Hamilton Institute

University *of* Glasgow

Making Sense of Fisheye Views

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

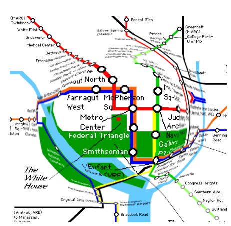
[1]Hamilton Institute, NUI, Maynooth,Ireland
[2]Department of Computing Science, University of Glasgow

Contents

- Background
- Advantages and disadvantages of Fisheye Views
- Human behavior and control theory
- New view to fisheye views
- Modeling fisheye views
- Gestures in information visualization & fisheye views
- Retrieving information, language model and fisheye views

Introduction

- The CPU power and memory is increasing fast. However, the common sizes of our computational display spaces have only minimally increased or in some cases, such as hand held devices, actually decreased.
- The size and scope of the information spaces we wish to explore are also expanding.
- Making effective use of the available display space has long been a fundamental issue in user interface design.
- Representing vast amounts of information on our relatively small screens has become increasingly problematic and has been associated with problems in navigation, interpretation and recognition.
- These problems have lead to a number of strategies in computer interfaces, including for example, fisheye views, the Perspective Wall, multiple-zoom viewers, and infinite interactive zoom viewers.



Background

- 1973: *Furnas cites a PhD thesis by William Farrand as one of the earliest use of fisheye views of information on computer screen.*
- 1982 : *Spence and Apperley presented the Bifocal Display, an early computer based distortion-oriented display technique.*
- 1986: *Furnas presented concepts of the Generalized Fisheye View based on textual trees.*
- 1991: *At CHI, Card, Mackinlay, and Robertson presented two views of structured information that have fisheye properties :*
  - Perspective Walls,
  - Cone Tress.
- 1992: *Sarkar and Brown introduced a number of Fisheye View-like applications and attempted to formalize the mathematical foundation for the Fisheye View.*
- 1996,1997 : *Keahey and Robertson presented non-linear magnification.*
- 2001, *Carpendale presented an elastic geometric framework that unifies distortion and non-distortion presentation paradigms.*

Advantages of Focus in Context methods

- These methods are said to support human potential for visual gestalt, to reduce the cognitive effort needed for the reintegration of information across separate views, and to address navigational problems by accessing spatial reasoning.
- Also, studies indicate that setting detail in its context is common practice in human memory patterns and that there is increased user performance in path finding tasks.
- To allow presentation and navigation of an information space, there are generally three basic interaction methods to effect a change of viewpoint using an input device: scrolling, pointing and selecting.
- Results have shown that in situations where magnification is required, distortion-oriented views can be fast and effective representations for interactive tasks.

Why Focus in Context methods have not been widely accepted?

- A general discomfort with the use of distortion and/or to the perception that the use of distortion and non-distortion based presentation methods are mutually exclusive.
- The relative merit of different presentation methods will be dependent on:
  - the type of task, the nature of the information, and the preferences, skills of the person using it and tasks.
- Focus targeting is difficult in interactrive fisheye views and makes the user to feel disoriented in the information space.

Is MODELING the future of HCI?

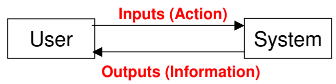
- The literature of HCI is full of organizational diagrams.
- In most cases these diagrams represent states about how some behaviours are caused by some internal processes and extrenal interactions.
- But in most cases there is no test of the model inherent in these diagrams.
- These diagrams are descirptions how their designers think the real systems are organized but they are not real working models.
- Once there is a working model it is hoped it does behave very closely to some specified behaviour of a real organization.
- Modeling is an iterative procedure.

Purposeful Behaviour

- In 1973 William Powers described a new approach to psychology based on the fact that behavior is control.
- The term behavior refers to a wide range of activities that any organism does but one obvious characteristic of most behavior is it repeats.
- Behavior is control because the events called behavior are consistent results of continuously changing effects produced, simultaneously, by the organism and the environment.
- Watching behavior without knowing that control exits, creates two illusions:
  - Behavior is output,
  - Stimulus control.

How & When do apply control?

- From observing the behavior in everyday life or experimental research labs identify a variable that might be under control.
- Apply disturbances to the variable. If disturbance produces less than the expected effect then the variable is likely to be under control.
- Usually human is in closed-loop interaction with the system. For instance, in tracking tasks the user observes information as well as monitors display to generate control actions to minimize the difference between actual & desired outputs. This type of task is called manual control.



- In modern interaction techniques such as gesture recognition, haptic feedback,and so on the user is in continuous, constant, tightly coupled interaction with the computing system.

