the conditions of Corrosion and Cracking, P (TFI | Co ^ Cr).

APPENDIX B: Expectation-Maximization (EM) algorithm

Maximum likelihood

A probability function p(X|Θ) is defined that is governed by a set of parameters of variables Θ. A set of data N is also defined (i.e., observed values), supposedly drawn from this distribution, i.e., X = {x_1,…,x_N}. We assume that these data are independent and identically distributed with distribution *p*. Therefore, the resulting probability for the samples is given by using Equation B.1:

$$p(X \mid \Theta) = \prod_{i=1}^{N} p(x_i \mid \Theta) = L(\Theta \mid X) \qquad \text{Equation (B.1)}$$

The function L(Θ|X) is called the likelihood of the parameters when the data are given. The likelihood is thought of as a function of the parameters Θ where the data X is fixed.

Basic theory of the EM algorithm

As before, the data X is observed and is generated by some distribution. X is called the incomplete data. We assume that a complete data set, Z = (X, Y), exists and also specify a joint probability function (Equation B.2). In our case (e.g., missing data values in samples of a distribution), we must assume a joint relationship between the missing and observed values.

$$p(Z \mid \Theta) = p(X, Y \mid \Theta) = p(Y \mid X, \Theta)p(X \mid \Theta) \qquad \text{Equation (B.2)}$$

With this new probability function, a new likelihood function, L(Θ|Z) = L(Θ|X,Y) = p(X, Y|Θ), can be defined, called the complete data likelihood. This function is a random variable since the missing information Y is unknown. The EM algorithm first finds the expected value of the complete data log likelihood, log p(X, Y|Θ), with respect to the unknown data Y given the observed data X and the current parameter estimates. That is, we define the E-step (Equation B.3).

$$Q(\Theta, \Theta^{(i-1)}) = E\left[\log p(X, Y \mid \Theta) \mid X, \Theta^{(i-1)}\right] \qquad \text{Equation (B.3)}$$

Where $\Theta^{(i-1)}$ are the current parameters estimates that we used to evaluate the expectation, and Θ are the new parameters that will be ultimately optimized in an attempt to maximize the likelihood. X and $\Theta^{(i-1)}$ are constants, Θ is a normal variable that we wish to adjust, and Y is a random variable.

The second step of the EM algorithm, M-step, finds the new values of the parameters that maximize the log likelihood of the data, given the expected values of the parameters (Equation B.4). These E- and M-steps are repeated as necessary. Each iteration is guaranteed to increase the log-likelihood and the algorithm is guaranteed to converge to a local maximum of the likelihood function.

