Proceedings of the Korean Nuclear Society Spring Meeting
Cheju, Korea, May 2001

A Study on HCI Design Strategy using Emergent Features and Response Time

Sung-Jin Lee, Jin-Gyun Park* and Soon Heung Chang

Korea Advanced Institute of Science and Technology
373-1, Kusong-dong, Yusung-gu, Taejon, Korea 305-701

* Korea Atomic Energy Research Institute
150 Dukjin-dong, Yusong-gu, Taejon 305-353, Korea

ABSTRACT

Existing design process of user interface has some weak point that there is no feedback information and no quantitative information between each sub process. If they're such information in design process, the design time cycle will be decreased and the contentment of HCI in the aspect of user will be more easily archived.

In this study, new design process with feedback information and quantitative information was proposed using emergent features and user response time. The proposed methodology was put together with three main parts. First part is to calculate distinctiveness of a user interface or expanded user interface with consideration of emergent features. Second part is to expand a prototype user interface with design option for purpose of design requirement using directed structure graph (or nodal graph) theory. Last part is to convert non-realized value, distinctiveness, into realized value, response time, by response time database or response time correlation in the form of Hick-Hyman law equation.

From the present validations, the usefulness of the proposed methodology was obtained by simple validation testing. It was found that emergent features should be improved for high reflection of real user interface. For the reliability of response time database, lots of end-user experiment is necessary. Expansion algorithm and representation technique of qualitative information should be somewhat improved for more efficient design process.

1. Introduction

With the increased demand for software technology and automation, the user interface of computers and displays have tremendously increased in engineering application field. There are two major trends in the development of ACRs(advanced control room). The one is increasing automation without any human interaction; the other is development of intelligent operator aid systems within computer technology. Especially, operator aid system in nuclear engineering has specific types of time-limited process [1,5]. Among them, HCI(human computer interface, or human computer interaction) designs are emphasized in case of the development of future ACR workstation, because operators should perform their task, such as monitoring and process control, through carefully designed and organized HCI with computer-based GUI(graphic user interface). The complexity of computer systems has created a growing awareness for the need to improve the design of human computer interfaces. From 1990s, some experts had interest of HCI evaluation, because in various engineering field there were a lot of user interfaces that were not insured by ideal-like design rule. From now on, there are no fixed ideal rule in HCI design phase and evaluation phase. Also there is no quantitative feedback method from evaluation phase to design phase [2].

HCI applications have the various types whose detailed cases are such as nuclear power plant's operator aid system, office automation program and web service. Many applications require the users to react to every situation correctly and timely. User interface designers of such application make the system by individually meaningful rule. This aspect is the origin of HCI performance evaluation research. This paper would make a ideal-like evaluation rule and somewhat quantitative feedback method. And it would be applicable to many HCI application areas.

Configurality and emergent features are related concepts that are influencing much basic work in form perception [3,4,9,10]. When simple visual elements are put together, new features sometimes appear that were not present in the original elements. For example, four straight lines can be joined to form a rectangle, which produces the features of closure, area, and possibly symmetry. These are called emergent features because they arise from the arrangement of the lines without being identifiable with any single line. The rectangle itself would be called a configural display because the configuration of its elements produced these emergent features. Unfortunately there are no widely agreed upon definitions of emergent features or configurality. Therefore, this study had considered the environment of user interface development process. Most developing tool in HCI applications is under Unix or Windows. The developing tool has basic interface elements that are text-box, shape, color, highlight, and symbols, etc [9,10]. Emergent features can be extracted from developing tools and Window software by commonly detectable elements.

Traditional process of user interface development has some disadvantages that one is feed-forward process of user interface and the other is no quantitative feedback information from evaluation result to design step. It is need to propose new process that has somewhat quantitative feedback. This study introduces the concepts of emergent features and method to apply measured user response time for a design strategy or top-level design requirement [7,8].

2. Literature Review

HCI evaluation methods have very various types. Evaluation method in this study is a kind of analytic methods but feedback method is related to observational and/or experimental methods because it uses user response time database. It is necessary to summarize whole evaluation methods as follows. Main objectives of HCI evaluation are as follows:

- To determine the effectiveness or potential effectiveness of an interface,
- To provide means for suggesting improvements,
- To establish that an interface operates effectively,
- To make improvements to an interface.

There is two kinds evaluation that are formative evaluation (User Feedback) and summative evaluation (No User Feedback). UI evaluation corresponds to a requirement analysis in some system design. Every evaluation takes place within a specific context:

- The experience level of the users,
- The types of task undertaken,
- The system being used,
- The environment in which the study is undertaken.

Widely agreed evaluation criteria or factors related to human actions have four items as follow:

- Efficiency,
- Accuracy,
- User satisfaction,
- Usability (potentiality).

2.1 Feature of evaluation methods

Whole HCI evaluation methods can divide into five groups by its development stage, user involvement and produced data. Analytical method is usable early in design. Few resources required therefore cheap. It has narrow focus on tasks, lack of diagnostic output for redesign and broad assumptions about users' cognitive operations. Expert method is strongly diagnostic. It has overview of whole interface, few resources, high potential returns, and many restrictions in role-playing. It is subject to experts' biases with problems in locating enough experts. Therefore it cannot capture real behavior of users. Observational method quickly highlights difficulties. It uses verbal protocols valuable source of information on users' models. It can be used for rapid iterative development with rich data. Observation can affect user performance. Analysis of data can be time and resource consuming. Survey method addresses user' opinions and understanding of an interface. It can be diagnostic and applied to users and designers. Questions can be tailored to individuals. It is useful for large groups of users and low response rates with some possible interviewer bias. Analysis of this method can be complicated and lengthy. These interviews are very time consuming. Experimental method is very powerful method. It can produce quantitative data for statistical analysis and compare different groups of users with reliable and valid data. This method needs high resource demands and time consuming. It cannot always represent a full system.

