Model-Based, Multimodal Interaction in Document Browsing

Parisa Eslambolchilar[1],Roderick Murray-Smith[1,2]

[1] Hamilton Institute, National University of Ireland, Maynooth, Co.Kildare, Ireland
parisa.eslambolchilar@nuim.ie
[2] Department of Computing Science, Glasgow University, Glasgow, Scotland
rod@dcs.gla.ac.uk

Abstract. In this paper we introduce a dynamic system approach to the design of multimodal interactive systems. We use an example where we support human behavior in browsing a document, by adapting the dynamics of navigation and the visual feedback (using a focus-in-context (F+C) method) to support the current inferred task. We also demonstrate non-speech audio feedback, based on a language model. We argue that to design interaction we need models of key aspects of the process, here for example, we need models for the dynamic system, language model and sonification. We show how the user's intention is coupled to the visualization technique via the dynamic model, and how the focus-in-context method couples details in context to audio samples via the language identification system. We present probabilistic audio feedback as an example of a multimodal approach to sensing different languages in a multilingual text. This general approach is well suited to mobile and wearable applications, and shared displays.

1 Introduction

In [1], McCullough writes about the need to simultaneously engage both a human's brain and hands, that media have to be dense enough to give the impression of a universe of possibilities. In this paper we present a continuous interaction, dynamic simulation approach which leads naturally to the sort of organic, rich interaction desired by McCullough. It also provides the potential for a solid, systematic way to develop future multimodal interaction systems.
We use tools to control, interact and operate on the physical objects rather than using our bare hands [2]. Instrumental Interaction [3] is an interaction model that operationalizes the computer-as-tool paradigm and extends human powers: a piece of technology, or applied intelligence for overcoming the limitations of the body and controlling information flow [1].
Continuous control is at the very heart of tool usage in the interaction between the human and computer as a tool [1]. It differs from discrete interaction in that it occurs over a period of time, in which there is an ongoing relevant exchange of information between user and system at a relatively high rate, somewhat akin to vision/audio/haptic interfaces which we may not model appropriately as a

series of discrete events [4]. It is also closely related to the development of dynamic systems since in these systems we can control what we perceive and we are dependent on the display of feedback (either visual, audio or haptic) to help us pursue our potentially constantly changing goals. Furthermore, feedback may influence an uncertain user's actions as more information becomes available [5]. In order to address the behavioral issues early in the design stage, formal modeling techniques for real-time systems supported by powerful analysis tools could be considered and for calibration and refinement issues, a more general framework that can guide the modeling approach is needed.
In this paper, as an illustration of how this approach can support multimodal interaction, we use the example of browsing and sensing multilingual texts. Here the focus-in-context method and the adaptive dynamics are coupled with sonification, based on a probabilistic language model, which can be linked to a wide range of inputs and feedback/display mechanisms.

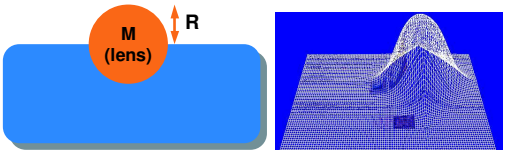
2 Continuous Interaction and Text Browsing

Our interaction model is an example of *continuous interaction* which means the user is in constant and tightly coupled interaction with the computing system over a period of time. Here, we use control theory as a formal framework for analysis and design of continuous interaction, multimodal feedback and overall system dynamics.
Focus-in-context methods are useful for displaying information in context and can be applied to various objects [6–10]. As our integrated system benefits from an Elastic Presentation Framework (EPF) [11], the presentation has an elastic nature. Elastic is a positive word that implies adjusting shape in a resilient manner, which means these materials can always revert to their original shape with ease. One popular way of describing a conceptual model [12] in terms of interaction metaphors [3] is based on an analogy with something in the physical world. Figure 1 is illustrating a conceptual model, a floating elastic ball in the water, for a fisheye lens. So in this analogy, changes in the height of the center of the ball outside the water, $y(t)$, adjusts the degree of magnification (DOM) and is function of time $\dot{y}(t)$. If we show the radius of the ball with R (maximum DOM), then

