1st CSCE Specialty Conference on Infrastructure Technologies, Management and Policy
1ere conférence spécialisée sur la gestion, les politiques et les technologies des infrastructures

Toronto, Ontario, Canada
June 2-4, 2005 / *2-4 juin 2005*

Location-based Mobile Bridge Inspection Support System

Y. Hu[1], A. Hammad[2]
1. Department of Building, Civil & Environmental Engineering, Concordia University, Montreal, Qc, Canada
2. Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Qc, Canada

ABSTRACT: Location-Based Computing (LBC) is an emerging discipline integrating geoinformatics, telecommunications, and mobile computing technologies. LBC utilizes geoinformatics and tracking methods in a distributed real-time mobile computing environment. 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. In this paper, we propose a new LBC approach to support the data collection activities of bridge inspectors. The proposed prototype system is equipped with a 3D detailed model of the bridge, and all the inspection results are registered on the 3D model. The system navigates the inspector to the locations he/she needs to inspect, provides information about the results of previous inspections as augmentation of the 3D model, and allows the inspector to add new information and to specify the location of a new defect simply by clicking on the point of the model where the defect has been found and then selecting the type and level of the defect from available menus. Furthermore, the system has a rule-based expert system that is used for data analysis and probabilistic diagnosis based on the location and the context of the inspection tasks in order to give the inspector suitable support. The system is implemented in Java language and a case study about Jacques Cartier Bridge is demonstrated.

1. INTRODUCTION

Bridge Management Systems (BMSs) are used to manage information of bridges and to assure their long-term health under budgetary constraints (Czepiel, 2004; 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. Among various tasks of bridge management, field inspection is essential in 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 on all of the individual elements of a bridge and enter this data into a BMS database.

Recent BMSs are introducing new information technologies to integrate multimedia information and to facilitate mobile data collection and manipulation using pen-based tablet PCs (Fujitsu, 2004) and Personal Digital Assistants (PDAs). 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. 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. In addition, 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).

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). This application leads to cost savings in terms of personnel, inspection time, and data processing. 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. In addition, research for developing an improved decision-support system for bridge management has been proposed. Mizuno et al. (2002) discussed the assessment of defect rating of bridge members by implementing a rule-based expert system. Sloth et al. (2004) developed DecisionWorks software to analysis bridge defect causes based on Bayesian Networks.

Location-Based Computing (LBC) is an emerging discipline focused on integrating geoinformatics, telecommunications, and mobile computing technologies (Beadle et al., 1997) (Karimi and Hammad, 2004). Based on LBC combined with a 3D model, bridge elements which are registered according to their positions in a spatial database can be located using suitable positioning devices, and defects on specific elements can be recorded more efficiently and accurately which will directly affect the bridge structure evaluation and maintenance strategy decisions. The second author 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 (Hammad et al., 2004a). This paper builds on the experience gained from the testing of LBC-Infra to further develop LBC-Infra and link it to a decision-support system.

In this paper, the proposed system is equipped with a 3D detailed model of the bridge and all the inspection results are registered on the 3D model. The system implements a friendly Graphical User Interface (GUI) for the convenience of data input by an inspector equipped with a wearable computing device. The system navigates the inspector to the locations he/she needs to inspect, provides information about the results of previous inspections as augmentation of the 3D model, and identifies the inspection sequences and the defect locations with navigation guidance. After that the system allows the user to add defect inspection information and to specify the location of a new defect simply by picking the element of the model near the location where the defect has been found, and then selecting the type and level of the defect from available menus. In addition, the system provides functions for element condition assessment and probabilistic diagnosis of defect causes using an expert system. The diagram in Figure 1 illustrates the structure of components and techniques used in the system. The software component contains three major parts, which are process control, data collection, and data analysis with their corresponding supporting techniques. The device component includes mobile technologies with the configuration of selected equipment to adapt to different situations. Both components are integrated to construct the system and are applied in bridge inspection for the quality improvement of BMSs. These components will be discussed in the following sections.