Input devices in rate control

- Tilt input devices are important in small screen devices, for example:
  - ❖ XSENS P3C:
    - 3D accelerometer
    - Stream interface driver and control Tray icon and control panel applet on Pocket PCs
    - Single-handed tilt-control

  - ❖ MESH:
    - 3D Accelerometers,
    - Gyroscopes
    - Magnetometers
    - Vibertactile device

- Stylus and touch screen

Fisheye Views on Mobile devices

Using accelerometer provides a direct intutive mapping from acceleration in the real world to the acceleration in the interface, which also suggests a range of other affordances, important in multimodal feedback and we can choose to explicitly use these features to design the system to encourage interaction to fall into a comfortable, natural rhythm.

State space modeling

The generic form of state equations is

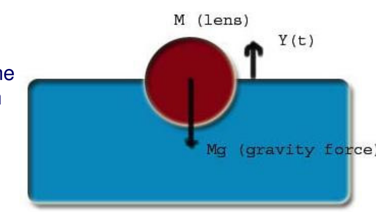
$$\dot{X} = f(x) + g(u) \qquad (1)$$
$$\dot{Y} = h(x)$$

where f(*x*), g(*u*) and h(*x*) can be non-linear functions *X(t)* is a state vector, *U(t)* is an input vector and *Y(t)* is an output vector.

New View to Fisheye Views

From Archimedes law:

Water displacement = Mass_of_water* Gravity;



The height of the centre of the ball above the water line which represents the magnification factor is a function of time, y(t).

From Newton's second law of motion we have :

Force = Mass * Acceleration

Cause of Change = Resistance to Change * Rate of Change

If we show radius of the ball with R , the height of displaced water will be

H = R-y(t);

Then:

M*ydotdot = M*g – D*g*PI*H*(R*H-H*H/3); (2)

PI*H*(R*H-H*H/3) is the volume of a segment of a ball of height H and D is the density of the water (usually equals to 1).

When we have an external force (Fe):
- In horizontal direction :

M*Acceleration_x = Fe_x – R*V -----> a = Fe_x/M – K*V/M

- In vertical direction :

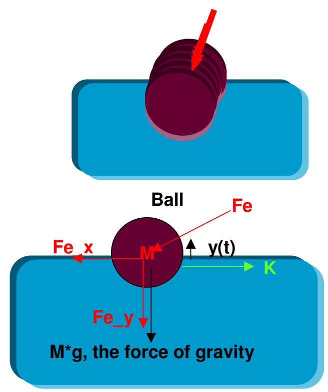
M*Acceleration_y= Fe_y + M*g – D*PI*H*g*(R*H-H*H/3)

To have a stable system we need a constrain:

0<fe_y + M*g<D*4*PI*R*R*R/3

If X is position then Velocity is the **first derivative** of the **position** and Acceleration is the **second derivative** of the **position.** So we can choose x1(t) for position, x2(t) for speed of scroll (V) and x3(t) for rate of change of DOM (Z) as state variables :

$$\dot{x}_1(t) = Velocity = x_2(t)$$
$$\dot{x}_2(t) = Acceleration = \frac{-K}{M}x_2(t) + \frac{Fy}{M}$$
$$\dot{x}_3(t) = \dot{Z} = \frac{b}{M}x_2(t) + \frac{-K}{M}x_3(t) + \frac{a}{M}Fy$$



K, Water Resistance (damping effects),

M, Mass   ydotdot= Acceleration_y

V, Velocity   g,gravity

Ax : Acceleration_x

R: Radius

The more specific case of a linear system

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (5)$$
$$y = Cx(t) + Du(t)$$

where *A* is a square matrix called the *system matrix, B* is *input matrix, C output matrix and D* is a matrix which represents any direct connection between the input and output.

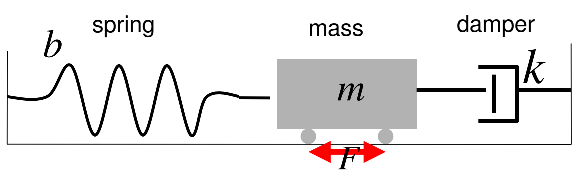
The standard matrix format of equations in (5) for lens is:

$$\dot{x}_1(t) = V = x_2(t)$$
$$\dot{x}_2(t) = \dot{V} = \frac{-K}{M}x_2(t) + \frac{Fy}{M}$$
$$\dot{x}_3(t) = \dot{Z} = \frac{b}{M}x_2(t) + \frac{-K}{M}x_3(t) + \frac{a}{M}Fy$$

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix} = \begin{bmatrix}0 & 1 & 0\\ 0 & \frac{-K}{M} & 0\\ 0 & \frac{b}{M} & \frac{-K}{M}\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{M}\\ \frac{a}{M}\end{bmatrix}u(t)$$