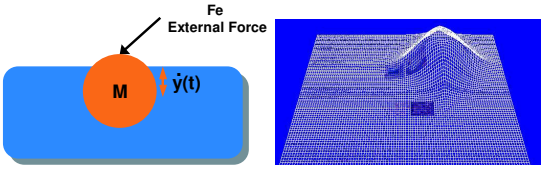
$$DOM(t) = R - \dot{y}(t) \tag{1}$$

When we apply an external force, f_e, we push the ball down in the water (not more than its radius) so the DOM decreases and when we release the force the DOM starts to increase (not more than its radius, see Figure 1). So the DOM is a variable which is continuously controlled by external force (mouse or tilting angles) and speed of movement. From Newton's second law of motion we can write the equation in vertical direction:

$$m\ddot{y}(t) = f_y - k\dot{y}(t) \tag{2}$$



(a) Left-A floating ball in the water. Right- A fully magnified fisheye lens.



(b) Left-The ball after applying external force and pushing it down in the water. Right- The DOM decreases as soon as the user starts to scroll.

Fig. 1: Interpreting Fisheye Lens as a floating ball.

k is the damping factor caused by water resistance, and the effect of gravity and the weight of ball is negligible. In the horizontal direction we can write:

$$\begin{aligned} ma &= f_x - kv \qquad\qquad or \\ a &= \frac{f_x}{m} - \frac{k}{m}v, \end{aligned} \tag{3}$$

where v and a represent velocity and acceleration and k is the damping factor caused by water resistance. We may assume f_x is a function of f_y and velocity (this assumption will couple rates of change in DOM to speed of movement, as well as input) as below:

$$f_y = cf_x - bv \tag{4}$$

Where c and b are coefficients. After substituting f_y in (2) we can rewrite it as below:

$$\ddot{y}(t) = \frac{c}{m}f_x - \frac{b}{m}v - \frac{k}{m}\dot{y}(t) \tag{5}$$

From classical textbooks in control theory [13] we can represent the mathematical model of our physical system as a set of input, output and state variables related by first-order differential equations in a state-space model. If we introduce x as position then velocity and acceleration will be first and second derivatives of the position respectively. The chosen state variables are $x_1(t)$ as position of cursor, $x_2(t)$ as velocity, $x_3(t)$ as rate of change of the DOM and u as f_x. So state

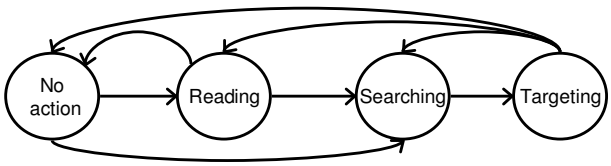


Fig. 2: Four discrete states of control mode in text-browsing example and transitions among them.

variables can be written as below:

$$\dot{x_1}(t) = v = x_2(t) \tag{6}$$

$$\dot{x_2}(t) = a = \dot{v} = \frac{-k}{m}x_2(t) + \frac{u(t)}{m} \tag{7}$$

$$\dot{x_3}(t) = \ddot{y}(t) = \frac{-b}{m}x_2(t) + \frac{-k}{m}x_3(t) + \frac{c}{m}u(t) \tag{8}$$

The standard matrix format of these equations is:

$$\begin{pmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & \frac{-k}{m} & 0 \\ 0 & \frac{-b}{m} & \frac{k}{m} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{m} \\ \frac{c}{m} \end{pmatrix} u \tag{9}$$

This matrix reproduces the standard second-order dynamics of a mass-spring-damper system which we used previously [14]. Also this has many parameters that can be tuned, usually as a series of interacting, but essentially separate equations. Here, a 2 degree of freedom input can control both velocity and magnification factor so it proves a simple dynamic model can be tuned for different interactive models and generate different behaviors in controlling the task (next section). For example, the focus-targeting problem [15] can easily be solved in state-space representation by tuning c in matrix A or the 'hunting effect' problem [15] when the user overshoots the target due to the system increasing the DOM as the user slows, becomes a matter of tuning the dynamics of the system by changing the entries in the A matrix (For more information refer to [14]).