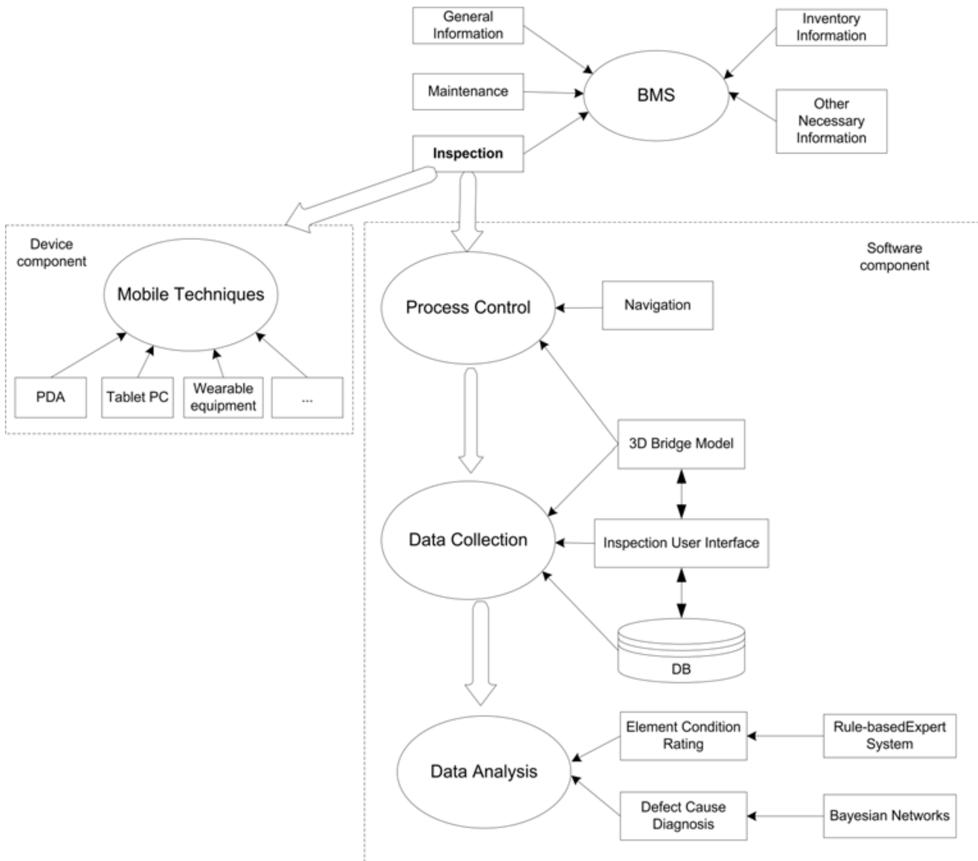

Figure 1. General Structure of Proposed System

2. PROPOSED APPROACH

2.1 Mobile computing techniques

Mobility is a basic characteristic of field tasks. A bridge 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. The application of mobile computing techniques can facilitate the inspector's activities allowing him to concentrate on the details of inspection tasks. These techniques include portable PCs, PDAs, wearable computers, Head-Mounted Displays (HMDs), digital cameras, wireless communications, speech and handwriting recognition, and dispersed collaboration.

2.2 Combination of 3D model and location-based application

Location-based bridge inspection is based on 3D bridge model. With the integration of the 3D model, several tools can be developed for supporting bridge inspection, such as visualization, simulation, and analysis. The inspector can navigate the 3D bridge model from various perspectives (Reinhardt et al., 2004) and select a bridge element and add defect location by picking that element at the approximate location of the defect. Based on the navigation model and picking behavior, the on-site inspection scenario can be simulated, and the collected data would eliminate the need to draw sketches as is usually required in present inspection practice.

2.3 Inspection user interface for facilitating data collection

Bridge inspection GUI should facilitate the interaction with the 3D virtual environment 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 sophisticated GUI will effectively guarantee the reliability of data collection which will ultimately affect the bridge structure evaluation and maintenance strategy decisions.