2.2 Evaluating with users.

It is impossible to tell how good or bad an interface is unless you try it out on real users. Several methods are used but first you must find and recruit a group of representative users.

Choosing users for evaluations

It's fairly obvious that they should be representative of the intended users of the system. If it is a systems for doctors then get real doctors to test it. This is not an easy task, real users are usually busy and it can be difficult to persuade their supervisors to release them. However it is up to you to keep trying. If it is completely impossible then we could try approximations. If you cannot get a group of doctors then we could try medical students perhaps.

A few concerns about the testers

Serving as an evaluator is a stressful experience so you must try to keep this stress to a minimum. It can also be embarrassing so be careful of this. Try not to pressurize them into taking part in the evaluations and emphasize that it is purely voluntary for them. If at any time, they wish to stop and pull out of the test then let them without asking for a reason.

- Observational evaluation**:** Probably the best method but not always. Usually quite an informal and straightforward method. Users are given the system to be tested and a set of tasks to undertake. The HCI expert who does not interfere in any way observes their actions. Very often the users are asked to make some form of report on their actions.
- Direct observation**:** The basic method in which expert observes individual users during task execution. This is an obtrusive method as users are always aware of the observer and this may alter their performance. (Hawthorne Effect, named after an American Psychologist/Sociologist). Only a single pass analysis is possible as no record of the full user activity is kept.
- Video recordings**:** A very popular method that removes the pressure of the presence of the observer but introduces new pressures with the presence of video cameras. Often several cameras are used to make a composite recording. Has the advantage that users' actions can be analyzed many times.
- Interactive observation**:** Sometimes called the 'Wizard of Oz' method. A hidden operator/expert simulates the actual system and users interact with an interface to receive the designed feedback from the hidden observer. User should have the impression that a complete system exists. It is very good approach for the early stages of design to speed up the iterative development of complex systems.
- Verbal protocols**:** Users are asked to 'think out loud' as they undertake set tasks in the system. Very useful as a supplement to video recording but unfortunately can put too many demands on the users.

2.3 Evaluation without users

Three main methods in this case:

- Cognitive walkthrough, Action Analysis, Heuristic Evaluation

Cognitive Walkthrough

It is a development tool that makes a formalized way of imagining users' thoughts and actions. It can be an informal 'in your head' approach and more useful if a group approach is adopted.

Action analysis

Two main approaches are used. Which one is actually used depends on the level of detail required:

- Keystroke level, Back-of -the –envelope.

Features of keystroke:

- Detailed, Fine grained, Formal

Keystroke level can make reasonably accurate forecasts of the time a user needs to perform a task. To do this a task is broken down (decomposed) into smaller and smaller sub-tasks until we are listing the actual keystrokes need to perform the task. To predict the time, we need average times for common actions.

Several methods of keystroke level:

- Macro level: Beta -Chi design, Open System task Analysis
- Intermediate level: Hierarchical Task Analysis
- Micro level: G.O.M.S., Cognitive Complexity Theory

Another action analysis method is back of envelope analysis. This method attempts to gain an overall picture of the system with no attempt to make a detailed prediction. We could list a natural series of actions to complete a task (imagine that you are explaining the task to someone).

You can get simple answers to these types of questions without a detailed second analysis. At the level you need you can assume that every action takes about 2 seconds. This can be useful for quick comparisons. This type of analysis can be useful when trying to amend or augment an existing interface.

Heuristic Evaluation

A method based on simple questions about the interface. Nielson and Molich proposed nine such questions that are in essence simple heuristics about interfaces

2.4 Expert Evaluation

It uses experts in the field of HCI to provide detailed reports based on their experiences.

- Efficient: small number of experts can identify a whole range of potential problems
- Feedback: do not need prompting to give prescriptive feedback considerations

Experts should:

- Not have been involved with previous versions of the system.
- Have experience in type of system under study and of HCI
- Have a clearly defined role so that they adopt the correct perspective

The Aim of this method is to obtain a common set of factors that address the most important interface problems. This method can be achieved by using a combination of methods:

- Structured reporting that is easy to analyze but requires time to categorize problem and the structure can inhibit spontaneous suggestions.
- Unstructured reporting which invite spontaneous comments but is more difficult to analyze and to categorize common problems.
- A predefined category that is very easy to analyze but this method completely inhibits spontaneous comments and almost certainly miss problems not already categorized.

3. Evaluation with emergent features

Configurality and emergent features are related concepts that are influencing much basic work in form perception. When simple visual elements are put together, new features sometimes appear that were not present in the original elements. For instance, four straight lines can be joined to form a rectangle, which produces the features of closure, area, and possibly symmetry. These are called emergent features because they arise from the arrangement of the lines without being identifiable with any single line. The rectangle itself would be called a configural display because the configuration of its elements produced these emergent features.