3 User Behavioral Models

In the 60's and 70's William Powers suggested [16, 17] that many kinds of behavior can be described as control systems, and he argued that behavior is not output but, is the *control* of perception. In the model-based text browser example, the user's input, mouse data, controls what s/he perceives via focus-in-context and sonification feedback. In this example we assume the user is acting in one of four different modes: *no-action, reading, searching and targeting.*

Figure 3 illustrates the general framework. Figure 3(b) shows the classification of the user behaviour being used to switch the control mode. This mode is then coupled to the visualization parameters, as shown in Figure 3(c), where

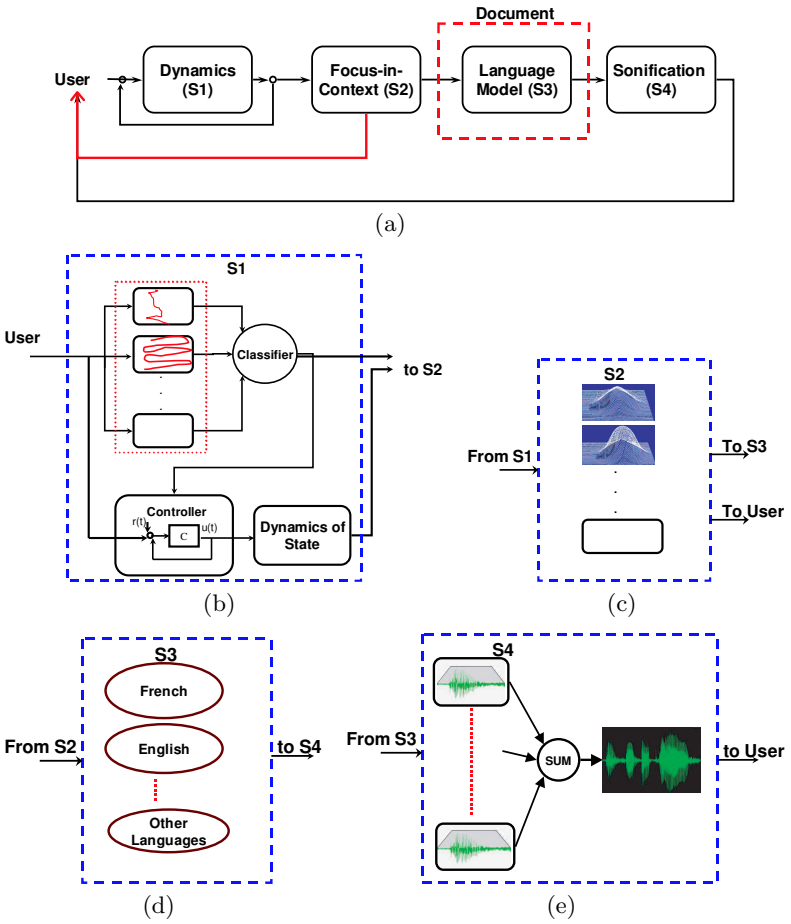


Fig. 3: (a) A general probabilistic framework of the model-based behavior system. (b) A Bayesian classifier classifies the user's input. Its output and the user's input come to the controller and change the dynamics (state variables). (c) State variables coupled to the focus-in-context change the size and shape of lens. (d),(e) The language identification method infers the most probable language inside the window around the lens and its output probabilities are fed to the audio synthesis algorithm.

the control mode changes the size and shape of the lens, and the controller provides the DOM, position and speed of the lens. For example, in reading mode the controller adjusts the DOM to stay in the maximum level, but as a long horizontal lens, while if the user 'breaks out' into general searching, the DOM is decreased smoothly to a lower level. This prevents the targeting problem [15] in focus-in-context techniques.