2.4 Element condition rating assessment using rule-based expert system

During an inspection, an attempt is made to determine the condition of an element based on the subjective opinion of a qualified expert. For each bridge element, the inspector records the defect attributes on that element depending on visual observations, then evaluates individual element condition rating based on the information of all the defects. The element condition ratings are later used in conjunction with other factors to calculate the load carrying capacity and sufficiency rating which is further used to decide on the maintenance strategy. As one of the critical values of bridge inspection, the element condition rating is determined based on the expertise of the inspector when the specific defect is detected. Therefore, this determination has some subjective factor. Using rule-based expert systems which maintain a large collection of rules and facts, condition rating can be inferred immediately from inspection input through activating the matching facts. Moreover, expert systems are intended to support engineers in automating other bridge management activities.

2.5 Defect cause probabilistic diagnosis using probabilistic methods

It is important to allow inspectors to understand the cause of observed defects during a visual inspection. When a defect is taking place, the most probable cause should be found and appropriate measures need to be taken to prevent further deterioration. Because the cause of a defect has a combination of factors, it is not easy even for a skilled inspector to determine which factor is the most probable. Application for probabilistic methods, such as Bayesian Networks (Jensen, 1996), can help in the diagnosis and forecasting of the causes of defects.

3. DECISION SUPPORT ASPECTS OF PROPOSED APPROACH

Bridge inspection is a knowledge-intensive process. A qualified inspector must have professional training and possess sufficient practical experience. As those personnel retire, there is a significant experience gap being created. Therefore, it is of great interest to develop decision-support systems using rule-based expert systems and probabilistic analysis using Bayesian Networks. The following subsections give a brief description of these methods.

3.1 Rule-based expert systems

The architecture of a rule-based expert system contains a knowledge base and an inference engine. The basic units represented in rule-based expert systems are the rules. Rules can be fired through matching the asserted facts from the knowledge base using the inference engine. Two types of inferences are common in rule-based expert system: 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) also can be used in rule-based expert systems to describe linguistic information that are imprecise and ambiguous (e.g., the defect level can be described as severe, moderate or minor).

3.2 Bayesian Networks

Bayesian Networks are directed acyclic graphical models combined with probabilities that follow the rules of the probability theory. 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 represent random variables, and the relationships 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 Bayesian Networks is using Bayes' rule for conditional probabilistic inference. Equation [1] gives a basic description of Bayes' rule. If $B_1, B_2, \cdots, B_k$ are the possible scenarios with an effect on the event A,

$$[1] \qquad P\,(B_j \mid A) = \frac{P(B_j \cap A)}{P(A)} = \frac{P(A \mid B_j) \cdot P(B_j)}{\sum_{i=1}^{k} P(A \mid B_i) \cdot P(B_i)}$$

where $P\,(B_j \mid A)$ is the conditional probability of B_j given that all we know is A. Bayesian Networks are used for diagnosing real world problems where uncertainty and incomplete data exist. Taking advantage of this method, defect cause probabilistic analysis is applied in the proposed bridge inspection system.

4. PROTOTYPE SYSTEM AND CASE STUDY

To demonstrate the feasibility and usefulness of the proposed methodology, a prototype system is developed and discussed in detail in this section. The prototype system is built using Java language and integrating a 3D bridge model, object-relational database, and an expert system (Hammad et al., 2004b). The 3D model 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. 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). In the prototype, inspection GUI, Java Media Framework (JMF) (JMF, 2004), and a hyperlink function are integrated to implement the bridge inspection application. The software of the prototype system also integrates several mobile hardware technologies, such as a pen-based tablet PC (Fujitsu, 2004), digital camera, and wireless communications.

In the prototype system, 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 2 illustrates how to achieve interaction between a Java applet and JESS inference engine through asserting facts from the applet into the working memory of JESS to activate the inference engine. Using Rete algorithm to process rules, JESS is an efficient mechanism for solving the difficult many-to-many facts matching problem. The expert system is used to support inspectors in making decisions based on the result of data analysis. JESS is used to evaluate condition rating after comparing all the available data on individual elements from the database. A Bayesian Network is designed to deduce the probability of defect cause so that adequate measures are taken to maintain the bridge in good condition.

