Journal of Theoretical and Applied Information Technology
15th November 2011. Vol. 33 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

APPLYING ISM AND ANN TECHNIQUES IN THE DEVELOPMENT OF USER EXPECTED COMMAND LOCATION PREDICTING APPROACH

CHUNWEI CHEN

Department of Technological Product Design, Ling Tung University, Taiwan, R.O.C.

E-mail: chenschool@yahoo.com.tw

ABSTRACT

The purpose of this research is to develop a method for obtaining information on user's expected command location and to verify its effectiveness in improving the inconsistency between the command locations of the software application tools and those expected by the users. This method proposed in this research is called User Expected Command Location Predicting Approach (UECLPA). It is defined as an approach to deconstructing user's expected idea of the command location and transform it directly to a treelike command element structure. This forecast method for user's idea of the command location is characterized with real-time modification of the predicting results of user's command locations. These two features are achieved by Interpretive Structural Modeling (ISM) and Artificial Neural Network (ANN). In terms of command locations, the inconsistency problem between the software application tools and the user's expectation is quite common in current software application tools. This often causes the users to have bad impressions on the product or even affect the success of the product. The interface designers should pay attention to this problem. In this study, users were invited to operate the old and new interfaces of a software application tool to verify the effectiveness of UECLPA. Results from the experiment indicated that there did exist differences between users and designers in the command location of software application tools. Moreover, UECLPA proposed in this study proved effective in improving the inconsistency problem between user's expectation and the ready-made command location in the software application tool. Therefore, it is, suggested that before organizing the command locations of software application tools, the interface designers should understand the users' ideas and demands. Last but not least, the application of UECLPA in the planning of command locations in software application tools will help decrease the inconsistency problem of software application tools.

Keywords: *Interface of Software Application Tools, Command Location Layout, Human-machine Interface, Interpretive Structural Modeling (ISM), Artificial Neural Network (ANN)*

1. INTRODUCTION

1.1. The Research Problem

The interface of software application tools is a user interface frequently used in 3C products such as mobile phones (Figure 1a), digital cameras (Figure 1b). Because digital commodities are products that modern people cannot do without, there are few people that have not touched such kind of user interface. It is, therefore, a fundamental issue for interface designers to consider how to make such kind of electronic products more user-friendly.

It is one of usability problem usually occurred on the interface of software application tools that command locations in the interface of software application tools does not meet the command locations the user wanted. For example, users expect that a specific command should appear somewhere but it doesn't. The inconsistency between command locations in the interface of software application tools and those expected by users is one of the reasons that impair the friendliness in the interface of software application tools [8]-[9]. And it is the gap between designers and users that causes the inconsistency. In most cases, designers plan the layout of commands in software application tools according to their own behavioral models. This results in the inconsistency between the command locations and those expected by users [17]. In planning the command locations of the interface of software application tools, designers often start with the construction of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

command path according to their behavioral and cognitive models. The so-called command path is referred to as the connected structural map designers used for references in the arrangement of command locations in software application tools. Such a connected structural map is often represented by an interrelated structural hierarchical graph. Based upon the command path, the layout of commands of software application tools is then organized. At this time, if the command path organized by the designer is inconsistent with the idea or cognition of command locations of users, the phenomenon of inconsistency between command locations in the software application tools and those expected by users may occur [17]. To solve the inconsistency problem in the interface of software application tool, it is essential to predict the command locations users expect or previously recognized. At the same time, if users' expected or previously recognized command locations can be transferred to the command path in the interface of software application tools, then such a command path can serve as a reference for the organization of command locations in software application tools so as to solve the above-mentioned inconsistency problem. Furthermore, the schedule of design and development of the interface of software application tools can be reduced in that two steps in understanding the user's expected or previously recognized command locations and transforming the command location layout into the command path have been integrated. This not only enhances the user friendliness in the interface of software application tools but also uplifts the efficiency of the interface design. It is, therefore, necessary and valuable for designers to develop a method to predict the structure of command locations the users expect and directly transfer it to the interrelated structural hierarchical graph for the arrangement of command locations in the interface of software application tools. In this study, a User Expected Command Location Predicting Approach (UECLPA) was proposed and verified in terms of the improvement of inconsistency between the command locations in software application tools and those expected by users.