3.1 Detecting state transitions

Figure 2 illustrates the possible state transitions. Initially, the user is in the no-action state. Depending on the input behaviour, the user can either go to the reading or the searching mode. A qualitative description of the automatic mode transitions is given below: In the reading mode, the user is making continuous increasing changes in x direction (left-to-right) and small changes in y direction (not more than height of a line) and at the end of the line makes a sudden change from right-to-left in x direction. If the changes in y direction are more than height of the line the system switches the mode to the searching mode.
After finding the target, the user slows down or stops scrolling until the lens is over the target point (targeting mode), or can return to the no-action mode directly.
A general technique for implementing this is to use a probabilistic classification of the likelihood of being in one of these four browsing behaviors according the joint probability of the input and output time-series. From Bayes' law, we write this as below:

$$P(Mode \mid X) = \frac{P(Mode)P(X \mid Mode)}{P(X)} \tag{10}$$

where X is an appropriate window of previous inputs and possibly also outputs. $P(X \mid Mode)$ can be identified from experimental data collected from test users using standard density estimation models.

3.2 Changing meaning of inputs

Given the inferred user task, the controller behaviour should be designed to support the user by enabling them to complete the task with as little effort as possible. This can include changing the interpretation of the inputs to being reference values, rather than direct control actions. Taking our inspiration from modern aircraft controllers, which have different interpretations of aircraft controls depending on flight mode (e.g. take off, altitude-hold, attitude-hold etc.), and which blend seamlessly between modes. See [18] for examples. For example, if the classifier infers that the user is in reading mode, then the controller automatically scrolls the lens from left to right and moves to the next line smoothly, rather than the user having to do this. Any left–right movement of the mouse now controls the reference reading speed that the reading mode controller is trying to achieve. Similarly other modes can reinterpret control inputs as browsing speed, or as position acquisition control while zooming in to a point of interest after browsing. This means that as the user performs the various tasks they switch between control modes automatically, and their inputs have different meanings, but that the transitions are always smooth and natural, and the user is often not even aware that their movements are having a different effect in the different modes.

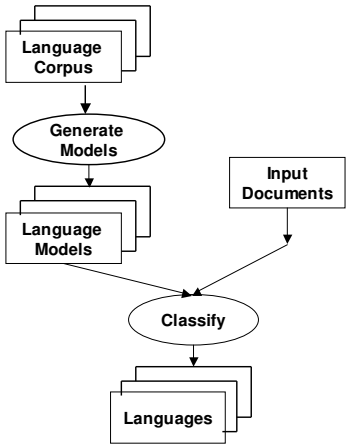


Fig. 4: The major stages of language identification system.(Top-left) The distinctive features for each language in a multilingual corpus are determined and stored in a language model tree. (Top-right) The word the user is pointing to in an untrained text is compared to the language models during the classification stage. The language model, which is the most similar to this word is then selected.

4 Language Identification System

Language classification consists of two major stages. From Figure 4 we see at the top we have the modeling stage. During this stage, the language-specific features of a text are learned and stored in a model. First, as can be seen on the upper left-hand side in this figure, the distinctive features for each language in a multilingual corpus are determined and stored in a language model. Later, seen on the upper right-hand side, the features of a specific text are determined and stored in a document model. In this application a language model based on partial predictive matching [19] is used to calculate the probability of letter, l, through a conditional probability distribution $P(letter \mid prefix)$, which specifies the view about future possible value of l, conditional upon the truth of that particular description *prefix* on a per-word basis. Then a tree with probability information is generated from a corpus [20]. In our application these trees are built from short texts collected from *BBC* and *Le Monde* news web-sites in English and French (only few paragraphs). For simplicity no grammar or word-level model is used, although this would be likely to improve performance significantly [21]. At the bottom of the Figure 4, the classification stage is shown. During this stage, a word (the user is pointing to) of an untrained text in a document is compared to these trained language models. The language model which is the most similar to the language of this word is then selected, and represents the language of the word the user has pointed to. The actual comparison method depends on the classification technique used.