Unfortunately there are no widely agreed upon definitions of emergent features or configurality. In some cases researchers have tried to develop a priori definitions that are driven by objective properties of the stimulus rather than by experimental outcomes. Researchers have also speculated that the perception of configural properties or emergent features may be primitives of vision mediated by special cerebral feature detectors: “emergent features may themselves be directly registered by specialized feature detectors at an early stage of processing”. In other cases researchers have striven toward empirical definitions whereby a particular pattern of experimental results determines post hoc whether a stimulus is configural. Finally, researchers have made recourse to phenomenological experience: emergent features should “pop out” of a visual field without serial search. Unfortunately, these different types of definitions do not always agree about what should be called a configural display or an emergent feature. Thus we will have to use a combination of definitions when presenting the justification for the present research.

Our goal is to exploit the emergent features present in the online monitoring tool to convey information about processes that vary over time. First, to use emergent features effectively for display design, knowledge of the general user’ s task and the goal of the interface is crucial. In this part, we consider the definitions and lists of emergent features that are using to measure the complexity of a user interface.

3.1 The Hick-Hyman law

It is intuitive that the increase of uncertainty or complexity among alternatives will increase user response time (RT) of human because the more complex decisions are required to choice the right alternative or target in each situation [7,8]. In addition, RT will increase when the responses that should be performed by human are difficult. Between these two factors, the uncertainty can be quantified by the amount of information (H_s) that should be processed by human, and it can be calculated by the following equation [11-13].

$H_S = \log_2 N$, where N is the number of equally likely alternatives.

$H_s = \sum_{i=1}^{n} p_i \cdot \log_2 \left[\frac{1}{p_i} \right]$, where p_i is the probability of occurrence of event i.

It is noted that the last equation will increase H_s for low probability events, and if all events occur with the same probability then the last equation is equivalent to the first one. From the above equations, it is quite obvious that low probability events will reduce it. In other words, if a signal is used to convey information that can characterize specific event then the uncertainty of the event increases when the amount of information conveyed by that signal increases. For example, if p_i equals to 1 then there is no uncertainty. In contrast, if p_i tends to zero then the uncertainty of the event tends to infinity. Thus, it is very reasonable to say that response time will be proportional to the amount of uncertainty. Or, it can be said that the increase of the number of alternatives will make the choice of the right alternative more difficult. This theory is known as the Hick-Hyman law or uncertainty principle, and it can be summarized as follows.

"RT increases with the uncertainty for the judgment to be made, and RT can be expressed by empirical correlation such as $RT = C_1 + C_2 H_S$, where C_1 and C_2 are constants and H_s is the uncertainty due to the number of alternatives."

According to the Hick-Hyman law, RT consists of two parts. The one is the time taken to process the stimulus for finding the right alternative among all alternatives, and the other is the time taken to execute the response. Among them, the constant C_2 represents the amount processing time that result from the amount of uncertainty to be processed by human. On the other hand, the constant C_1 describes the sum of those processing latencies that are unrelated to the amount of uncertainty.

In the view of information search tasks, the Hick-Hyman law plays an important role to evaluate HCIs provided in CRT based information display systems because such tasks can be two types of human task that are selection type and retrieval action for display information. Based on this consideration, it can be said that the increase of information amounts in a given user interface means the increase of complexity that interferes selection task for the right alternative. Thus, the complexity of a given user interface can be represented by the uncertainty that is evaluated by the Hick-Hyman law.

However, a lot of study in this area reveals that there are five additional factors that are influence to RT, and they were not easily quantified by information theory. They are known as distinctiveness due to emergent features, repetition effect, response factors, practical and stimulus-response compatibility. Generally, it is well know that RT is lengthened when an alternative is made less distinguishable from another ones. According to related research, distinctiveness is inversely proportional to shared emergent features. For example, the numbers '4' and '7' are quite distinct, but the numbers '621834' and '621837' are quite similar because the numbers in the later pair are largely shared by the same digit alphabets. Therefore, the distinctiveness should be emphasized for information search task.

3.2 Distinctiveness measure

In this study, the calculation of distinctiveness would be processed by the excess entropy concept. The term of the excess entropy is suggested to evaluate the modularity between subsystems based on the entropy concept, because the entropy is widely adopted to assess the amount of information. The brief description for the excess entropy is as follows.

If there is a system S that consists of two subsystems A and B, and these subsystems have the relations such as $A \cap B = \varnothing$ and $A \cup B = S$ then it can be said that these two subsystems are totally modularized. In other words, subsystem A and B are independent because they do not share any kinds of information. In this case, the excess entropy is zero. Possible types of information that are shared are: (1) relationships that must exist prior to the execution of a subsystem for its effect to be realized, (2) data structure or data values and (3) environment in which a subsystem is executed. However, some information will be shared between subsystem A and B in most cases. Such shared information can be easily understood from Fig. 3-1.

<Figure 3-1. Shared information by terms of Venn diagrams>

In Fig. 3-1, H(A) and H(B) mean the amount of information that contained by a subsystem A and B, respectively. H(S) means the amount of information that contained by a system S, and $H(A \cap B)$ means the amount of information that are by a subsystem A and B. In this situation, the excess entropy (C(S)) means the amount of shared information, and it can be calculated by the following equation.

$$C(S) = H(A) + H(B) - H(S)$$

From the above equation, it can be seen that the excess entropy can be used as a measure to evaluate the degree of independence between subsystems, because the increase of the excess entropy means the decrease of information contained in each subsystem that represents the characteristics or modularity of each subsystem.