1.2. Motivation and Purpose of Study

Currently, the construction of command layout in software application tools is primarily divided into two fields: usability engineering and human factors engineering. In usability engineering domain, heuristic evaluation (HE) and cognitive walkthrough (CW) are frequently applied to construct the command locations in the interface of software application tools. To apply HE method, four steps should be followed [7], [10], [11], [12], [15], [16]:

(1)Conduct a usability survey in which users are asked to operate and evaluate the command locations of the software system on their own;
(2)After the individual subjects finish the usability survey, process a group discussion to gather the command locations problems in using the interface of software;
(3)In evaluating the usability of software, the subject will repeat the command locations a couple of times and examine all command searching in the software interface;
(4)Compare and contrast the command location problems with usability principles to check whether or not each interactive command location has been well-designed in the interface of software system.

In addition, CW method verifies the user behaviors in interacting with commands by simulated experimental tasks. The user behaviors obtained from the experiment can be used for the arrangement of command locations in software application tools. To explore the cognitive activities users apply in executing the experiment tasks, CW method often adopts observation and think aloud or protocol to reveal the characteristics of user's behaviors in operating the commands [14], [15], [23], [24].

Furthermore, card sorting is a method frequently applied in the construction of command locations in human factors engineering. Its operation is composed of the following steps [3], [5], [18]:

(1)Give the subject (the user) a set of cards with titles of commands previously printed for the interrelated hierarchical structure of a specific software system; an item of command is printed on each card;

(2)According to his or her own way of thinking, the subject divides the cards into several clusters and gives a title to each cluster;
(3)The other subjects finish Steps (1) and (2) and finish the command clustering task;
(4)After all subjects finish the clustering and title giving of the command cards, the exprimenter analyzes the results so as to work out the information connecting path, which is called the information structure.

From the above-mentioned construction methods for the interface of software application tools

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

command locations, it is clear that, in terms of the command locations, most approaches do not predict the users' expectation of commands locations. At the same time, they are not equipped with the function of transferring users' expected command locations to the command path, in the meanwhile, to directly transfer the user's expected command locations to the command path by an interrelated structural hierarchical graph. For these reasons, the author attempted to propose a method that can do the two things mentioned above. In other words, in a proper way, the command location layout expected by the users will be predicated and transformed into an interrelated structural hierarchical graph for the command path in interface design for the software application tools. The purpose of the study is to test feasibility of applying UECLPA to predict user's expectation of the command locations of software application tool interface is explored.

2. THE USER EXPECTED COMMAND LOCATION PREDICTING APPROACH (UECLPA)

2.1. ISM Technique (transfer the user's expected command locations to the command path)

2.1.1. Theory of ISM technique

Interpretive Structural Modeling (ISM) is a computer-assisted process that enables individuals or groups to develop a map of the complex relationships between the many elements involved in a complex situation [21]. It was first proposed by J. Warfield in 1973 to analyze complex socioeconomic systems. Its basic idea is to use a user's practical experience and knowledge to decompose a complicated system into several sub-systems (elements) and construct a multilevel structural model [21]. Because the theory of ISM technique lies in the application of Hierarchical Digraph in Graphic Theory, with ISM technique, the relations among different elements in a complicated system can be transformed into the interrelated structural hierarchical graph (Figure 2) to describe the relationships among different elements [19]-[22]. This implied that ISM technique provides a fundamental understanding of complex situations using constructing an interrelated structural hierarchical graph. In light of this, if the elements in the interrelated structural hierarchical graph are set to be the commands previously set in the interface of software application tools, then the ISM interrelated structural hierarchical graph will be the command path. This means that ISM is able to directly transform the user's expected command locations into the command path in term of the interrelated structural hierarchical graph. Therefore, UECLPA method employs ISM for the core technique. Currently, ISM technique has been widely applied in production, business management, and educational research domains. For instance, ISM has been used to investigate the labor issue [1]-[2]. In business management domain, ISM has been used to evaluate values [4]. And in educational research field, ISM has been used in curriculum planning.