Language Prediction

Prediction in this application is done using Bayes' Law to infer the most probable language given text from a document.

$$P(Language \mid Word) = \frac{P(Language)P(Word \mid Language)}{P(Word)} \tag{11}$$

The document we have considered in this applications contains sentences and paragraphs both in English and French. When the user is scrolling over the text the application provides a virtual window (with the size of the lens' width, which is dynamic and adapts with any change to the DOM) around the cursor (Figure 3). Then the probabilistic language models calculate the probabilities of all words in the window in each language. For example, for only two words, $w1$ and $w2$ in the window, we have:

$$P(Language \mid w1, w2) = P(w1, w2 \mid Language).P(Language)/P(w1, w2) \tag{12}$$

As we have made the simplifying assumption that words in the window are independent, we can write the generalized form of equation (12) as below:

$$P(Language \mid Window) = \left[\prod_{i=1}^{i=n} \frac{P(w_i \mid Language)}{P(w_i)}\right] P(Language)$$
$$n \; is \; window \; size, \quad \forall \; i = 1 \; to \; n \quad w_i \in Window \tag{13}$$

So, we infer the language from a number of words from a document contained by the fisheye lens. The most probable language for any part of the text can be estimated as accurately as desired by making the window (or Drop-Off function's width in the fisheye lens [22]) sufficiently small.

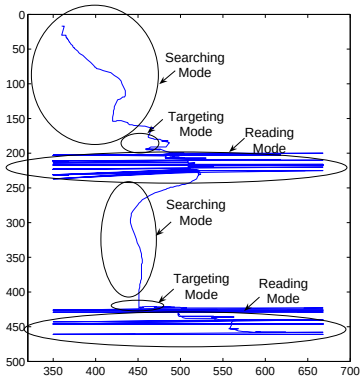
5 Language Model and Granular Synthesis Feedback

As an intuitive model of the sonification process, we can imagine the words in the text to be embossed on the surface. Similar to [23] we simulate this model in our implementation by drawing an audio sample and placing that in an audio buffer, as each word belongs to a certain class of language "hits" the lens. This technique is a form of granular synthesis; [24] gives other examples of granular synthesis in interaction contexts. A real world analogy would be the perception of continuous levels of radiation via frequency of discrete pulses from a Geiger counter; here the continuous variable is the word flow rate in a specific language. At a higher rate-of-scroll the acoustic response of the system, e.g. sampling frequency and volume of the audio sample decreases and provides the sense of distance to the text. At lower rates-of-scroll the sampling frequency and volume of the audio increases and the user feels he is getting closer to the text. Also, the volume and audio frequency are inversely related to the rate of scroll, so the audio texture as we pass over the text gives both an impression of the language of the text, as well as the speed at which we are passing it.

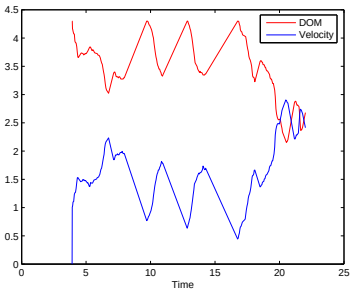
Similar to [24], the sonification technique can be extended to language recognition. We can sonify a probabilistic language recognizer by associating each language model with a source waveform, and each model's output probability then directly maps to the probability of drawing a grain from the source corresponding to that model (Figures 3(d) and 3(e)). The temporal distribution of grains inside the source waveforms maps to the probability of the language of the words inside the virtual window. The overall grain density is dynamic throughout the sonification when the user scrolls over the text. In practice, during the searching mode this produces a sound that's unclear when text features are blurred and the DOM is in the minimum level, and it means the information entropy inside the virtual window around the cursor is high. This features resolve to a clear, distinct sound as system's mode switches to the targeting. The sonification's primary effect is to display the current goal distribution's entropy, i.e. language, audio and text content.
The concept of entropy in information theory describes the level of uncertainty of a random variable. An alternative way to look at this is to talk about how much information is carried by the signal. For example, in an English text, encoded as a string of letters, spaces, and punctuation the signal is a string of characters. The letter frequency for different characters is different, and we cannot perfectly predict what the next character will be in the string: it is, to some degree, 'random'. Entropy is a measure of this randomness, suggested by Shannon [25].
So model-based behavior in this task couples the user's input (speed of scroll) to the visualization technique via the dynamics and the focus-in-context method couples detail-in-context to audio samples via the language identification system (Figure 3(a)).

