# A taxonomy of visual feedback for interactive systems

Jaime Muñoz Arteaga
Ciencias computacionales
Instituto Nacional de Astrofísica Optica y Electrónica
Luis Enrique Erro No. 1
Sta. María Tonantzintla
72840 Puebla, México

*Abstract:* - In an interactive system, the term of visual feedback is applied to any graphic form of communication directed from the application towards the users. The visual feedback is predominant in current interactive systems, but it is difficult for a designer to identify the pertinent visual feedback for this type of system. This paper addresses these issues with a taxonomy of visual feedback which is described on detail with meaningful examples. The goal is to offer a designer a high level description of visual responses of an interactive system independent of any graphical environment.

*Key-Words:* - Interactive system, visual feedback, taxonomy, interaction styles and architectural models

## 1 Introduction

In an interactive system, the term visual feedback (rendering or system's visual response) is applied to any graphic form of communication from the application towards the user. Essentially, the goal of visual feedback is twofold: the visualization the internal state of system and the visualization of user actions.

The visual feedback appears under different forms (e.g. windows, icons, maps and error messages) of user interface with the purpose to establish the most natural visual communication with the user. For example in a system with WIMP (Windows, Icons, Menus and Pointers) interfaces the interaction objects support divers levels of visual feedback and divers interaction styles such as indirect manipulation as well as direct manipulation.

Although a lot of work has devoted to study divers aspects of presentation of interactive system; the visual feedback has received no much considerations. Even if the visual feedback is predominant in most current interactive system, the designer doesn't account with any mean to know about the visual information required for guide user actions. In fact, the diversity of visual feedback forms require its classification. This works deals with the specification of visual feedback in order to identify the type of system information given to user. The goal is to offer a designer a high level description of visual feedback independent of any graphical environment.

The first section of this paper starts to explain the architectural aspects of visual feedback. The next section presents a taxonomy of visual feedback according to its function in the application. The next three sections describe in some detail every categories of the taxonomy and some examples of interactive application are showed in order to illustrate how these categories are taken into account by the taxonomy of visual feedback proposed here.

## 2 Architectural aspects

It is quite difficult to develop the external aspect of system without being immediately stuck into the inherent relation with the internal aspects of system. In order to structure the information the HCI community has devoted a lot of work to structure the information en terms of generic architectural models for interactive systems. One of the first and most famous is the Seeheim model (Fig. 1) that splits the user interface of an interactive system in three components: *Presentation*, *Dialogue Control* and *Application Interface Model.*

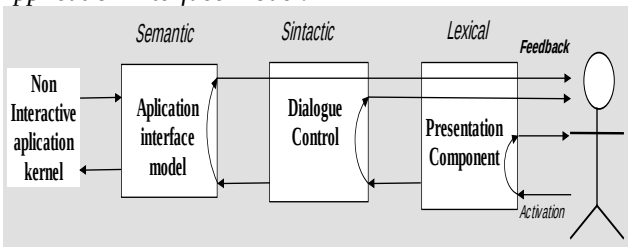


Fig. 1.Seeheim model [6] making emphasize in the three linguistic level of visual feedback.

The *Dialogue* handles the syntactic aspects of the interaction and is responsible for the dynamics of the system. The *Application Interface* provide a semantic interpretation to syntactical information received for the dialogue component. The *Presentation* handles the lexical aspects of the interaction, in input (communication user -> system) and in output (communication system -> user). These two communication channels, that we call respectively activation and visual, are fundamentally different, as activation is event-based while visual feedback is state-based [1]. Even if the activation and feedback are different, they are close related because for only one user action, it is possible multiple levels of information of feedback that could come from divers software components of an interactive system (see Fig. 1).