Thus, distinctiveness of a subsystem A can be represented by signal to noise ratio because distinctiveness is inversely proportional to shared information. Accordingly, distinctiveness (D) for a subsystem A of a system S can be defined as follows.

$$D(A) = \frac{\text{signal}}{\text{noise}} = \frac{H(A) - C(S)}{H(A^C) - C(S)}$$

$$C(S) = H(A) + H(A^C) - H(S)$$

Calculation of information amount in any user interface is archived by the concepts of second order entropy and emergent features. Emergent features are shown in Table 3-1. These features are collected to related research papers.

<Table 3-1. List of emergent features>

For the second order entropy, nodes are considered to be equivalent if they have the same number and type of neighbors within one arc distance. This means that the amount of information to describe an asymmetrical graph will increase because each node position becomes more unique. Furthermore, second order entropy will increase when the size of the graph increase, because the asymmetry of the graph generally will increase. Therefore, the second entropy can represent complexity resulting from the size of graph that have the same mean as a user interface. Arbitrary structured graph are shown in Fig. 3-2. And, its equivalent one arc distance table is in Table 3-2.

<Figure 3-2. Two arbitrary structured graphs>

<Table 3-2. Identified classes of Figure 3-2 to calculate the second order entropy>

Again, the second order entropy of graph G and G’ can be calculated as follows.

$$H_G^2 = -\sum_{i=1}^{5} p(A_i)\log_2 p(A_i)$$
$$= -\left[\left(\frac{1}{7}\log_2\frac{1}{7}\right)+\left(\frac{2}{7}\log_2\frac{2}{7}\right)+\left(\frac{1}{7}\log_2\frac{1}{7}\right)+\left(\frac{2}{7}\log_2\frac{2}{7}\right)+\left(\frac{1}{7}\log_2\frac{1}{7}\right)\right]$$
$$= 2.236$$

$$H_{G'}^2 = -\sum_{i=1}^{6} p(A_i)\log_2 p(A_i)$$
$$= -\left[\begin{array}{l}\left(\frac{1}{7}\log_2\frac{1}{7}\right)+\left(\frac{1}{7}\log_2\frac{1}{7}\right)+\left(\frac{2}{7}\log_2\frac{2}{7}\right)\\ +\left(\frac{1}{7}\log_2\frac{1}{7}\right)+\left(\frac{1}{7}\log_2\frac{1}{7}\right)+\left(\frac{1}{7}\log_2\frac{1}{7}\right)\end{array}\right]$$
$$= 2.521$$

In software engineering, these entropy measures are used to evaluate not only program control graphs but also data structure graphs because a data structure can be represented by a hierarchical directed graph. This graph is called data structure graph. In data structure graph, the nodes mean data entities and the arcs indicate a relationship between nodes. In addition, the successor nodes represent the different data fields within their predecessor node, and the base level of a graph represents the base data type such as real, integer, character, boolean or other user defined data types. For example, Table 3-3 shows a data record and its data structure graph.

<Table 3-3. An example of data structure graph>

Based on the data structure graph, its complexity can be calculated by the first order and the second order entropy. The meaning of the first and the second order entropy for a data structure graph are equivalent to that of a program control graph. That is, the first and the second order entropy can be used to evaluate the regularity and the size of a data structure graph, respectively. However, the first order entropy is less meaningful in case of a data structure graph evaluation, because complexity of a data structure graph may be more affected by the size of it.

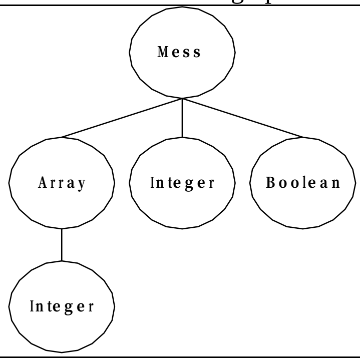
Table 3-1. List of emergent features

Emergent feature	Content
Color	One of the most effective visual attributes for coding information in displays
Symbol	Including the numerical, alphabetical, and imagery characters
Shape	A geometrical figure (Rectangle, Circle, Line, Dot, etc)
Layout and grouping	An area based on the context or the intentional subspace
Highlight	Blinking or animated visual attribute

Table 3-2. Identified classes of Figure 3-2 to calculate the second order entropy

Graph	Node	Neighbor Nodes
G	{a}	{b,c}
G	{b,c}	{a,d}
G	{d}	{b,c,e,f}
G	{e,f}	{d,g}
G	{g}	{f}
G’	{a}	{b,c}
G’	{b}	{a,g}
G’	{c}	{a,d,e}
G’	{d,e}	{c,f}
G’	{f}	{d,e,g}
G’	{g}	{b,f}

Table 3-3. An example of data structure graph

Data record	Data structure graph
Mess = RECORD X : array of integer Len : integer Side : boolean End	