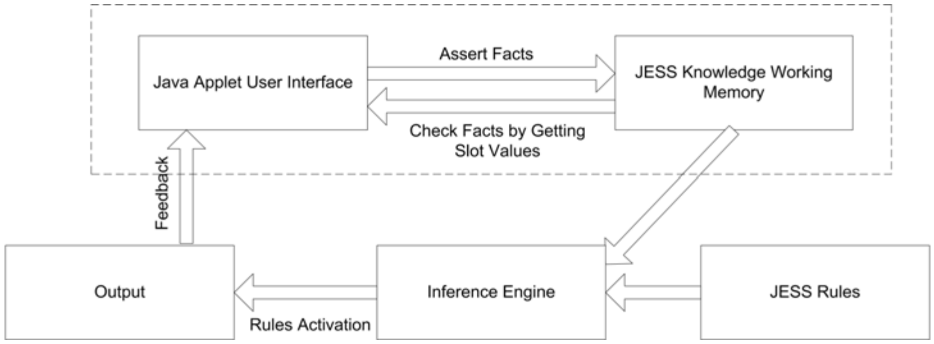

Figure 2. Interaction between Java and JESS

The case study is about Jacques Cartier Bridge in Montreal. Data related to the bridge were acquired from the bridge management authority including CAD drawings and inspection and maintenance records (PJCCI, 2004; Zaki and Mailhot, 2003). The inspection GUI is developed based on Ontario Structure Inspection Manual (OSIM) (OSIM, 2000), bridge inspection standards published by the American Association of State Highway and Transportation Officials (AASHTO, 1983), and "Bridge Inspector's Reference Manual" (FHWA, 2002). An inspector can apply inspection procedures through a number of ordered tabbed panes. The panes are *Inspector, Schedule, Instrument, Element, Damage,* and *Task* (Figure 3). In the first two tabbed panes, some general inspection information needs to be input about the inspector and 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. In the *Element* pane, the inspector can choose the exact 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 will be retrieved from the database and added to the pane, such as type, material, and dimension, etc. *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. Hyperlink functionality links to Hypertext Markup Language (HTML) format file according to defect type to allow the user to access inspection manuals. The link is context sensitive and will extract only the relevant information. In addition, detailed inspection contents can be recorded in a different defect pane after choosing the defect type. Before the input of inspection data, defect historic records can be retrieved from the database and displayed on the 3D element as a reference. After that the current defect can be marked on the 3D model at the picking point, and defect information is saved into the database. The last pane, *Task*, is to summarize the previous inspection information for future assessment (Figure 3).

Figure 4 shows the location-based defect visual inspection process with the combination of the navigation and picking functions. Before the navigation, element inspection sequences should be scheduled and be input into the database. At the beginning of inspection activities, virtual arrows will automatically guide the inspector with the predefined element inspection order to indicate the current inspection objective. Following this step, the inspector is requested 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 location of the inspector (obtained from tracking) and the location of the defect. 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 inspection interface. Figure 4 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 a specific shape and color, which are defined based on the defect type and deterioration degree, respectively. For instance, in Figure 4, the black sphere represents very serious section-loss.