## 3 Taxonomy of visual feedback

Most work in Human-Computer Interaction agree that the feedback is indispensable for the dialogue between the user and the system [12] [2]. In the same way that the Seeheim model this work consider the visual feedback as a language

to guide the user actions and inform about the internal state of an interactive system. Then, it is possible classify the visual feedback with different abstraction levels of visual feedback according to the linguistic nature of dialog between the user and an interactive system (see Fig. 2):

- *The semantic category* defines the visual response in function of internal state of objects that belong to the functional core of the system.
- *The syntactic category* consists of a set of rules by which primitive output information can be composed or joined to form a visual ordered sequences to user.
- *The lexical category* refers to output information derived from interactive objects manipulated directly by the physical devices in particular mouse and keyboard.

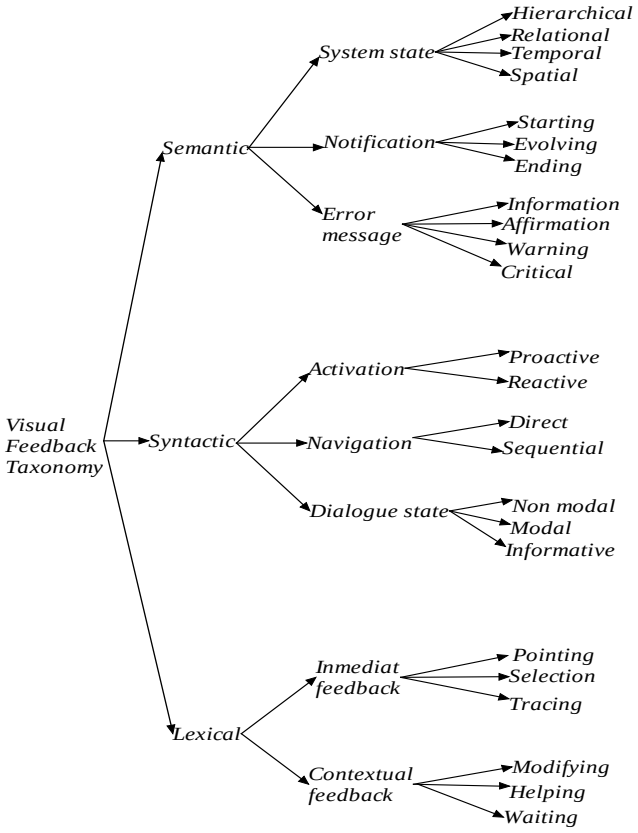


Fig. 2. Taxonomy of visual feedback.

The taxonomy of Fig. 2 illustrates that every linguistic category of visual feedback could have different levels of granularity and the information to display to user could be located throughout the architectural components of an interactive system. Then, it is possible to map this information with the software modules in order to design the visual response of the interactive system to develop. In addition, the taxonomy proposed here, it considers that the visual feedback could be represented by a large number of interactions styles from textual to direct manipulation.

Next sections of this paper are devoted to described every category of this taxonomy with simple but meaningful examples. The graphical environment of WindowsNT was used here only to show the graphic examples but the criteria of classification could remain valid for any platform.

# 4 Semantic visual feedback

The ultimate goal of an interface is to provide means for a user to interact with semantic objects that belong to the functional core of the system. The user actions may trigger the creation or deletion of objects, obtain information on the object's state, or change its state.

## 4.1 System state

Any interactive system has to display part or all of the system state. In an object-based world, the application may wish to display any public attribute of objects belonging to the functional core, or the result of any method call performed on these objects or any functional combination thereof. The classification proposes here for the system state feedback is in the spirit of Shneiderman's criteria [13]:

### 4.1.1 Hierarchical

It is the information associated to system state which is composed by elements related hierarchically. It is the case of a files system, a hierarchy of classes of objects, a taxonomy of animal kingdom etc.

### 4.1.2 Relational

It is the information associated to system state which is composed by elements linked by a relation but the structure that they form isn't hierarchically. It is the case for visualization the links between documents hypertext, the nodes of computer science networks and the exchange of messages between the objects of a program;

### 4.1.3 Temporal

Temporal feedback covers the visualisation of information of system state that change continually during a period of time. Example is the visualization of changes of a value that can represent the hour or the date of a computer system.