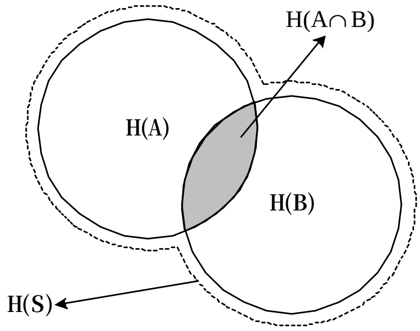


Figure 3-1. Shared information by terms of Venn diagrams

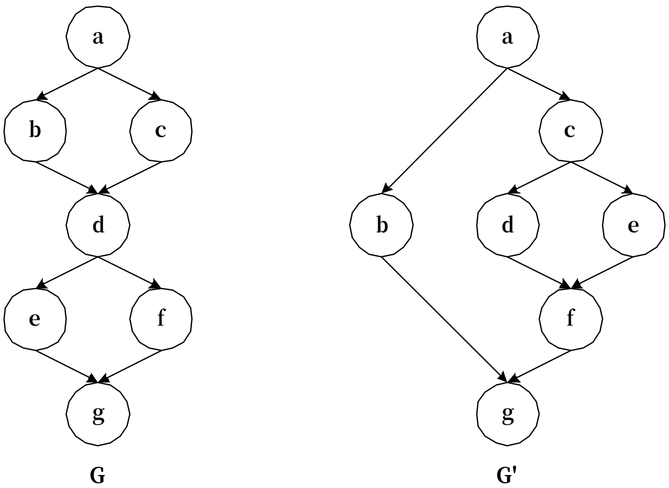


Figure 3-2. Two arbitrary structured graphs

4. Design strategy

Traditional design process for developing some user interface is feed-forward method in Fig. 4-1. This process has some disadvantages. First, it is difficult to optimize a user interface to the designer' s requirements because most developers do not consider the evaluation method and user' s search task. Second, design requirements are qualitative descriptions so that its implementation is difficult work.

<Figure 4-1. Traditional design process>

This study will propose a new approach for the design process as a feedback method using simple graph theory and its expansion rule. And this approach has quantitative feedback information to the design step for more efficient understanding of changing option. This new approach is shown in Fig. 4-2.

<Figure 4-2. Scheme of new design strategy>

For applying this new approach, it is necessary to define some concepts. Design step can be divided into design option and design requirement for any design of user interface. Design option means that it is part of changing user interface to fit the design requirement. For example, if we made some user interface then we could add color property to specific areas. Design requirement means the ultimate goal of development. All of sub items of user interface are not important to its real end-user. Design requirement can decide important parts of user interface and total performance by terms of user response time. However, all qualitative information in design process should represent directed structure graph as mentioned in chapter 3 in the form of quantitative information that has some entities. Quantitative feedback information is the most important advantage with this new approach. Optimizing method can be implemented by distinctiveness and its related user response time correlation or correlation database. The explanation of relationships between distinctiveness and user response time is mentioned in previous chapter. The reasons that use user response time are as follows.

- A criterion of overall system performance
- Summative and/or representative factor for usability
- Ease of measure

Furthermore, RT is eventually produced by information processing in human brain with state transition of machine or user interface.

Computer display includes various programs by user's some action. Various programs have each specific user interface with various visual elements called as emergent features. Therefore, all user interfaces represent directed structure graph and adding and/or removing visual elements can change them. If we change a user interface, distinctiveness of sub items will be changed and it is affected to user response time. Now new approach of design process will be explained.

4.1 New design process

RT database (or RT correlation) is the main part of new design process. Without RT database, quantitative feedback information has only numerical values without realistic meaning called as distinctiveness. This information is useless to developers, designers and real end-users. Fig. 4-3 shows how use RT database and the whole design process in our new approach.

<Figure 4-3. Schematic diagram of new design process>

Main process of new design process is iterative to find out optimized user interface to the design requirement. For each iteration, prototype user interface is again evaluated to calculate the distinctiveness of its sub items and design option is differently adopted. Within this iterative process, RT database is adopted to the distinctiveness value for comparison of design requirements that have time related factors. RT database is constructed by experiments with real-end user and any produced user interface.

For example, Fig. 4-4 can be represented as directed structure graph Fig. 4-5. And its distinctiveness values are shown as Table 4-1.

<Figure 4-4. Sample user interface for turbine cycle state>

<Figure 4-5. Directed structure graph of Fig. 4-4>

<Table 4-1. Distinctiveness value of Fig. 4-5>

In Table 4-1, we can find out that 'LP1 입구압력' and 'HX4 입구온도' have largest distinctiveness values. This fact explains that both sub items are best visual item to real end user so that users react to those items with minimum response time in that user interface. If those items are most important part, there are no changing options to improve user interface. Otherwise distinctive values of those items should be decreased by changing options for improvements.

RT database could be adopted into the result of Table 4-1 as shown in Table 4-1. Generally RT database is extracted from various ways by experiments with any user interface. The type of RT database is not specific because it is sufficient to supply some correlations with raw numerical data. But the type of RT correlations should follow the equation of Hick-Hyman Law.

<Table 4-2. Correlated RT with Table 4-1>

In Table 4-1, RT fields are calculated by RT correlation that is experimental results with real end user as following equation.

$$RT = 0.403 - 2.247 g \log(Distinctiveness)$$

RT fields of Table 4-1 are similar to each value because the user interface has few emergent features and all sub items are almost commonly visual elements. Park's experimental results agree with Hick-Hyman law. We can change this sample user interface with some design options. For example, we can add color property to item I1, I2, I3 and I4 and text property to item I2 and I4 with Fig. 4-4. As mentioned, design requirements are necessary to optimize user interface with those design options. Most adoptable design requirements are as follows.