The following steps are the procedure for constructing the ISM interrelated structural hierarchical graph. In the beginning, ISM calculates the binary matrix (or correlation matrix) through the individual or group psychological model from which the relative correlation is represented by the values of elements. For example, if the set of Matrix A has n elements and the relation between element a_i and element a_j, then it can be denoted as $A=(a_{ij})_{n\times n}$ (Equation(1)). If $a_{ij}=1$, then it means that element a_i is a reachable matrix of a_j (Equation(2), (3)). After the reachable ***M****** is obtained, Equation (4) can be used to find the mutually related elements. Organize the mutually related elements in their ranking order and define the correlation directions according to the relationships among elements in reachable matrix ***M****** (by arrows), we can construct the interrelated structural hierarchical graph as shown in Figure 2.

$$A=[a_{ij}]=\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{bmatrix} \quad (1)$$

$$M = D + I \quad (2)$$

$$M^{*} = M^{k} = M^{k+1} \quad k>1 \quad (3)$$

$$M(a_i)\bigcap A(a_i) = R(a_i) \quad (4)$$

2.1.2. ISM Steps in the UECLPA approach

The following steps illustrate how to apply ISM technique in UECLPA:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Step 1: Organize an users group which is composed of 12 users with background and knowledge of user interface operation and an expert group which is composed of 2-3 experts with background and knowledge of user interface design. Two of 12 users participated ISM steps in UECLPA approach and the others participated ANN algorithm steps in UECLPA approach.

Step 2: In a team work, identify command items to construct command menu.

Step 3: Encode the identified command items for the software.

Step 4: Construct an adjacency matrix of identified command items, $M = (a_{ij})_{n\times n}$ (Equation (1)).

Step 5: Ask the subjects to make judgments for every command code in terms of the command location relationship. For example, ask the subjects whether command a_i should link to command a_j. If the subjects consider it is necessary that command a_i should link to command a_j, then $a_{ij} = 1$; otherwise, $a_{ij} = 0$, meaning that command a_i is not supposed to link to command a_j.

Step 11: Determine the reachability matrix *M* of the command items by Equations (2) and (3).

Step 12: From the analysis of hierarchical matrix, generate the command path by the interrelated structural hierarchical graph (Equation (4)). Then, the interface design experts organize the command locations in the interface based upon the interrelated structural hierarchical graph. For instance, in the interrelated structural hierarchical graph in Figure 1, the command item on the top layer in the interrelated structural hierarchical graph (Command a_2 in Figure 1) is assigned as the entry command (the command item in the very beginning). The other command items are then organized to different layers according to their orders in the interrelated structural hierarchical graph. For example, the Command items a_3 and a_4 in the second layer of Figure 1 are arranged under Command a_2; Command $a5$ is under Command a_4. In this way, the command items on the same layer of the interrelated structural hierarchical graph (e.g., Commands a_3 and a_4 in Figure 2) will be assigned in the same location. For example, Commands a_3 and a_4 in the second layer of Figure 1 are assigned in the lower layer of Command a_2.

2.2. ANN Technique (predict the users' expectation of commands locations)

2.2.1. Theory of ANN technique

Artificial Neural Network (ANN) is an artificial intelligent algorithm developed from medical science, mathematics, physics, information and electronics. It is an algorithm generated by the operation of human brain. Artificial neurons are the simplified neurons of creatures. The data from outer world (e.g., user entry) or other artificial neurons will be calculated and compared with a specific variable. If the data is bigger than (or smaller than) the specific variable, then it will trigger the neuron to execute a certain response, just like people exercise responses to the outer information by the storage data accumulated in our brains. This is the reason why it is called Artificial Neural Network (ANN). Through learning, ANN can calculate a predicted value from the old data and compare with the observed value. If the predicted value is different from the observed value, then its weight in ANN will be adjusted. In this way, an optimum weight to solve a specific predicted value will be identified through comparison of multiple previous data (Figure 3). This means that ANN has the learning and prediction abilities. Meanwhile, if ANN technique is incorporated with ISM technique in the construction of correlation matrix, then ISM technique will have the effectiveness to integrate new information that meets the command locations expected by the users. In other words, the application of ANN technique will help predict and adjust the user's expectation of command locations in the application software tools. Therefore, the prediction method for the command path expected by users in the interface of application software will cover the core ANN technique.