### 4.1.4 Spatial

It is the information associated to the internal state of system and require a spatial visualization. Here it is possible to cite a cloud of points, a matrix or vector of data for example.

## *4.2* Notification

The state of objects that belong to the functional core may change due to causes independent from the user interface (e.g. when a software monitors some kind of physical system). This change of state is spontaneous in the user's point of view, because it is not triggered by a user action. This kind of changes needs also to be made apparent to the user [4]. The system can notify such changes either at the start, during or at the end of the execution of an internal operation of a system.

#### 4.2.1 Starting notification

This visual feedback inform the nature of operation just before its execution. This type of feedback appear at the beginning of the execution of an operation of core function, and it is recommend to used when the operations are longer and the consequences of the operation can be hazardous.

#### 4.2.2 Evolving notification

This type of visual feedback appear during the execution of an operation of core function when the period of time of this execution is more than 10 seconds [15].

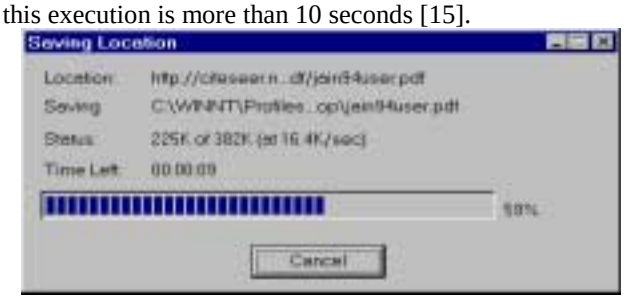


Fig. 3. Notification messages about the evolution of downloading a file.

The evolving notification must assure and give the control to user on a long process operation. For example in Fig. 3, the user could stop at any moment the downloading operation. In addition this message give the status and the reaming time to achieve this operation.

#### 4.2.3 Ending notification

This type of feedback appear at the end of the execution of an operation of core function. This visual feedback inform the status of finished operation. It is the case of the status information at the end of disk-format operation, indexing a data base and compiling a program.

#### 4.2.4 Error messages

A user-initiated action may trigger a call to the functional core that fails to complete properly, and the call will usually return some form or error status or exception. These error reports must be notified in some way to the user, providing as much information as possible on the nature of the error, its potential cause and possible cures. The taxonomy of visual feedback of Fig. 2 distinguishes four types of error messages.

#### 4.2.5 Information message

The purpose of an information message is to guide the user actions after one interruption of data processing or one modification of parameters.

#### 4.2.6 Affirmation message

This type of messages must be used for operations that require the user's validation. This type of messages is used frequently before an attempt of destruction or of overwriting a user's document. For example the message of Fig. 4 asks the confirmation of user to overwrite over an already existed file.

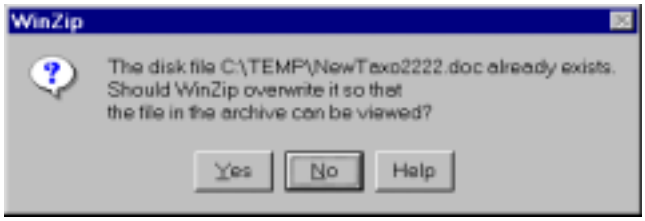


Fig. 4. An affirmation message when the user try to overwrite a file.

#### 4.2.7 Warning messages

The purpose of this feedback is alert to user about an irreversible operation could be executed by the interactive system. A typical message warning messages appear when the user execute the disk-format operation.

#### 4.2.8 Critical error messages

This information is given to user when the current state of the system is not recoverable. The type errors messages generally are caused by a badly function of hardware (e. gr. physical fault in an input device) and some critical error of software (e. gr. division by zero).