DR1 - Min. user response time of 'LP1 입구압력'
DR2 - Min. user response time of 'LP1'
DR3 - Min. user response time of 'LP2'

DR4 - Min. user response time of ' LP3'

After expansion process with such design options, design requirements and prototype user interface, the result can be put in the order of design requirements as shown in Table 4-3.

<Table 4-3. Results of new design process>

4.2 Expansion

If all required design information is represented to directed structure graph, the graphs of design option and prototype user interface will be combined in the case of all possible meaning. This study realized evaluation and expansion algorithm using C language in Unix environment for automatic expansion process. Simple process of expansion is shown in Fig. 4-6.

<Figure 4-6. Flowchart of expansion algorithm>

If the nodal graphs of prototype user interface and design option are provided, we can expand prototype user interface as following algorithm.

1. Evaluation of current user interface
2. Assign relations between additional nodes and current graph
3. Calculation of entropy of total information
4. Calculation of distinctiveness of each sub item
5. If there is no possible assign condition then go to 6 step otherwise go to 2 step
6. Select appropriate case by design requirement

The method of combination in this study is sequential permutation in mathematics.

Table 4-1. Distinctiveness value and Correlated RT of Fig. 4-5

	Distinctiveness	RT
LP1 입구압력(I1)	0.03758	3.60502
LP1	0.01892	4.27471
LP2	0.0288	3.86491
LP3	0.0288	3.86491
LP4	0.0288	3.86491
LP5	0.0288	3.86491
LP5 출구압력	0.02809	3.88903
Condenser	0.02521	3.99459
HX4 입구온도	0.03758	3.60502
HX4	0.01892	4.27471
HX3	0.0288	3.86491
HX2	0.0288	3.86491
HX1	0.0288	3.86491
HX1 출구엔탈피	0.02809	3.88903

Table 4-3. Results of new design process

Design Requirement	Item	(Previous) RT (sec)	(After) RT (sec)	Assigned Option
DR1	I1	3.6050	3.1864	(Color ⇒ I1, I4) (Text ⇒ I4)
DR2	I2	4.2747	3.5447	(Color ⇒ I1, I2) (Text ⇒ I2, I4)
DR3	I3	3.8649	3.4462	(Color ⇒ I3, I4) (Text ⇒ I4)
DR4	I4	3.8649	3.5880	(Color ⇒ I1, I4) (Text ⇒ I2)