$$D_{netij} = \lfloor a_{ij} \rfloor \times [net_h] \times \lfloor net_j \rfloor \quad \text{where} \quad i, j = 1,2,\cdots,m \qquad (5)$$

$$net_h = \sum_{i=1}^{n} W_{ih} X_i - \theta_h \quad (6)$$

$$H_j = a \times \tanh(b \times net_h) \quad (7)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$net_j = \sum_{h=1}^{n} W_{hj} X_h - \theta_j \ (8)$$

$$Y_j = a \times \tanh(b \times net_j) \ (9)$$

2.2.2. ANN Algorithm steps in the UECLPA approach

The application of ANN technique in the prediction of user expected command locations in interface of software application tools consists of the following steps:

Step 6: Start learning the correlation of user expected command locations by using Equation (5);
Step 7: Read the weight value and threshold value obtained from the leaning procedure (Equation (6));
Step8: Read the input vector of the test sample (i.e., the correlation matrix for command items expected by every subject) (Equation (7));
Step 9: Calculate the output values of the hidden layers and the output layers (Equation (8));
Step 10: Calculate the output value of the output layer (i.e., the predicted correlation matrix for command items by all subjects) (Equation (9)).

2.3. Limits to the Application of UECLPA

The UECLPA proposed in this study is limited to the prediction of command locations in the interface of software application tools. The prediction of command locations in other types of interface is beyond the scope of this study.

3. VERIFICATION METHODS AND THE EXPERIMENTAL RESULTS

The interface of Sony Ericsson K530i mobile phone, a typical interface of software application tools, is used as an example to illustrate the operational procedure of UECLPA. In this section, the effectiveness of UECLPA is verified by two aspects: (1) the operational smoothness the UECLPA have in predicating the command locations, and (2) the predicating accuracy comparison of users' expected command locations obtained by UECLPA (the new command path) and that of the original Sony Ericsson K530i mobile phone interface (the old command path). The operational smoothness of UECLPA is defined as the number of repeated operations for a certain task due to the errors in operation. The fewer operation errors means the better operational smoothness. In addition, the predicating accuracy is defined as the number of prediction, the prediction time and satisfaction of command locations. The fewer prediction number, less prediction time and higher satisfaction of command locations means the better predicating accuracy.

3.1 The Operational Smoothness of the UECLPA

3.1.1. Verification method

In the operational smoothness experiment, 12 Sony Ericsson K530i mobile phone users and 2 interface design experts were invited for the evaluation task. The 12 Sony Ericsson K530i mobile phone users were graduate students at the Graduate School of Technological Product Design, Lin Tung University (7 males and 5 females at the average age of 22.4 (*SD*=0.8)) and had used Sony Ericsson K530i mobile phone for more than two years. Two interface experts were designers at the Department of Mobile Phone Design, Taiwan BenQ Corporation, with the average age of 35.4 and more than 7 years of experience in the design of interface of software application tools. The major experiment task of operational smoothness experiment is that the 12 mobile phone users and 2 interface design experts re-layout the command locations of Sony Ericsson K530i mobile phone using Steps (2)-(12) of the UECLPA method. At the same time, a video is also used for recording the real procedure in operation UECLPA method so as the errors number of repeated operations for each step of the UECLPA method can be observe.

3.1.2. Verification results and discussions

Comparison of the command paths from UECLPA and the original mobile phone interfaces

Figure 4 illustrates the result of UECLPA in predicating the user's expected command locations of the interface of Sony Ericsson K530i mobile phone. The new command path in the interrelated structural hierarchical graph obtained from UECLPA is shown in Figure 4b, defined as the new command path for Sony Ericsson K530i mobile phone. The original command path in the interrelated structural hierarchical graph for Sony Ericsson K530i is shown in Figure 4a, defined as the old command path. And Figure 4c is the newly planned interface for Sony Ericsson K530i according to the new command path from EUCLPA. From the comparison of the new and old command