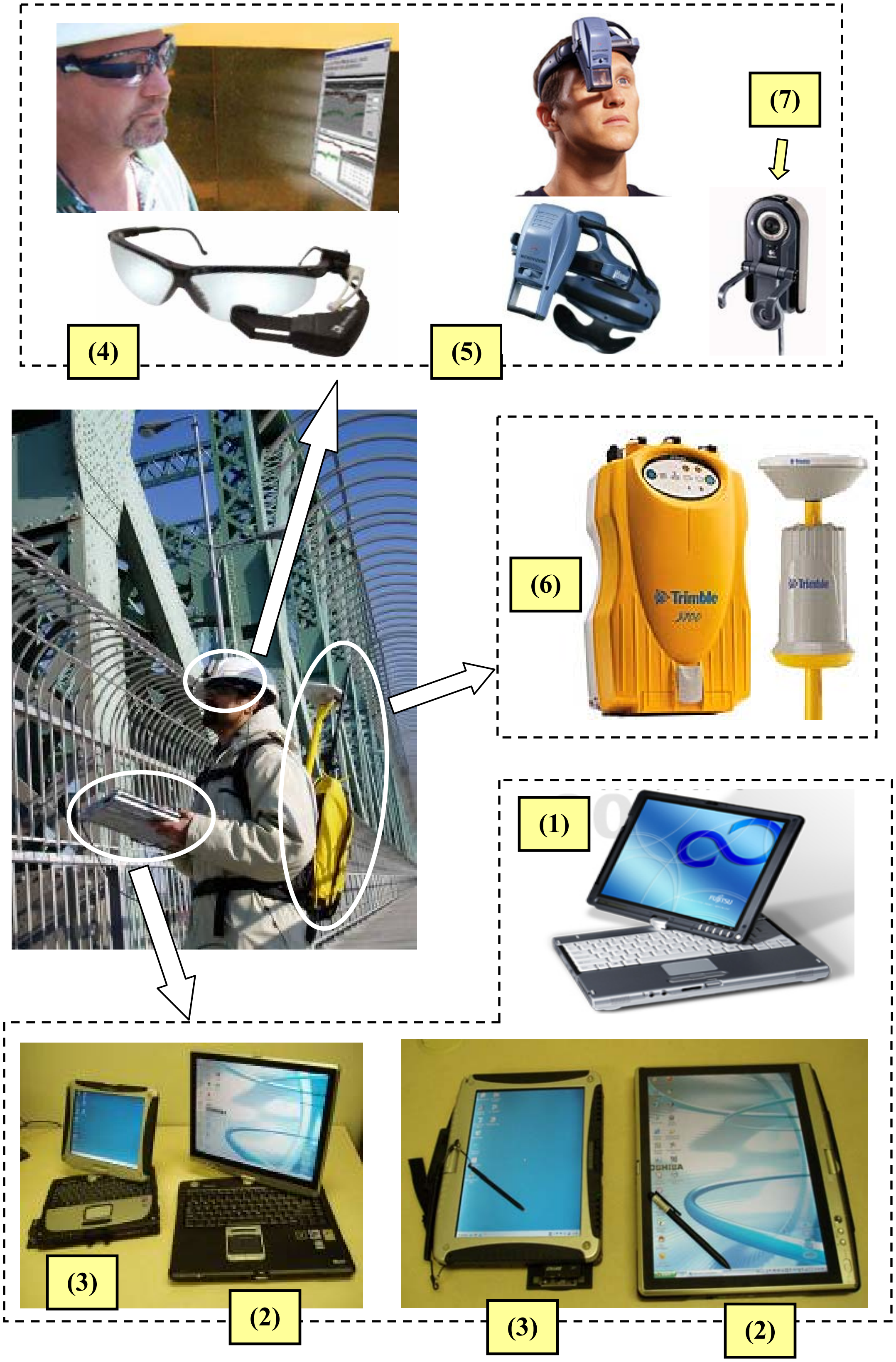
$$\Theta^{(i)} = \underset{\Theta}{\operatorname{argmax}}\, Q(\Theta, \Theta^{(i-1)}) \qquad \text{Equation (B.4)}$$

APPENDIX C: Structure types and numbers in Quebec (2004)

Category	Structure type	Material	System	Frequency	Total
Culvert	11	R.C.	Solid slab	16	1141 **11.1%**
	12	R.C.	Rigid frame	0	
	13	R.C.	Box section	499	
	14	R.C.	Circular section	2	
	15	Steel	Circular section	141	
	16	Thermoplastic	Circular section	1	
	17	Steel	Elliptic section	25	
	18	Steel	Curved closed section	285	
	19	R.C.	Arc	106	
	20	Steel	Arc	66	
Slab Bridge	31	R.C.	Solid slab	615	1883 **18.2%**
	32	P.C.	Solid slab	39	
	33	R.C.	Hollow slab	153	
	34	P.C.	Hollow thick slab	22	
	35	R.C.	Portal frame	496	
	36	R.C.	Portal frame below ground	394	
	37	P.C.	Portal frame	1	
	38	R.C.	Rigid frame	154	
	39	P.C.	Rigid frame	9	
Beam Bridge	41	R.C.	Rectangular beams	1430	5984 **57.9%**
	42	P.C.	Precast beams	849	
	43	P.C.	Rectangular beams	210	
	44	Steel	I-beams under R.C. slab	782	
	45	R.C.	I-beams under wood slab	2420	
	46	Wood	Rectangular beams	30	
	47	R.C.	Portal frame	27	
	48	R.C.	Portal frame below ground	1	
	49	Steel	Portal frame	0	
	50	R.C.	Rigid frame	50	
	51	Steel	Rigid frame	8	
	52	Steel	Covered with concrete	177	
Box-Girder Bridge	56	R.C.	Two boxes	57	146 **1.4%**
	57	P.C.	One box	52	
	58	Steel	Two boxes	37	
Truss Bridge	61	Steel	Through N truss	104	313 **3.0%**
	62	Steel	Intermediate N truss	3	
	63	Steel	Through W truss	73	
	64	Steel	Through bailey truss	12	
	65	Steel	Deck N truss	35	
	66	Wood	Triangular truss	3	
	67	Steel	Covered truss	83	
Arch Bridge	71	R.C.	Through arch	1	74 **0.7%**
	72	Steel	Through arch	15	
	73	R.C.	Intermediate arch	1	
	74	Steel	Intermediate arch	0	
	75	R.C.	Deck arch	52	
	76	Steel	Deck arch	5	
Cabled Bridge	81	Any	Suspension bridge	5	11 **0.1%**
	82	Any	Cabled-stayed bridge	6	
Others	91	Any	Movable bridge	2	783 **7.6%**
	92	Any	Foot bridge	0	
	94	Any	Tunnel	12	
	95	Any	Signals support	0	
	96	Any	Platform	0	
	97	Any	Retaining wall	726	
	98	Any	Pumping station	30	
	99	Any	Others	13	