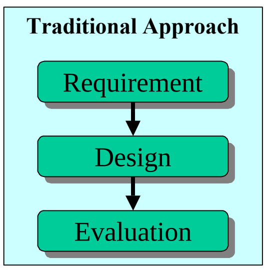


Figure 4-1. Traditional design process

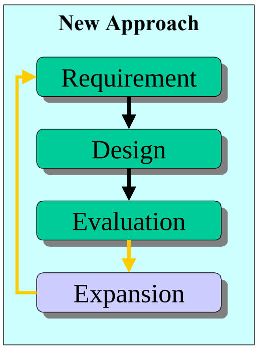


Figure 4-2. Scheme of new design strategy

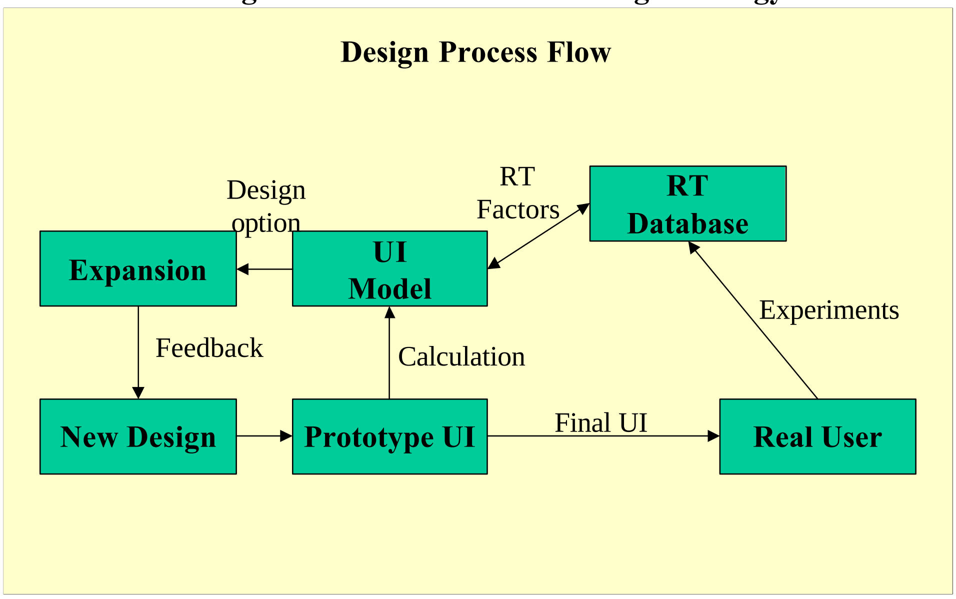


Figure 4-3. Schematic diagram of new design process

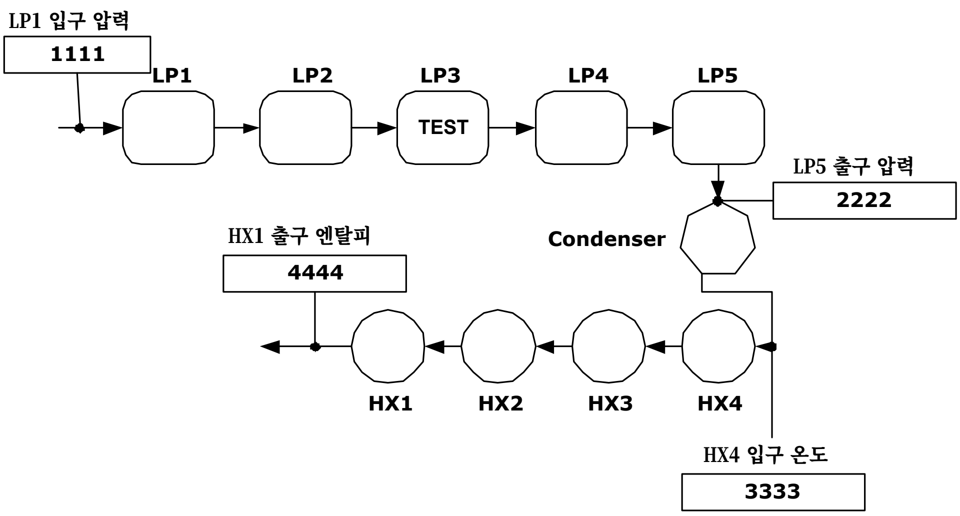


Figure 4-4. Sample user interface for turbine cycle state

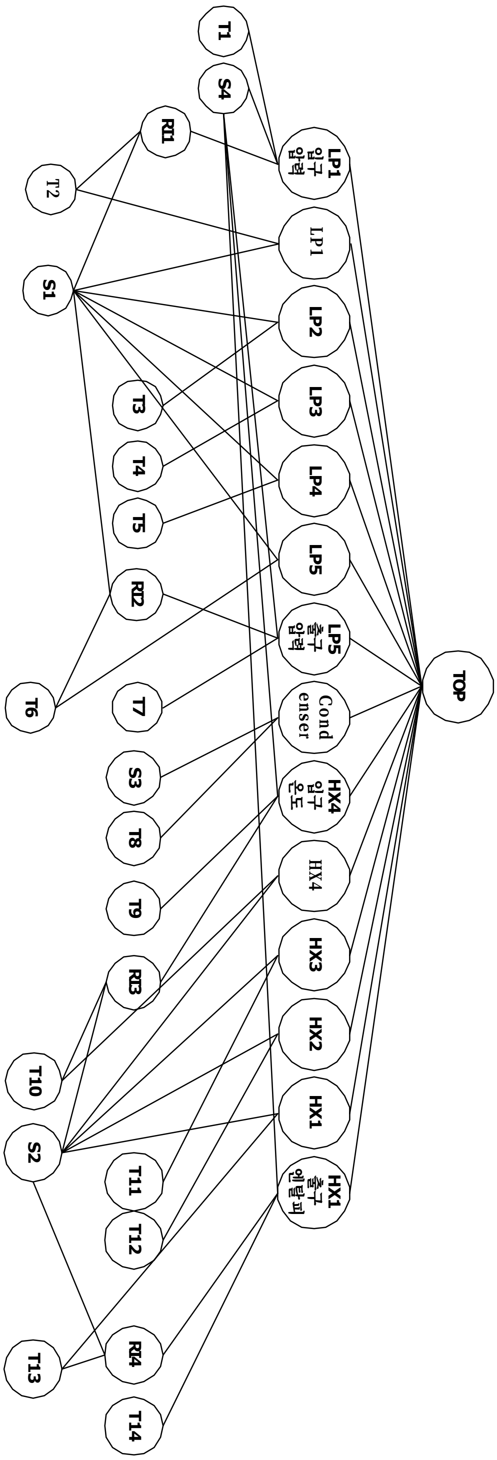


Figure 4-5. Directed structure graph of Fig. 4-4

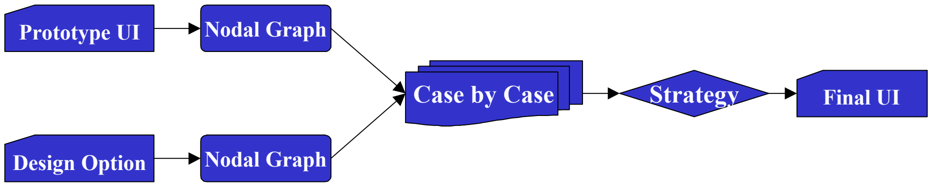


Figure 4-6. Flowchart of expansion algorithm

5. Results and Discussion

Basically, one of primary roles of information display systems is the provision of information or cues that are useful for the human task with HCI. For better HCI, it is appropriate to construct user interface using new design process alternative to traditional design process. In this study, to propose new design process, user response time and emergent features are used. In the measure of information amount, distinctiveness is calculated by the ratio of different features for specific visual item to total information amount based on the emergent features.

The important merit of developed design process is that it can consider the realistic quantitative feedback information from original prototype user interface to design step. Shapes, colors, symbols, layouts and highlights are used for emergent features. Such features will enhance the description of any user interface for directed structure graph. The calculation of distinctiveness is improved from other past research because there are more emergent features to measure the information amount. But it is difficult to describe the relative, content and type of information with current emergent features.