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

paths, several finding can be obtained. (1) The number of layers of the new command path is bigger than that of the old command path. While there are only two layers in the old command path, there are 5 layers in the new command path. (2) The classification type of command items in the new command path is different from that of the old command path. The type of classification of command items in the newly obtained command path features the episodic and contextual characteristics. For example, it is clear in the new command path that similar command items will be categorized into the same group. All of the command items related to communication like message type (B, B1, B2, B3, B4, B5, B6, B7, B8), transmission type (D, D1, D2, D3, D4), and contact person type (C, C1, C2) are connected together. Moreover, command items related to the basic functions of mobile phone such as the functional setting type (A, A1, A2, A3, A4) and organizational tool type (F, F1, F2, F3, F4, F5) are linked together. At last, command items related to entertainment such as game type (A5, A6, A7)(G), web type (H), and radio channel type (I) are grouped together. In the original command path of Sony Ericsson K530i, these command items that linked together are separated in terms of their command locations. (3) In the new command path, all command items are linked to the transmission type commands (D, D1, D2, D3, D4). This highlights the episodic and contextual characteristics and makes it a lot different from the old command path. The differences between the old and new command paths indicate that users and designers are different in terms of the demands of the command locations in the interface of common software application tools. More importantly, the UECLPA proposed in this study is effective in the transferring of the user expected command locations of the interface in software application systems.

The operational smoothness of UECLPA

The number of repeated operation in each step of UECLPA is shown in Table 1. It is clear that subjects would repeat the operations only in a few steps of UECLPA, for example, Step (5) and Step (6). There was no repeated operation in other steps, indicating that it was not difficult to operate the procedure in UECLPA. Overall, the subjects had no problems in conducting the UECLPA tasks.

In addition, the reasons for the repeated operations in steps of UECLPA were explored from interviews, which would help improve the efficiency in applying UECLPA procedures.

(1) Step 5: The arrangement of the command items in the correlation matrix expected by the users was not suitable for the correlation judgment of command locations when the number of command items was too big.

(2) Step 6: Many subjects claimed that the steps for the entry of the correlation matrix of command items were tedious and annoying and should be simplified.

3.2. The Predicating Accuracy of UECLPA

3.2.1. Verification Method

In order to verify the effectiveness of the UECLPA method in the predicating accuracy of command locations in software application tool interface, the task performance of new and old interfaces of Sony Ericsson K530i mobile phone were compared in terms of the number of prediction, the prediction time and satisfaction of command locations. In this study, the new interface is defined as operational interface of Sony Ericsson K530i (Figure 4c) planned by the command path from the interrelated structural hierarchical graph by UECLPA while the old interface refers to the original operational interface of Sony Ericsson K530i. The number of prediction is defined as the number of prediction the subject correctly predicts the command locations; the prediction time refers to the time the subject spends in predicting the command locations; the satisfaction of command locations is looked upon the degree of satisfaction the subject has towards the command location layout on a five-point Likert scale. These three criteria are used to compare whether the newly planned interface is better than the old interface, from which the effectiveness of UECLPA in predicting the user's command locations is verified.

In the experiment, the subjects are asked to perform three tasks: Task 1: to predicate for the command location of contacting a person; Task 2: to predicate for the command location of date setting; Task 3: to predicate for the command location of command of game playing. The predicating task is that the subject would tell the experimenter the command location of target command where he or she possibly guessed. 20 graduate students at the Graduate School of Technological Product Design, Lin Tung University (12 males and 8 females at the average age of 21.6 (*SD*=2.4)) were invited. They had 2-3 years in using Sony Ericsson K530i mobile phone. The premise of the experiment is that the subjects have the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

experience in using Sony Ericsson K530i. They were randomly selected for the experiment.

The experiment was conducted in the following procedures. (1) First, the subjects were asked to perform the command location predicating task of contacting a person command, to predicate the command location for the command of date setting, and to predicate the command location for the command of game playing in the newly planned interface. Then, the same tasks were performed in the original interface. (2) The order for the subjects to perform the three tasks is the same in the new and old interfaces of Sony Ericsson K530i. (3) The number of prediction and the prediction time were recorded at the same time when the subject performed the task while the satisfaction of command locations was evaluated after the tasks. (4) After the subject finished all three tasks, the differences between the new and old interfaces were analyzed by the researcher in terms of each criteria of performance.

3.2.2. Verification Results and Discussions

The number of prediction

Table 2 lists the task performance of the old and new interfaces in terms of three tasks. It is clear that in each task, the subject could spend fewer times in predicting the command locations correctly in the new interface. For instance, in predicting for the command to contact a person, the prediction number in using the new and old interfaces are 2.2 and 3.8 respectively. In predicting for the command of date setting, the numbers of prediction in the new and old interfaces are 3.4 and 4.6 respectively. And in the task to predicting for the command of game playing, the subject's numbers of prediction in the new and old interface are 3.1 and 4.2. Because the smaller number of prediction for commands means that the command locations meet the user's expectation so that the subjects didn't try so many times to predict the target commands. It is clear that the command locations the newly interface planned is better than the command locations in the original interface.