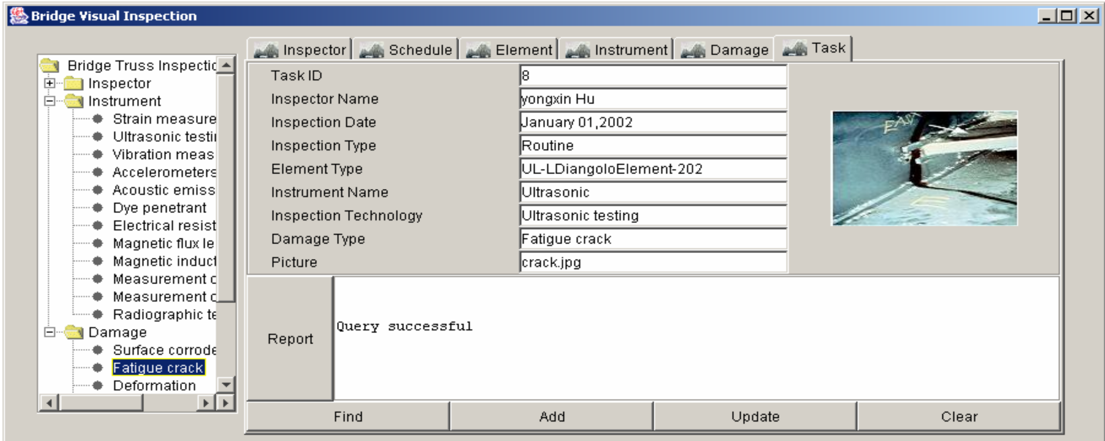

Figure 3. Inspection Task Report Tabbed Pane

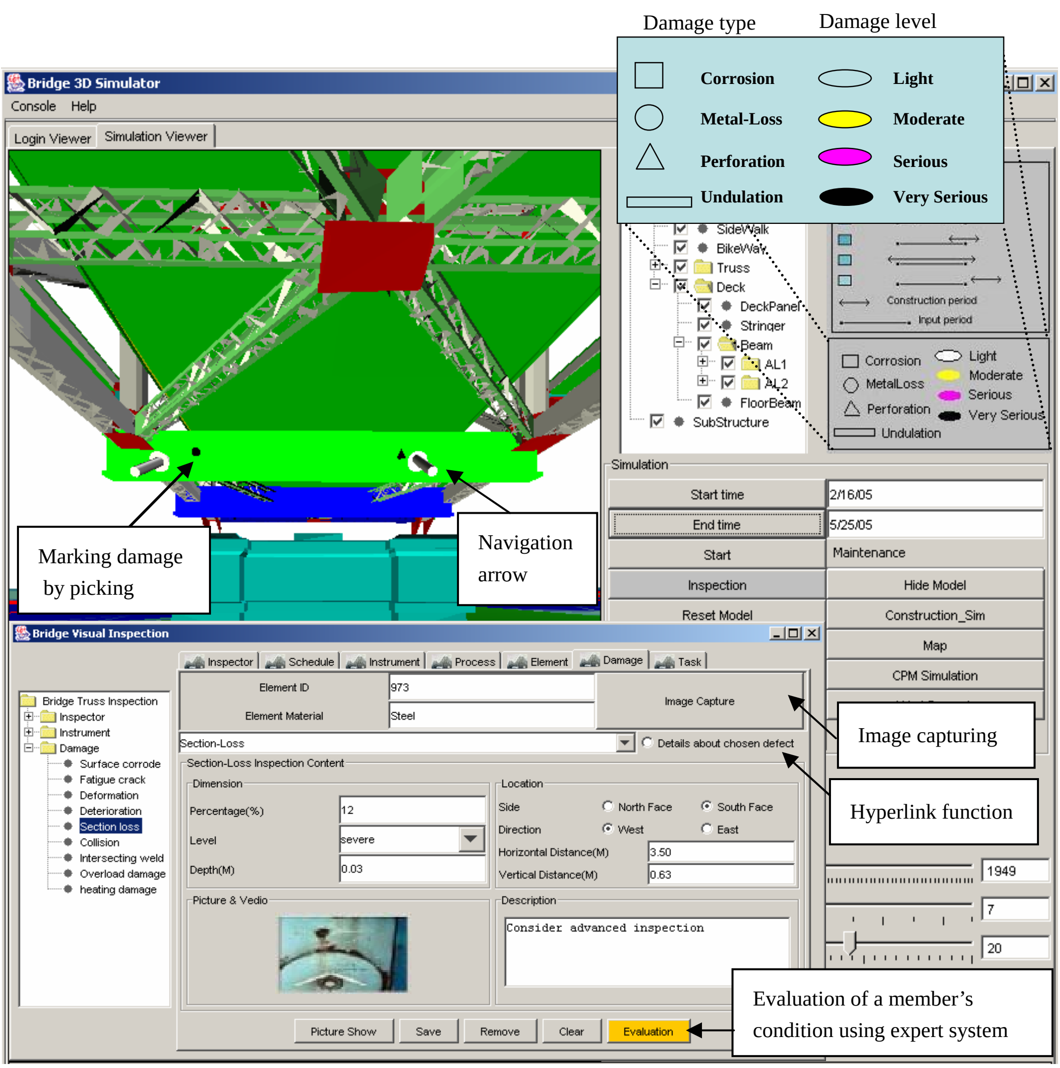

Figure 4. Location-based Bridge Visual Inspection

Figure 5 illustrates how rules are defined for bridge inspection. Inspection codes and manuals and relevant experiences have been converted into rules and saved in the knowledge base. Ponits, which is a component of AASHTO, is used as bridge elements condition rating description. Specific rules have been customized to be activated when asserted facts (inspection data input) 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 expert system. Through extracting relative data from the database using SQL in the java application, existing facts can be asserted into the expert system. Then, predefined rules in JESS are fired when their LHS are satisfied. The condition rating acquired from JESS will be displayed and saved into the database for further evaluation. In addition, the rule-based expert system can deduce element suspected performance deficiencies (SPD) based on deficiency deterioration degree. If an element's ability to perform its intended function is in question, possible actions will be suggested. For example, 20% section loss at mid-span of a steel girder will result in the load carrying capacity being suspected, and strength evaluation will be undertaken.