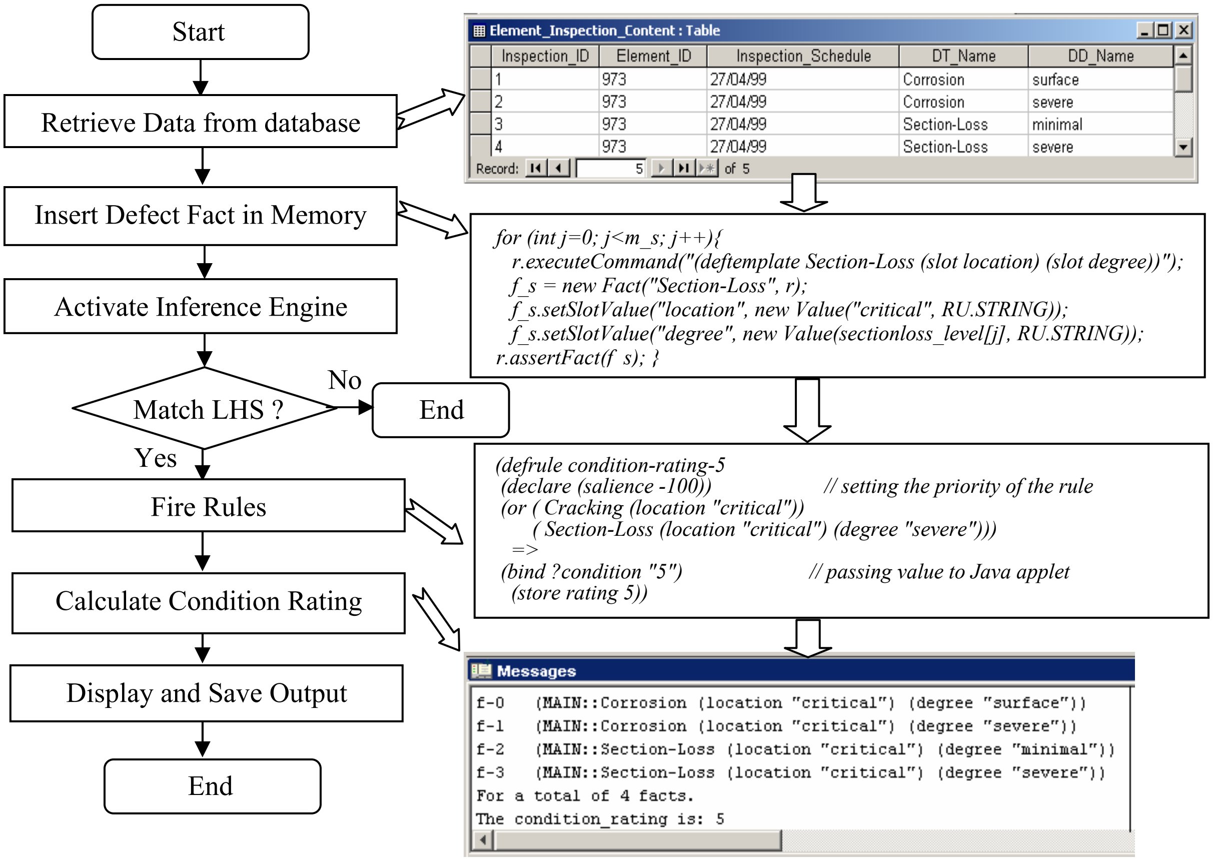
Total Number of Structures: 10335 **100%**