In all experiments *K=10, M=1, a=3 and b=0.5*

Spring-Mass-Damper System



k : damping coefficient
b : spring constant

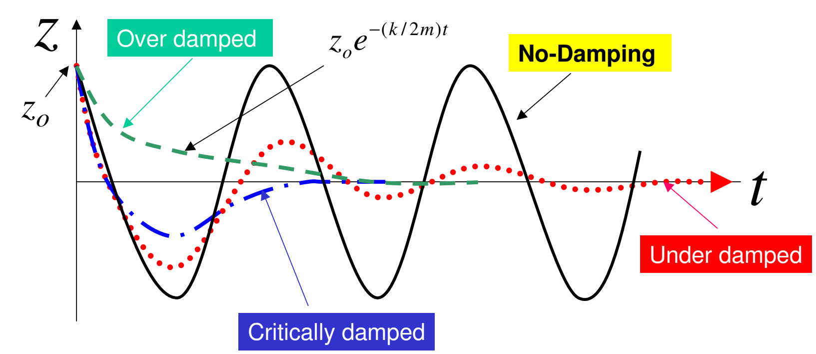
Equation of forced motion ----- >
$$m\frac{d^2z}{dt^2} + k\frac{dz}{dt} + bz = F$$

So we can introduce x1(t) for position (z), x2(t) for velocity:

$$\dot{x}_1(t) = Velocity = x_2(t) = \dot{z}(t)$$
$$\dot{x}_2(t) = Acceleration = \ddot{z}(t) = \frac{F}{M} - \frac{K}{M}x_2(t) - \frac{b}{M}x_1(t)$$

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}0 & 1\\ \frac{-b}{M} & \frac{-k}{M}\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{M}\end{bmatrix}u(t)$$

Spring-Mass-Damper System



- **Under Damped : small damping, several oscillations**
- **Critically Damped : important level of damping, overshoot once**
- **Over damped : large damping, no oscillation**

Calibrating fisheye views and state space approach

The state-space formulation allows multiple variables, and derivative effects (e.g. position, velocity) can be coupled with DOM, without any further coding, by just changing the entries of the *A* matrix,

To enhance the smoothness of the transition between low DOM and high DOM,

- In state space representation ‘a’ is a function of velocity. When speed is above the dead-zone threshold (here set to 0.1), $a = 3$ but below this threshold $a=0$.

Rapid change in DOM problem when making a rapid change of direction

- When the user changes the direction of the movement which means the sign of velocity and input are different, we set $a=a*0.3$,
- When sign of velocity is minus we keep *‘ a ‘* always positive,

Hunting effect,

- One approach to this would be to switch to a ‘diving’ control mode if $dz/dt < z_{thresh}$, where $a=0$, preventing DOM increases, unless a major change in velocity, occurs, which would switch the state back to velocity control.

Oscillatory behaviour

- Large damping, no oscillation
- Large spring constant , big oscillation,

Fisheye View, Tilt Input & Control Modes

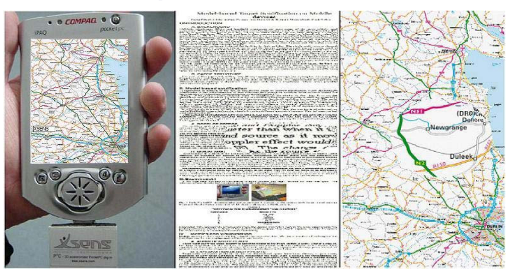
The state of tilt inputs changes the control modes automatically in the system:

- No tilt input from the user means s/he tends to read the text. So the ball without any effort from the user floats on the surface automatically with highest degree of magnification (DOM) from left-to-right (in a text written in any western language) with a minimum speed.
- By tilting the device a sudden increase in speed of the lens causes a dramatic decrement in DOM and reduces that to a minimum level and the system's mode switches to the searching mode.
- The position control mode helps the user to move the lens to the target without tilting the device, just by tapping on the touch screen over the target. So the lens moves to the target automatically. We can switch from tilt-angle to stylus and indicate the desired position by introducing a reference value ‘r’ and creating a control law $u=L*(r-x)$ and change the state equations to :

$$\dot{x} = Ax + Bu = Ax - BLx + BLr$$
$$= (A - BL)x + BLr$$

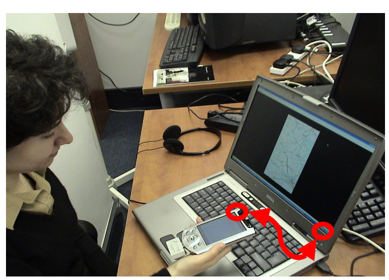
- By tilting the device the position control mode switches to the velocity control mode.