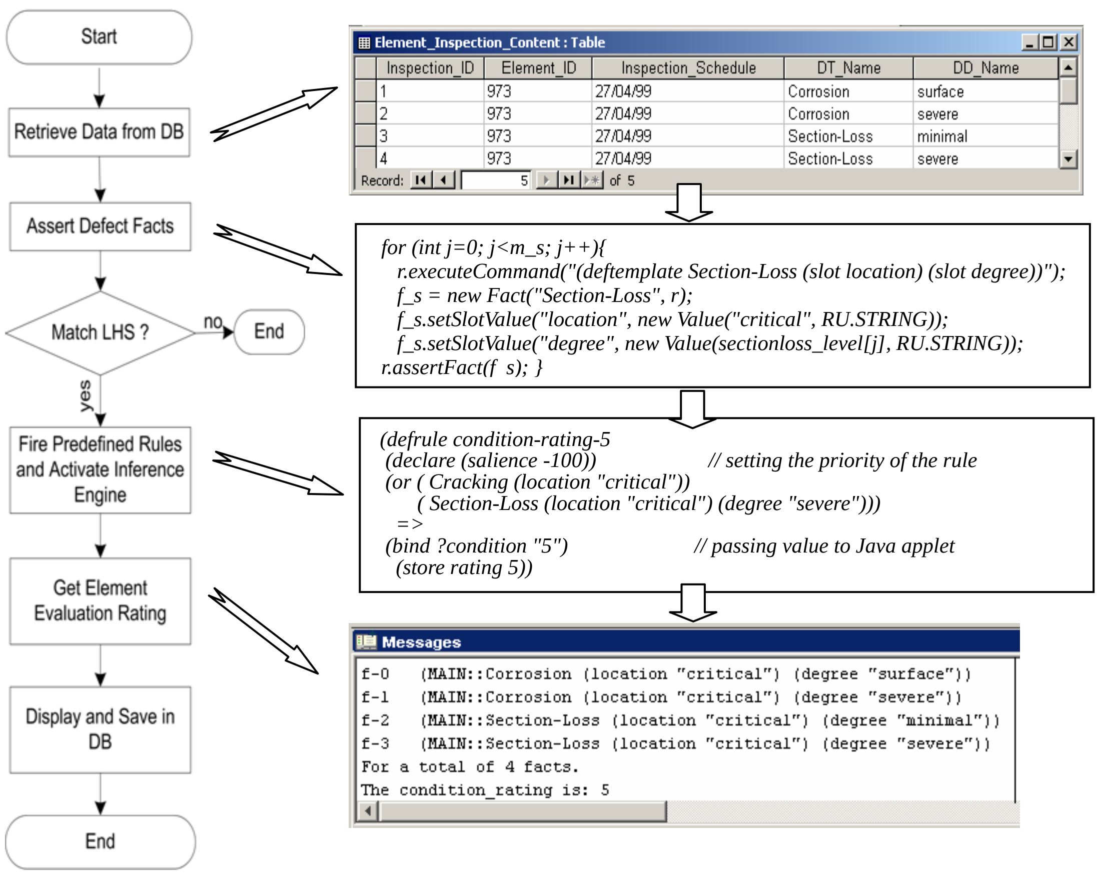

Figure 5. Bridge Element Condition Rating Evaluation

The application of Bayesian Networks in the proposed system is to obtain the most probable cause of the observed defect. Figure 6 shows the Bayesian Network representing the cause-effect relationships for steel bridge corrosion defects using Hugin Expert (2004) (FHWA, 2000). Based on the network, causes and their probabilities can be modeled and input into each node. When a defect is discovered, different questions are requested to be answered by the inspector. Each answer will adjust the cause probability until the diagnostic result is found.

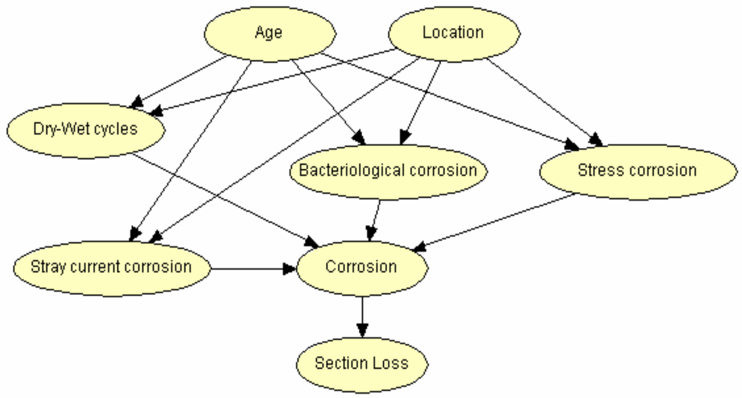

Figure 6. Bayesian Network of Steel Bridge Corrosion Defects Diagnosis

5. CONCLUDING REMARKS

In this paper, we proposed a new LBC approach to support the data collection activities of bridge inspectors. The proposed prototype system is equipped with a 3D detailed model of the bridge, and all the inspection results are registered on the 3D model. The system navigates the inspector to the locations he/she needs to inspect, provides information about the results of previous inspections as augmentation of the 3D model, and allows the inspector to add new information and to specify the location of a new defect simply by clicking on the point of the model where the defect has been found and then selecting the type and level of the defect from available menus. Furthermore, the system has a rule-based expert system that is used for data analysis and probabilistic diagnosis based on the location and the context of the inspection tasks in order to give the inspector suitable support. The system was implemented in Java language and a case study about Jacques Cartier Bridge was demonstrated. Mobility techniques are expected to facilitate the on-site activities and data collection, and improve safety under the severe field conditions. Future development will include further testing of the proposed prototype to improve its functionalities and usability in practical situations.