The prediction time

From Table 2, it is also clear that, in each task, the subjects would spend less time predicting the commands in the newly planned interface. For example, in the predicting of the command to contact a person, the time the subjects spent in the new and old interface are 4.3 seconds and 5.4 seconds respectively. In the task to predict for the command of date setting, the subjects spend 5.5 seconds in the new interface but 7.1 seconds in the old interface. Similarly, to predict for the command of game playing, it only took the subjects 5.3 seconds in the new interface but 6.2 seconds in the old interface. This also demonstrated that the newly planned interface could meet the user's expectation in terms of the command locations. Therefore, they could find the target commands in much less time in the newly planned interface.

Satisfaction of command locations

The comparison of the satisfaction of command locations also reflects the better performance of the newly planned interface in all tasks. In predicting for the command of contacting a person, the subjects had 3.2 in the new interface but only 2.6 in the old interface. In the task to predict for the command of date setting, the degrees of satisfaction to the new and old interfaces are 3.2 and 2.5. And to predict for the command of game playing, the subjects also had a higher degree of satisfaction to the new interface, 3.5 and 2.6 for the new and old interfaces. It is, therefore, rational to conclude that the newly planned interface could better meet the user's expectation in terms of command locations.

Summary

From the comparison of the new and old interfaces in different tasks, it indicates that the UECLPA method can really help solve the problem that the command locations in the interface of software application tools is not compatible with that expected by the users. And its effectiveness can reach more than 40% in improving the command locations in the interface of software application tools. The results of the experiment demonstrated that in terms of three performance criteria, the new interface can help improve more than 40% of the subjects' task efficiency and satisfaction. This indicates that UECLPA method has the advantage in meeting the user's expectation in the command locations of the interface of software application tools.

4. CONCLUSIONS AND SUGGESTIONS

In this study, the author proposed a UECLPA method to solve the inconsistency problem in the interface of software application tools. Subjects were invited to operate the current and newly developed interfaces to verify the effectiveness of UECLPA from the comparison of their performance. Results of the experiment demonstrated that there

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

exist differences between users and designers in the command location of software application tools. Moreover, UECLPA proposed in this study are effective in improving the inconsistency problem between user's expectation and the ready-made command location in the software application tool. Therefore, it is, suggested that before organizing the command locations of software application tools, the interface designers should understand the users' ideas and demands, from which the layout of the commands in the interface of software application tools can be further organized. At last, designers can apply UECLPA in the planning of command locations in software application tools to help decrease the inconsistency problem in the interface of software application tools.

REFERENCES:

[1] A. Agarwal, R. Shankar, and M.K. Tiwari, "Modeling agility of supply chain", *Industrial Marketing Management*, Vol.36, No.4, 2007, pp.443-457.

[2] M.T. Chandramowli, and F.A. Felder, "Analysis of barriers to development in landfill communities using interpretive structural modeling", *Habitat International*, Vol.35, No.2, 2011, pp.246-253.

[3] C. Courage, and K. Baxter, "Multiple Level Weighted Card Sorting", *Understanding Your Users*, 2005.

[4] D.R. Farris, and A.P. Sage, "On the use of interpretive structural modeling for worth assessment", *Computers & Electrical Engineering*, Vol.2, No.2, 1975, pp.149-174.

[5] J. Harloff, "Card Sorting Prototyping, Methodology", *European Journal of Research Methods for the Behavioral and Social Sciences*, Vol.1, No.4, 2005, pp.119-128.

[6] R.W. Hawthorne, and A.P. Sage, "On applications of interpretive structural modeling to higher education program planning", *Socio-Economic Planning Sciences*, Vol.9, No.1, 1975, pp.31-43.

[7] R. Jeffries, and H.W. Desurvire, *Usability testing vs. heuristic evaluation: was there a context*, ACM SIGCHI Bull, 1992.

[8] R.J. Kamper, "Extending the usability of heuristics for design and evaluation: lead, follow and get out of the way", *International Journal of Human-Computer Interacttion*, Vol.14, No.2, 2002, pp.447–462.