Examples on PDA



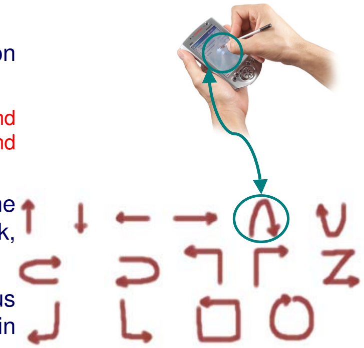
Pocket PC and XSENS P3C-3D accelerometer attached to the serial port, (Middle) Fisheye lens in reading mode over a text document, (Right) Fisheye lens over a map in targeting mode.

A Pocket PC and laptop have been paired via Bluetooth connection. Pocket PC sends tilt input sensor data to laptop and laptop controls the rate of scroll and degree of magnification (DOM) of the lens over the map.



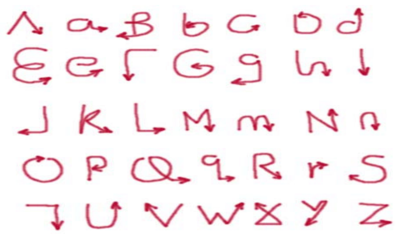
Gestures & Fisheye Views

- The relative merit of different presentation methods will be dependent on:
  - the type of task, the nature of the information, and the preferences, skills of the person using it and tasks.
- Gestures will help the user to change the presentation of the views based on the task, context, nature of the information and so on.
- A two-layer NN was used to recognize 15 stylus gestures. For instance the speed of the lens in automatic reading mode is changed with left-to-right or right-to-left gestures, DOM is increased or decreased with clockwise and anti-clockwise gestures respectively, and so on.



Letter Recognition on PDA

- Another two layer perceptron NN with hyperbolic tangent activation function was used to recognize 27 characters (26 English letters and backspace) .
- After recording the stylus path it is smoothed to few base point.
- These points are transformed to a vector of angles (0-360 degree) and their sine and cosine values are calculated.
- At last the transformation of a stylus path into a vector of cosines and sinuses is passed to the network's input.



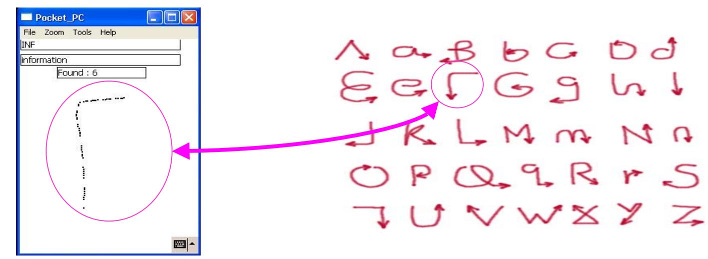
Making Sense of Fisheye Views

- Stylus path is a set of points (x,y values) in a 2D small screen (240*320). We can write cosine and sine of apoint *i* as below:

$$\text{cosines}_i = \frac{p_i.y}{\sqrt{p_i.x^2 + p_i.y^2}}$$

$$\text{sinuses}_i = \sqrt{(1 - \text{cosines}_i^{\,2})}$$

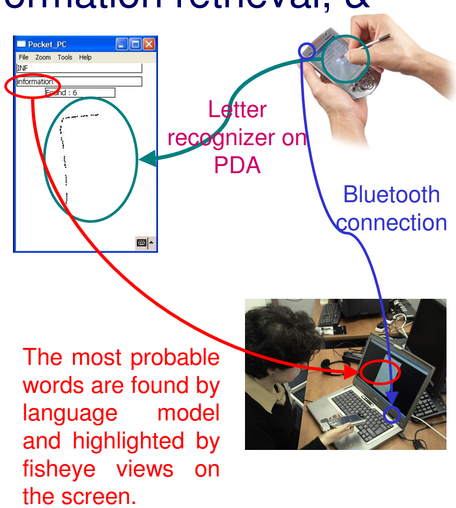
- We get 25 sample points from the stylus path and calculate cosine and sine for each of them and pass them as a vector to the input layer of NN. It means number of neurons should be 50, 25 for cosines and 25 for sinuses.



Making Sense of Fisheye Views

Language model, Information retrieval, & Fisheye Views

- A simple probabilistic language model is used to infer user's intention and produce probability of the letter on a per-word basis. A tree with probability information is generated from a corpus.
- In each letter writing the language model finds the most probable word(s) in the document close to the set of letters have already been written and fisheye view highlights them in the document. This provides a strong link between the probabilities and the display.



Making Sense of Fisheye Views

Conclusions

- In Fisheye Views the degree of magnification is the controlled variable,
- Presenting a state-space, dynamic systems representation of the dynamic coupling involved in Fisheye Views,
- Controlling reading, searching and browsing behaviours automatically,
- Demonstrating the applicability of the approach by implementing a fisheye view interface for a document and a map browsing system on a PDA instrumented with an accelerometer,MESH stylus control,and Bluetooth connection,
- Altering lens' parameters via gestures,
- Retrieving information via language model and Fisheye Views in a document browser.