APPENDIX D: Equipment used in the prototype system



Type	No.	Device	Feature
Tablet PC	(1)	Fujitsu LifeBook T4020B Tablet PC	Processor: Intel® Pentium® M Processor 760 GHz, Memory: 1 GB, Hard Disk Driver: 80GB, Display: 12.1" XGA TFT (indoor/outdoor viewable), Pointing Device: Fujitsu Tablet Pen, Battery Life: Standard Lithium ion, 5.25 hours
	(2)	Toshiba Tecra M4 Tablet PC	Processor: Mobile Intel® Pentium® M1.86GHz, Memory: 1GB DDR2, Hard Disk Driver: 80GB, Display: TFT Active Matrix colour LCD display w digitizer, Pointing Device: Touchpad + Accupoint, Toshiba Tablet Pen, Battery Life: 3.5 hours
	(3)	Panasonic Toughbook-18	Processor: Intel® Pentium® M Processor ULV 753, 1.20GHz, Memory: 512MB+256MB SDRAM standard, Hard Disk Driver: 40GB Display: (XGA) transmissive daylight readable TFT Active Matrix Color LCD, Pointing Device: Pressure sensitive touchpad with vertical scrolling support Battery Life: 7.0 hours
HMD	(4)	Micro Optical SV-6 PC Viewer	Display Format: 640 x 480 pixels, 60 Hz refresh rate, Display Color: 18-Bit (262,144 colors), Field of View: Approximately 16º horizontal, 20º diagonal, Input Signal: VGA, SVGA, XVGA, 60-75 Hz, Focus Range: Adjustable focus from 2 to15 feet, Continuous Operation: 5 hours on full charge
	(5)	Microvision Nomad ND2000	Display Format: SVGA 800x600 pixels, 60 Hz refresh rate, Display Color: Monochrome Red, Field of View: 17.25º horizontal, 23º diagonal, Input Signal: SVGA format, Focus Range: Adjustable from 1 feet to infinity, Continuous Operation: 8 hours on full charge
GPS	(6)	Trimble 5700 RTK GPS	Dual-frequency, RTK, GPS and WAAS/EGNOS receiver, Portable Zephyr antenna, 128 MB of compact flash memory, USB port to transfer data to a PC at speeds of more than 1mb/s.
Digital Camera	(7)	Logitech-3000	Resolution: 640 x 480 (Image), 640 x 480 @ 15fps (Video), Features: Built-in microphone , Connectivity: USB

APPENDIX E: Condition rating example of interaction between Java and JESS


Start
Retrieve Data from database
Insert Defect Fact in Memory
Activate Inference Engine
Match LHS ?
No
End
Yes
Fire Rules
Calculate Condition Rating
Display and Save Output
End
Element_Inspection_Content : Table
Record: 5 of 5
Messages

Inspection_ID	Element_ID	Inspection_Schedule	DT_Name	DD_Name
1	973	27/04/99	Corrosion	surface
2	973	27/04/99	Corrosion	severe
3	973	27/04/99	Section-Loss	minimal
4	973	27/04/99	Section-Loss	severe

```
for (int j=0; j<m_s; j++){
   r.executeCommand("(deftemplate Section-Loss (slot location) (slot degree))");
   f_s = new Fact("Section-Loss", r);
   f_s.setSlotValue("location", new Value("critical", RU.STRING));
   f_s.setSlotValue("degree", new Value(sectionloss_level[j], RU.STRING));
r.assertFact(f_s); }
```

```
(defrule condition-rating-5
 (declare (salience -100))              // setting the priority of the rule
 (or ( Cracking (location "critical"))
      ( Section-Loss (location "critical") (degree "severe")))
   =>
 (bind ?condition "5")                  // passing value to Java applet
   (store rating 5))
```

```
f-0   (MAIN::Corrosion (location "critical") (degree "surface"))
f-1   (MAIN::Corrosion (location "critical") (degree "severe"))
f-2   (MAIN::Section-Loss (location "critical") (degree "minimal"))
f-3   (MAIN::Section-Loss (location "critical") (degree "severe"))
For a total of 4 facts.
The condition_rating is: 5
```

APPENDIX F: Software requirements and installation guide of the prototype system

Software requirements:

(1) Borland JBuilder 2005 Enterprise: used to develop the prototype system of BMS;
(2) MS Access (MS Access XP): used to store the lifecycle data of the bridge;
(3) ArcGIS (ESRI 2004): used to develop GIS application;
(4) Netica: used to develop the prototype of learning-based BN model;
(5) Windows XP: used as the operation system.