[9] G.J. Meyers, *Reliable Software Composite Design*, New York: Petrocelli/Charter Press, 1975.

[10] R. Molich, and J. Nielsen, "Improving a human–computer dialogue", *Commun. ACM*, Vol.33, No.3, 1990, pp.338–348.

[11] J. Nielsen, Finding usability problems through heuristic evaluation, In: Bauersfeld, P.m, Bennet, J. and Lynch, G.(Eds.), *Proc. ACM CHI'92 International Conference on Human Factors in Computing Systems*, New York: ACM Press, 1992, pp.373–380.

[12] J. Nielsen, Usability Engineering, Cambridge, MA: Academic Press., 1993.

[13] E. Olsson, and A. Jansson, "Participatory design of a collaborative clinical trial protocol writing system", *Interacting with Computers*, Vol.17, No.2, 2005, pp.147-166.

[14] P. Polson, C. Lewis, and J. Rieman, "Cognitive walkthroughs: a method for theory-based evaluation of user interfaces", *International Journal of Man-Machine Studies*, Vol.3, No.6, 1992, pp.741–773.

[15] A. Sears, and D.J. Hess, "Cognitive walkthroughs: understanding the effect of task description detail on evaluator performance", *International Journal of Human-Computer Interacttion*, Vol.11, No.3, 1999, pp.185–200.

[16] A. Sears, "Heuristic walkthroughs: finding the problems without the noise", *International Journal of Human-Computer Interacttion*, Vol.9, No.3, 1997, pp.213–234.

[17] A. Sundin, M. Christmansson, and M Larsson, "Participatory design in community informatics", *International Journal of Industrial Ergonomics*, Vol.33, No.1, 2004, pp.1-14.

[18] M.A. Tinker, A.J. Imm, and C.A. Swanson, "Contextual Inquiry, Task Analysis, Card Sorting", *Journal of Experimental Psychology*, Vol.15, No.2, 1932, pp.206-211.

[19] J.N. Warfield, "On arranging elements of a hierarchy in graphic form", *IEEE-SMC(3)*, 1973, pp.121–132.

[20] J.N. Warfield, "An introduction to the application of interpretive structural modeling", *Proceedings of the IEEE*, Vol.63, No.3, 1975, pp.397–404.

[21] J.N. Warfield, *Societal Systems: Planning, Policy and Complexity*, Wiley, New York. 1976.

[22] J.N. Warfield, A Science of Generic Design: Managing Complexity Through System Design, *Intersystems, Salinas*, CA. 1990.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[23] C. Wharton, J. Rieman, C.Lewis, and P. Polson, The cognitive walkthrough method: a practitioner's guide. In: J. Nielsen, & R. Mack (Eds.), *Usability Inspection Methods*. New York: Wiley Press. 1994, pp.105–141.

[24] L. Yuanhua, A.L. Osvalder, and S. Dahlman, "Exploring user background settings in cognitive walkthrough evaluation of medical prototype interfaces: a case study", *International Journal of Industrial Ergonomics*, Vol.17, No.5, 2005, pp.5379–390.

AUTHOR PROFILES:

Dr. Chunwei Chen received the master degree in industrial design from the National Chungkong University in Taiwan, R.O.C.. He received the Ph.D. degree in design from the National Yunlin University of Science and Techonology in Taiwan, R.O.C.. Currently, he is a professor at Ling Tung University in Taichung, Taiwan. His research interests include usability engineering, ergonomic design and consumer information investigating.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

a.

b.

Figure 1 Examples of The interface of software application tools, a: mobile phones, b: digital cameras

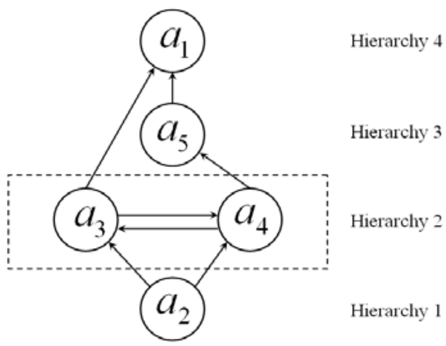

Figure 2 An interrelated structural hierarchical graph developed from ISM

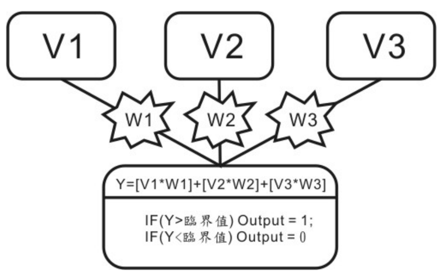

Figure 3 The concept of ANN technique

Journal of Theoretical and Applied Information Technology
15th November 2011. Vol. 33 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