## 5 Syntactic visual feedback

The category of *syntactic visual feedback* identify any form of visual feedback that aims at keeping the user informed of the evolution of the interaction that is going on with the system. The objective of this kind of feedback is display the evolution of communication state between the user and system.

### 5.1 Activation

The activation visual feedback corresponds to the actions that are allowed by the system at any given moment. As the user-system dialogue proceeds, the set of legal actions offered to the user changes dynamically. We call the set of valid actions offered by the system at a given moment *the interaction space* of the user**.**

#### 5.1.1 Proactive feedback

Proactive feedback displays the space of interaction to the user at all moment of the interaction. The interaction space needs to be made apparent to the user, and this is usually done by greying out any widget or menu item that might be used in the interface to trigger a currently illegal action. The technique that is applied frequently is to greying out any widget or menu item that might be used in the interface to trigger a currently unavailable service.


Fig. 5. Rendering the interaction space

Applying the previous technique to the application of Fig. 5, the set of services available to the user is those that are associated to button "Close", "Refresh" and "Help". This technique for visual feedback the interaction space is quite common, but presents the drawback of providing very little information on the reasons why a particular action is

currently disabled, and gives even less clues on the possible ways to make it available. Other forms of interaction space visual feedback that does not suffer these drawbacks can be devised in [11].

#### 5.1.2 Reactive feedback

A reactive feedback differs of proactive feedback by the notion of reactivity. For example when the mouse cursor pass over a widget, this widget reacts changing from disable to enable. This technique is very common in the current hypertext and hypermedia applications.

### 5.2 Navigation

Any complex application faces the need to partition its presentation into several screens or windows in order to prevent screen cluttering or to group information into logical units. This induces the need to offer the user some means to navigate the application's screens, and of course to render this navigation. It must be emphasized, that the navigation between windows is a coarse grained dialogue between application and user, while interaction within a window can be considered as fine grained dialogue [3]. This work focuses only the coarse grained dialogue feedback and for this distinguish the sequential navigation of direct navigation.

### 5.3 Sequential navigation

It allows to user navigate consecutively between the windows of an interactive application either toward back or forth from current window. This feedback take into account the previous windows visited by user during a work section. Nowadays, this technique is very useful for the current browser of internet which have a button "Back" for to go to previous page and a button "Forward" for to go to next page.

#### 5.3.1 Direct navigation

Unlike the sequential navigation feedback allow the user to navigate over any windows of the application.

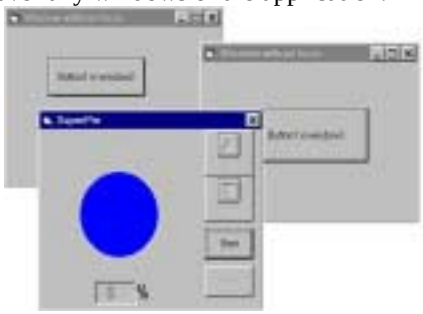

Fig. 6. The focus window an example of direct navigation.

Most often, the attribution of focus to one window is handled transparently by the window manager. The window that hold the focus is displayed with normal colors while windows without focus are either hidden (not displayed) or grayed out if they are still visible. However, in that case part of the graphical representation can be partly recovered by other windows. An example of such feedback is given in Fig. 6.

### 5.4 Dialogue state

Other kinds of information that belong to the state of the dialogue may be of interest to the user. It is sometimes necessary to show the user that he/she is in a given state of interaction (e.g. by showing information in the status bar of window). Values of this kind are maintained by the dialogue component of the application, but are not always directly related to the activation or deactivation of widgets.

#### 5.4.1 Modal feedback

Modal feedback occur when the current window must be closed in order to return of remaining windows of system. This is common when the user customize the configuration for a driver of a physical devices integrated to system.

#### 5.4.2 Non modal Feedback

Unlike the modal feedback, the user don't have any restriction to access any interactive component of user interface.