6 Example Use of Working System

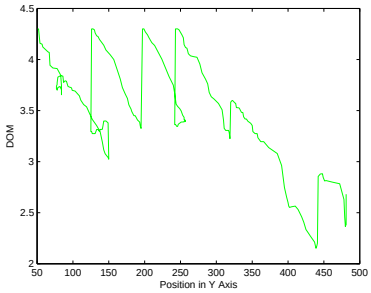
We developed a document viewer using the EPF library [22] for the focus-in-context method to browse a PDF, PS or DOC file which have been converted to an image (Bitmap) file. The document we presented was a 5 pages scientific document in English and a paragraph, a figure caption and few sentences written in French. The interaction is controlled via a mouse. The results in Figure 5(a) highlight the different navigation styles of the different interfaces and input methods. In the focus-in-context implementations the user had smooth navigation, which also included smooth changes in the DOM (See Figure 5(b)). If the velocity rises above a threshold DOM smoothly decreases and the reading mode switches automatically to searching mode, for instance in Figure 5(b) this has happened around t= 7, 11 and 14 seconds. So the velocity of the input device provides a smooth switch between different modes of control. Figure 5(c) presents how the DOM changes when the user has found the French sections in the document, stopped for a brief check and clicked over the text. The French sections are around pixels: 150, 190, 270, 330, and 420 and we see the user has found the most of sonically highlighted sections.



(a) The user's trace in the text browser example. The user starts the scrolling from top-left corner (beginning of the document) and scrolls down. The searching behavior becomes targeting and then reading behavior.



(b) Change in the DOM and velocity versus time.



(c) Change in the DOM versus position in finding French sections around pixels 120 ,190, 270, 330, and 420.

Fig. 5: Plot of logged data in searching French sections by one of users. Note that the presented document in (b) and (c) is different from the document in (a).

7 Conclusions and Future Work

In this paper we presented a novel approach to designing interaction between the user and the system. This approach is a model-based interactive method for browsing a multilingual text based on a language model, focus-in-context method and continuous interaction interface. We presented a floating ball model as an example of how the dynamic approach can be used creatively to design interaction, and suggest new metaphors. The state-space, dynamic system representation coupled the user's intention to the visualization technique via only the

two degree of freedom mouse input, allowing the user to switch smoothly among reading, targeting and searching modes by only moving the mouse. A probabilistic language model was used for online classification of the focus content in a multilingual input document.
Our probabilistic audio feedback based on granular synthesis is an example of a multimodal approach to sense different languages in the document. The focus-in-context method representing this document coupled details in context to audio samples via the language identification system. So the system could provide both visual and audio feedback to the user.
A motivating factor behind the approach in this paper is that we can in the long-term, potentially develop the dynamic systems simulation approach as a systematic approach to creating designs which can shape interaction and provide rich multimodal feedback, in the same way that has been successful in other areas of computing, where physics and model-based approaches revolutionized the field, such as ray tracing algorithms in computer graphics [26].
More refinement of the prototype system would be required, and a thorough usability study needed to determine the practical applicability of the specific interface described here, but some initial observations are made below. Initial informal evaluation of the implementation of sensing multilingual texts on a laptop instrumented with a mouse and headphone were positive, and users felt that this provided an intuitive solution to the problem of finding information in a particular language in a multilingual text without reading the text. Sonifying each language in the document gave users a sense of their motion through the document, which allowed them to continue their interaction while being involved in other tasks. The system allowed users to browse the document and locate targets (here the idea was searching and locating French written parts of the document) without looking at the screen. Supporting *intermittent interaction*, where a user can spend varying amounts of attention on interaction while carrying on with other activities, is very important for usable interaction, while on the move, making this approach interesting for use in mobile phones and small screen devices.

Acknowledgements

The authors gratefully acknowledge the support of IRCSET BRG SC/2003/271 *Continuous Gestural Interaction with Mobile devices*, HEA project *Body Space*, and SFI grant 00/PI.1/C067, the IST Programme of the European Commission, under PASCAL Network of Excellence, IST 2002-506778. This publication only reflects the views of the authors.