In directed structure graph, the nodes mean data entities and the arcs indicate a relationship between nodes. In addition, the successor nodes represent the different sub items within their predecessor node. Therefore, even if all nodes in a directed structure graph have different value (i.e. content), they are treated as the equivalent nodes if they are represented using the same base emergent feature.

Although distinctiveness measure has some drawbacks in representing content or related information, and a lot of experimental validations should be additionally performed to ensure the appropriateness of this measure method, it seems to be useful for the calculation of HCIs.

Expansion algorithm can find all possible combinations with design options and prototype user interface. Its implementation method is sequential permutation so that some shortcomings are produced at pure permutation method. For example, the algorithm produced equivalent cases of user interface in some extent. Therefore total time of calculation and expansion is longer than optimized processes. Another shortcoming is the guarantee level of suggested feedback information at the result of expansion process. It is natural to validate automated solutions with real end user testing by checklist method.

RT database has been constructed from Park's experimental result. It contains the relationship between user response time (or searching time) and distinctiveness so that correlating equation can be produced by RT database as form of Hick-Hyman law. The correlation gives a realistic entity to distinctiveness value to produce quantitative feedback information in expansion and feedback steps.

Every part of new design process has been produced in the form of programming language. And every part of information entity of user interface had the form of directed structure graph or nodal graph. Therefore, other researchers can easily implement the proposed design process. But it is necessary to understand the representing rule of design option, design requirement and prototype user interface because the code has been developed under the Unix environment based on textual display.

However, the results of this study proposed that new design process with quantitative feedback information is better approach than traditional design process to meet the design requirements.

6. Conclusions and Recommendations

6.1 Conclusions

In this study, new design process methodology of HCI has been proposed and applied to a user interface of turbine cycle diagram. From the results of new approach, the following conclusions can be summarized.

1) New design process that can extract an optimized user interface in the view of design requirements with design options is developed.
2) Distinctiveness measure that can evaluate an user interface in common personal computer program is improved from past measure method.
3) Optimized user interface as the result of new design process is generally agreed with the results of intuitive considerations.
4) Emergent features can reasonably identify most parts of any user interface.
5) Calculated RT from RT database and distinctiveness is generally agreed with the results of other researches.
6) Quantitative feedback information from calculation and expansion process is very realistic value for designers and developers of a user interface.
7) Developed codes for calculation and expansion seem to be useful for easy testing of various environments.

6.2 Recommendations

In addition to the above conclusions, the followings are recommended for further study. To encapsulate the relative or content effects to the distinctiveness measure are strongly recommended. Some additional experiments to consider the result of this approach are recommended to enhance the reliability of RT database and its correlations. Some additional emergent features to describe a user interface are necessary to improve the reflection of real user interface. Alternatives of expansion algorithm in this study are recommended for efficient search of optimized user interface.

REFERENCES

[1] Jin Kyun Park and Soon Heung Chang, "A Study on Dynamic Evaluation Methods for Human Machine Interfaces In Advanced Control Rooms" Ph.D. Thesis, KAIST, 1998.

[2] Larry E. Wood, "User Interface Design" CRC Press, 1998

[3] Lindsay W. MacDonald, "Using Color Effectively in Computer Graphics", *IEEE Computer Graphics and Applications*, 1999

[4] Arvind S. Ramakrishnan, Robert L. Cranston, Armida Rosiles, Dan Wagner and Anil Mital, "Study of Symbols coding in airway facilities", *Int. Journal of Industrial Ergonomics*, Vol. 25, pp. 39-50, 1999

[5] Myung Hwan Yun, Sung H. Han, Sang W. Hong, Jiyoung Kwahk and Yong H. Lee, "Development of a systematic checklist for the human factors evaluation of the operator aiding system in a nuclear power plant", *Int. Journal of Industrial Ergonomics*, Vol. 25, pp. 597-609, 2000

[6] B.J. Oommen, K.Zhang, and W. Lee, "Numerical Similarity and Dissimilarity Measure Between Two Trees", *IEEE Trans. on Computers*, Vol. 45, No. 12, pp. 1426-1434, 1996

[7] Eric Lee and James MacGregor, "Minimizing User Search Time in Menu Retrieval Systems", *Human Factors*, Vol. 27, No. 2, pp. 157-162, 1985

[8] Carol A. Simpson and Douglas H. Williams, "Response Time Effects of Alerting Tone and Semantic Context for Synthesized Voice Cockpit Warnings", *Human Factors*, Vol. 22, No. 3, pp. 319-330, 1980

[9] Mary A. Buttigieg and P. M. Sanderson, "Emergent Features in Visual Display Design for Two Types of Failure Detection Tasks", *Human Factors*, Vol. 33, No. 6, pp. 631-651, 1991

[10] Elizabeth S. Bednall, "The Effect of Screen Format on Visual List Search", *Ergonomic*, Vol. 35, No. 4, pp. 369-383, 1992

[11] E.J. Weyuker, "Evaluating software complexity measures", *IEEE Trans. on Software Engineering*, Vol. 14, No. 9, pp. 1357-1365, 1988

[12] A. Mowshowitz, "Entropy and the complexity of graphs: I. An index of the relative complexity of a graph", *Bulletin of Mathematical Biophysics*, Vol. 30, pp. 175-204, 1968

[13] J. S. Davis and R. J. LeBlanc, "A Study of the applicability of complexity measures", *IEEE Trans. on Software Engineering*, Vol. 14, No. 9, pp. 1366-1372, 1988