#### 5.4.3 Informative feedback

The informative feedback lets user know that he/she is in a given status of interaction. For example the number of times the user has accessed any particular function, or the number of trials remaining for a password authentication, etc.

## 6 Lexical visual feedback

Current user interfaces increasingly adopt a direct-manipulation style of interaction, giving the user the illusion of directly acting on the objects of interest rather than indirectly accessing them through command buttons and data-entry widgets. Direct manipulation require immediate feedback and contextual feedback, the management of these feedbacks calls for a much finer grained dialogue modelling than is required for conventional indirect manipulation interfaces.

### 6.1 Immediate feedback

Modern user interfaces increasingly adopt a direct-manipulation style of interaction, giving the user the illusion of directly acting on the objects of interest rather than indirectly accessing them through command buttons and data-entry widgets. Direct manipulation requires immediate feedback of very low-level user actions such as pointing, selection and tracing. This feedback is qualified of rapid or immediate because it carried out in the computer screen between an interval of time of 0.02 seconds [14].

#### 6.1.1 Pointing

Pointing feedback visualizes the movement of cursor for all manipulation of input physical devices such as the keyboard and the mouse. For example in the case of keyboard, it is necessary a coherent representation to follow the cursor when the user enter data.

Fig. 7. Examples of pointers used frequently for notify movements with physical devices

In the case of mouse, the immediate feedback refresh the mouse cursor every time the user change the position of this device (see Fig. 7).

### 6.1.2 Selection

The selection of object(s) corresponds to feedback of user actions that take the mouse cursor over the interested object, next the user must press the left button of mouse. It is possible to use at the same time a functional key of keyboard (such as shift, control, alt, etc.) in order to get different forms of selection.

### 6.1.3 Tracing

Tracing feedback associate un element visual to movement of positioning device's cursor, the result is a trace of path carried out by the cursor. This type of feedback is frequently used by the graphical editors in order to let user draw his/her desired graphic form.

## 6.2 Contextual feedback

The contextual feedback like the immediate feedback handle the user actions coming from output devices but in addition it take into account the state of every objects that user access. This is with the aim to inform user about different kinds of contextual feedback: modifying, helping and waiting [7].

### 6.2.1 Modifying

This type of feedback gives information about user actions that try to modify the attributs of an object. A typical case of this type of feedback is when the user modify the size or move a window using the mouse, the pointer of this mouse change to a form as a line with double arrows (see Fig. 8).

Fig. 8 Examples of mouse pointers used for notify modification of object's attributes.

### 6.2.2 Helping

Helping feedback give information about the current object manipulated by the user. Hence, the novice user has the opportunity to learn quickly about the basic functionalities of system and the experienced user can work comfortably in his/her task.

### 6.2.3 Waiting

Waiting feedback appear every time that one operations of system need between 2 seconds to 10 seconds to be achieved. During this time frequently the mouse cursor change to one icon relative to time informing to user that he/she need to wait for a little time.

# 7 Related works

Divers research works in HCI are also proposed some classifications of visual feedback but they don't cover the three linguistic levels of visual feedback. For example the feedback classifications proposed by Fekete [4] and Shneiderman [13] cover the visual feedback only at semantic level. Foley's classification [5] contributes in particular to syntactic level and the works of Hudson [6] and [8] contributes notably to lexical level. Most of these works deal the visual feedback at general level without taking in account the relation of visual feedback with structural aspects of an interactive system.

It is important to say than another area of HCI worried to study the visual feedback is the formal methods of interactive system-based on models. Some examples are TADEUS [3], ICO [10] and TRIDENT [15] project. TADEUS prone for take in account the three linguistic visual feedback in a dialogue model and TRIDENT takes in account the ergonomic aspects for visual feedback. The ICO formalism [10] allows specify explicit the behavior of an interactive system in term of oriented object approach and petri nets.

The author has proposed on previous work [9] a simple categorization of visual feedback in order to integrate the rendering specification into the ICO formalism. The current paper expands this categorization in order to unifying a framework of visual feedback according to linguistic levels of dialogue between the user and the interactive systems.