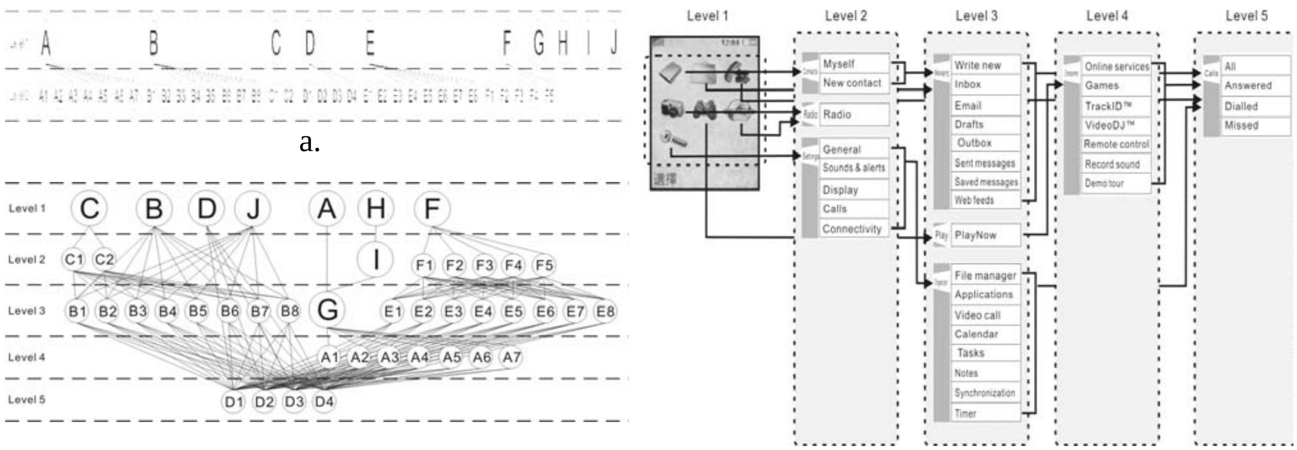

Figure 4 a. The old command path; b. The new command path from UECLPA; c. The newly planned interface

Table 1 The operational smoothness of UECLPA

No. of Step	1	2	3	4	5	6	7	8	9	10	11	12
Number of repeated operations	0	0	0	0	1.2	1.4	0	0	0	0	0	0

Table 2: Comparison of the task performance of the old and new interfaces in terms of three tasks

Tasks	Criteria	Task performance		
		Newly planned interface	The old interface	*t*-test
Predicting for the command of contacting a person	the number of prediction (time)	2.2 (*SD* = 0.78)	3.8 (*SD* = 0.41)	F=2.8*
	the prediction time (secs)	4.3 (*SD* = 1.62)	5.4 (*SD* = 1.23)	F=2.1*
	satisfaction of command locations	3.2 (*SD* = 1.33)	2.6 (*SD* = 1.25)	F=1.8*
Predicting for the command of date setting	the number of prediction (time)	3.4 (*SD* = 0.41)	4.6 (*SD* = 0.33)	F=2.5*
	the prediction time (secs)	5.5 (*SD* = 1.44)	7.1 (*SD* = 1.68)	F=3.1*
	satisfaction of command locations	3.2 (*SD* = 0.89)	2.5 (*SD* = 1.01)	F=1.2
Predicting for the command of date setting	the number of prediction (time)	3.1 (*SD* = 0.53)	4.2 (*SD* = 0.44)	F=3.6*
	the prediction time (secs)	5.3 (*SD* = 1.13)	6.2 (*SD* = 1.31)	F=2.9*
	satisfaction of	3.5 (*SD* = 0.92)	2.6 (*SD* = 0.73)	F=2.1*

*: $p < 0.05$