Installation guide:

1. Copy four folders to corresponding driver and change the associated code in the project to match the driver path. The contents in these folders include:
   - *Currproject* or *infra_project* (The folder includes all codes of our project)
   - *Javasoft* (The folder includes all libraries which are required in our project)
   - *Bridgere* (The folder includes all 2d information)
   - *BridgeResources* (The folder includes all 3d information and models)

2. Click Start->Control Panel-> Administrative Tools->Database Source. And add ODBC data source as below:
   - Microsoft Access Driver: Name: bridge, Location: C:\ BridgeResources\db1.mdb

   (No password for data source is required, so just leave the password as blank.)

3. Launch Jbuilder, open the project.jpx file. Then click the menu of Jbuilder: Project->Project Properties. Click the tab ”Required Libraries”. Then edit or add the path of libraries as below:

VRML97 (VRML File Loader API)

- Download the library from: https://j3d-vrml97.dev.java.net/ and install it. Or you can get it from the path: D:/javasoft/loaders/vrml97.jar
- Edit or add the vrml97.jar file to the path of VRML97 library.

Jess61p4 (Java Expert System Shell API) - Version 6.1

- Download the library from: http://herzberg.ca.sandia.gov/jess/ and install it. Or you can get it from the path: D:/javasoft/Jess61p4/jess.jar.
- Edit or add the jess.jar file to the path of Jess61p4 library.

Jdk3D (Java development Kit 3D) - Version 1.3.1

- Download the library from: http://java.sun.com/products/java-media/3D/downloads/index.html and install it. Or you can get it from the path: D:/javasoft/ JRE/1.3.1_09/lib/ext
- Edit or add all jar files under the Jdk3d directory to the path of Jdk3D library.

Javacomm (Java Communications API) - Version 3.0

- Download the library from: http://www.sun.com/download/products.xml?id=43208d3d and install it. Or you can get it from the path: D:/Javasoft/commapi/
- Edit or add the comm.jar file under the commapi directory to the path of Javacomm library .

DXFLoader (DXF file loader API) - Version 1.0

- Download the library from: http://www.johannes-raida.de/index.htm?cadviewer and install it. Or you can get it from the path: D:/Javasoft/DxFloader/
- Edit or add the dxfloader.jar file under the dxfloader directory to the path of DXFLoader library.

MOJ (MapObject Java API) - Version 2.1

- Download the library from: http://www.esri.com/software/mojava/ and install it.
- Edit or add all jar files under the directory C:/ESRI/MOJ21/lib to the path of MOJ library. Please also add the tutsource.jar and tutorial.jar files that are available at the directory MOJ21\Samples\Tutorial.

Netica (Netica Java API) - Version 2.17

- Download the library from: http://www.norsys.com/netica-j.html#download and install it. Or you can get it from the path: D:/Javasoft/ NeticaJ_Win/bin
- Edit or add the path of the directory NeticaJ_Win/bin to the path of Netica library.

JMF (Java Media Framework API) - Version 2.1.1

- Download the library from: http://java.sun.com/products/java-media/jmf/2.1.1/download.html
  and install it. Or you can get it from the path: D:/Javasoft/ jmf211e_scst/build/win32/lib
- Edit or add the path of the directory jmf211e_scst/build/win32/lib to the path of JMF library.

JARToolkit (Java ARToolkit API) - Version 2.0

- Download the library from: http://jerry.c-lab.de/jartoolkit/ and install it. Or you can get it from the path: D:\JavaSoft\JARToolkit Dlls
- JARToolkit need the dll files as below:
  - JARFrameGrabber.dll
  - JARToolkit.dll
  - JARVideo.dll
  - libARvideo.dll
  - libARvideod.dll
  - msvcr70.dll

Make sure you put these dll files in your JARToolkit Dlls directory, then click start->Control Panel->System->Advanced->Environment Variables, please add the path of JARToolkit Dlls directory to the option”path” of User Variables.