# 8 Conclusion

The main purpose of this paper was to present a taxonomy of visual feedback with the purpose to identify different abstraction levels of output information according to the linguistic nature of dialog between the user and an interactive system. Every categories of taxonomy has been described on detail with simple but meaningful examples. As a summary the taxonomy of visual feedback purpose here help to designer of an interactive system:

- To identify the output information managed by the current interactive system.
- To find the relation between different levels of visual feedback used and architectural component of an interactive system.
- To have a high level description of visual responses of an interactive system independent of any graphical environment.

One of expectations of the present work is to integrate ergonomic factors to visual feedback with the purpose to evaluate the visual respones of an interactive system in function of user task. Another expectations is to extend the taxonomy to dynamic output forms such sound, animation and haptic feedback. This last objective is to integrate and specify divers type of dynamic feedback on highly interactive systems such as multimedia and computer supported learning systems.

*References:*

[1] Campos, J. C. and Harrison, M., "Formally verifying interactive systems*: A review*," *4th Eurographics workshop on Design, Specification and Verification of Interactive System (DSV-IS'97)*, Granada, Spain, pp. 109-124, 1997.

[2] Dix, A. J., Finlay, J., Abowd, G. L., and Beale, R. *Human computer interaction*, Prentice Hall, 1998.

[3] Elwert, T., Continuous and explicit dialogue modelling *Conference on Human Factors in Computing Systems (CHI'96)*, vol. 1996.

[4] Fekete, J. D., "Les trois services du noyau sémantique indispensables à l'IHM," *8emes journées sur l'ingénierie des l'Interaction Homme-Machine (IHM'96)*, Grenoble, France, pp. 45-50, 1996.

[5] Foley, J., vanDam, A., Feiner, S. K., and Hughes, J. F. *Computer Graphics: Principles and Practice*, Addsinon Wesley , 1990.

[6] Green, M., "report on dialogue specifications tools ," *User interface management systems*, New York, pp. 9-20, 1985.

[7] Hudson, S. E., "Adaptive semantic snapping - a technique for semantic feedback at the lexical level," *CHI'90 Human Factors in Computing Systems*, pp. 65-70, Apr. 1990.

[8] Hutchins, E. L. , Hollan, J. D., and Norman, D. A. Direct manipulation interfaces. In: *User Centred System Design: New Perspectives on Human-Computer Interaction*, eds. Norman, D. A. and Draper, S. Hillsadale, New York: Lawrence Erlbaum Associates, 1986.pp. 87-124.

[9] Muñoz Arteaga, J., Intégration de specifications du rétroactions dans un formalisme pour la conception de sistèmes interactifs 2000. PHD thesis of Université Toulouse I.

[10] Navarre, D., Une technique de description formelle et un environnement pour une modélisation et une exploitation synergiques des tâches et du système 2001. PHD thesis of Université Toulouse I.

[11] Palanque, P., Bastide, R., and Dourte, L., "Contextual help for free with formal dialogue design," *5th international conference on human-computer interaction, HCI International'93*, Orlando Florida, pp. 615-620, 1993.

[12] Preece, J., Rogers, Y., Sharp , H., Benyon, D., Holland, S., and Carey, T. *Human-Computer Interaction*, Cambridge University Press : Addison Wesley, 1994.

[13] Shneiderman, B., "The eyes have it: A task by data type taxonomy for information visualizations," *HCIL Dept. of technical report* , University of Marylan , CS-TR-3665, 1996.

[14] Shneiderman, B. Designing the User Interface, *Great Britain: Prentice Hall*, 1998.

[15] Vanderdonckt, J., Conception assistée de la présentation d'une interface homme-machine ergonomique pour une application de gestion hautement interacive. 1997. PHD thesis of Facultés Universitaires Notre-Dame de la Paix.