REFERENCES

AASHTO (1983). Manual for Maintenance Inspection of Bridges.

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

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

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 web site (2004). <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.

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.X. and Mozaffari, E.(2004b). “Mobile Model-Based Bridge Lifecycle

Management Systems," Proc.,Conference on Construction Applications of Virtual Reality.ADETTI/ISCTE, Lisbon,. 109-120

Hugin expert website (2004). <www.hugin.com>.

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

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

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

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

Mizuno, Y., Abe, M., Fujino, Y., Abe, M. (2002). "Development of Interactive Support System for Visual Inspection of Bridges," International Forum on Infrastructure Maintenance, November, 2003. Tokyo, Japan.

OSIM (2000), "Ontario Structure Inspection Manual," ISBN: 0-7794-0431-9.

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

Reinhardt J., Akinci B., Garrett J. H. (2004). "Navigational Models for Computer Supported Project Management Tasks on Construction Sites," Journal of Computing in Civil Engineering, 18(4), October 1, 2004, 281-290.

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

Russell, S. and Norvig, P. (2003)." Artificial Intelligence, A modern Approach," Prentice Hall.

Sloth, M., Johnsen, T.H., Larsen, E.S., Rasmussen, E. and Skaanning. C. (2004). "Use of Probabilistic Approach as Decision Support during Condition Evaluation"

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., 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, 417-424.

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

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

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, 2-15.