Notes:

* When you use a different account in the same computer, you have to separately set the path of libraries for every account. It means you cannot just set the libraries for all accounts at the same time.

* If your code cannot be compiled following the above instructions, please carefully check the installation instruction. If the error information is about Java 3D, that means your computer does not have Java 3D. You can get the Java 3D package from the Java 3D folder.

Table E.1 Summary of libraries used in the prototype system

Library	Description	Source	Version
VRML97	VRML File Loader API	https://j3d-vrml97.dev.java.net/ D:/javasoft/loaders/vrml97.jar	
Jess61p4	Java Expert System Shell API	http://herzberg.ca.sandia.gov/jess/ D:/javasoft/Jess61p4/jess.jar	6.1
JDK 3D	Java 3D API	http://java.sun.com/products/javamedia/3D/downloads/index.html D:/javasoft/ JRE/1.3.1_09/lib/ext	1.3.1
Javacomm	Java Communication API	http://www.sun.com/download/products.xml?id=43208d3d D:/Javasoft/commapi/	3.0
DXFLoader	DXF File Loader API	http://www.johannes-raida.de/index.htm?cadviewer D:/Javasoft/DxFloader/	1.0
MOJ	MapObject Java API	http://www.esri.com/software/mojava/ C:/ESRI/MOJ21/lib	2.1
Netica	Netica Java API	http://www.norsys.com/netica-j.html#download D:/Javasoft/ NeticaJ_Win/bin	2.17
JMF	Java Media Framework API	http://java.sun.com/products/java-media/jmf/2.1.1/download.html D:/Javasoft/jmf211e_scst/build/win32/lib	2.1.1
JARToolKit	Java ARToolKit API	http://jerry.c-lab.de/jartoolkit/ D:\JavaSoft\JARToolkit Dlls	2.0

APPENDIX G: List of publications

a. Articles accepted in or submitted to refereed journals

(1) Hammad, A., Zhang, C., **Hu, Y.,** and Mozaffari, E. (accepted in June, 2005). "Mobile Model-Based Bridge Lifecycle Management Systems", *Journal of Computer-Aided Civil and Infrastructure Engineering*, Blackwell Publishing.

(2) Hammad, A., **Hu, Y.,** and Zhang, C. (Accepted with minor modifications in February, 2006). "Location-Based Mobile Computing for Bridge Inspection", *Advanced Engineering Informatics*, Elsevier.

(3) **Hu, Y.** and Hammad, A. (Submitted in 2006). "Bridge Inspection DSS Using Learning-Based Belief Network", *Journal of Infrastructure Systems, ASCE.*

b. Articles in refereed conference proceedings

(4) Hammad, A., Zhang, C**.**, **Hu, Y.,** and Mozaffari, E. (2004). "Mobile Model-Based Bridge Lifecycle Management Systems", *In Proceedings of Conference on Construction Application of Virtual Reality*, ADETTI/ISCTE, September, 2004, Lisbon, pp. 109-120.

(5) **Hu.Y.** and Hammad, A. (2005). "Location-based Mobile Bridge Inspection Support System", *1st CSCE Specialty Conference on Infrastructure Technologies, Management and Policy,* June, 2005, Toronto, Ontario, Canada.

(6) Hammad, A., **Hu, Y.,** and Zhang, C. (2005). "Mobile Model-Based Bridge Inspection System", 11th Annual Urban Infrastructure Week, November, 2005, Montreal.

(7) **Hu, Y.** and Hammad, A. (2006). "Learning-based Bridge Inspection Decision-Support System", *7th International Conference on Short & Medium Span Bridges (Repair, Maintenance and Evaluation)*, August, 2006, Montreal, Quebec, Canada.

(8) **Hu, Y.** and Hammad, A. (2006). "Location-Based Mobile Bridge Inspection System", *7th International Conference on Short & Medium Span Bridges (Management of Bridge Asset)*, August, 2006, Montreal, Quebec, Canada.

(9) Hammad, A. and **Hu, Y.** (2006). "Location-Based Mobile Bridge Inspection-Support System", *11th International Conference on Computing in Civil and Building Engineering*, ICCCBE-XI, June, 2006, Montreal.