References

1. McCullough, M.: Abstract Craft: Practical Digital Hand. The MIT Press (1998)
2. Kelley, C.R.: Manual and Automatic Control. John Wiley and Sons, Inc., New York (1968)
3. Beaudouin-Lafon, M.: Designing Interaction, not Interfaces. In: AVI '04: Proceedings of the working conference on Advanced visual interfaces. (2004) 15–22
4. Doherty, G., Massink, M.: Continuous Interaction and Human Control. In Alty, J., ed.: Proceedings of the XVIII European Annual Conference on Human Decision Making and Manual Control. (1999) 80–96

5. Faconti, G., Massink, M.: Continuous interaction with computers: Issues and Requirements. In C.Stephanidis, ed.: Proceedings of Universal Access in HCI. Volume 3., Lawrence Erlbaum Associates (2001)
6. Bederson, B.B.: Fisheye Menus. In: UIST '00: Proceedings of the 13th annual ACM symposium on User interface software and technology. (2000) 217–225
7. Furnas, G.: Generalized Fisheye Views. In: Proceedings of CHI'86. (1986) 16–23
8. Lamping, J., Rao, R., Pirolli, P.: A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In: Proceedings of CHI 95. (1995) 401 – 408
9. Mackinlay, J.D., Robertson, G.G., Card, C.K.: The Perspective Wall: Detail and Context Smoothly Integrated. In: Proceedings of CHI'91. (1991) 173–179
10. Sarkar, M., Brown, M.H.: Graphical fisheye views of graphs. In Bauersfeld, P., Bennett, J., Lynch, G., eds.: Human Factors in Computing Systems, CHI'92 Conference Proceedings: Striking A Balance, ACM Press (1992) 83–91
11. Carpendale, M.S.T.: A Framework for Elastic Presentation Space. PhD thesis, Department of Computing Science, Simon Fraser University, Canada (1999)
12. Preece, J., Rogers, Y., Sharp, H.: Interaction Design: Beyond Human Computer Interaction. John Willey (2002)
13. Sheridan, T.B., Ferrell, W.R.: Man-Machine Systems: Information, Control, and Decision Models of Human Performance. MIT press (1974)
14. Eslambolchilar, P., R.Murray-Smith: Tilt-based Automatic Zooming and Scaling in mobile devices-a state-space implementation. In: Mobile Human-Computer Interaction MobileHCI 2004: 6th International Symposium. (2004) 120–131
15. Gutwin, C.: Improving focus targeting in interactive fisheye views. In: Proceeding of CHI'02. (2002) 267–274
16. Powers, W.T.: Living Control Systems: Selected papers of William T. Powers. The Control Systems Group Book (1989)
17. Powers, W.T.: Living Control Systems II: Selected papers of William T. Powers. The Control Systems Group Book (1992)
18. Tischler, M.B.: Advances in Aircraft flight Control. Taylor & Francis (1994)
19. Bell, T., Cleary, J., Witten, I.: Text Compression. Prentice Hall Advanced Reference Series. Prentice Hall (1990)
20. Williamson, J., Murray-Smith, R.: Dynamics and probabilistic text entry. In Murray-Smith, R., Shorten, R., eds.: Hamilton Summer School on Switching and Learning in Feedback systems. Volume 3355 of Lecture Notes in Computing Science., Springer-Verlag (2005) 333–342
21. Lesher, G., Rinkus, G.: Leveraging word prediction to improve character prediction in a scanning configuration. In: Proceedings of the RESNA 2002, Annual Conference. (2002)
22. Carpendale, S., Montagnese, C.: A framework for unifying presentation space. In: Proceedings of UIST'01. (2001) 82–92
23. Eslambochilar, P., Williamson, J., Murray-Smith, R.: Multimodal feedback for tilt controlled speed dependent automatic zooming. In: UIST'04: Proceedings of the 17th annual ACM symposium on User interface software and technology, (ACM)
24. Williamson, J., Murray-Smith, R.: Sonification of probabilistic feedback through granular synthesis. In: IEEE Multimedia. Volume 12, Issue 2. (2005) 45–52
25. Shannon, D.: A mathematical theory of communication, Bell Labs (1948) http://cm.bell-labs.com/cm/ms/what/shannonday/paper.html.
26. Foley, J., Dam, A.V., Feiner, S., Hughes, J.F.: Computer Graphics, reissued 2nd Ed. Addison Wesley, ISBN: 0201848406 (1995)