

Computer Display Control and Interaction Using Eye-Gaze

M. Farid (1), F. Murtagh (1) and J.L. Starck (2)

(1) School of Computer Science, Queen's University Belfast, Belfast BT7 1NN, Northern Ireland, UK.

Email: f.murtagh@qub.ac.uk Tel: +44 28 9027-4620. Fax: +44 28 9068-3890.

(2) DAPNIA/SEI-SAP, CEA-Saclay, 91191 Gif-sur-Yvette Cedex, France.

Abstract

Innovative systems for user-computer interaction based on the user's eye-gaze behavior have important implications for various applications. Examples include user navigation in large images, typical of astronomy or medicine, and user selection and viewing of multiple video streams. Typically a web environment is used for these applications. System latency must be negligible, while system obtrusiveness must be small. This paper describes the implementation and initial experimentation on such an innovative system.

Keywords – Eye-gaze tracking, image compression and decompression, streaming video, visual interactive user interface.

Introduction

Vision-based user-computer interfaces include both eye-gaze pointing (Quek, 1995) and gesture recognition. Istance and Howarth (1994) have reviewed user interface interaction based on eye-gaze control. Eye-gaze, when tracked and made use of for control of image displays, may suffer from computational requirements leading to latency problems, and obtrusiveness of the camera and positioning apparatus. In the work described in this paper, we completely overcome problems related to latency, as well as achieve a relatively successful solution relating to obtrusiveness, based on the eye tracking environment used.

In work that is of immediate relevance for the objectives of this paper, Goldberg and Schryver (1995) describe the (non-realtime) clustering and discrimination of eye-gaze locations, based on multivariate clustering and discriminant analysis. They sought to determine when the user wanted images zoomed. This is an important goal that will be discussed in more detail in following sections. Goldberg and Kotval (1999) describe the analysis of eye movements for assessment of user interface designs. Gee and Cipolla (1994) describe an unobtrusive system based on use of a normal axis to the face. The latter work achieves 15 degree precision, compared to 1 degree in the case of Goldberg and Schryver. A system for computer control by the motor handicapped, EagleEyes, developed over many years by Gips (see Gips and Olivieri, 1996) is also relatively obtrusive, requiring electro-oculographic sensors to be placed on the subject's face. The ASL 504 system which we use (ASL, 2000) has optics which are remote mounted, has a 0.5 degree visual angle accuracy, a 0.1 degree visual angle precision, and allows head movement within a 1 cubic foot volume. A measured data record, 16 bits measuring pupil diameter and x- and y-coordinates, can be delivered by a serial port (RS232) connection at 115,000 bits/second, which is many times faster than the sampling rate of the system. A good review of this eye-gaze tracking system in experimental and operational situations can be found in Yang et al. (2002).

The longer-term goal of the work described here is to develop a new way for the human to interact with multimedia data and information streams. Eye-gaze device control is of interest to the motor handicapped, and to surgeons during medical operations when the hands are occupied. Zooming of a large radiology image, or a large astronomy image, can be the result of concentration of the observer's interest, as expressed by his or her eye-gaze behavior. A novel way to view simultaneous video streams – indeed, TV programs – is made possible, whereby video is delivered only on the basis of the viewer expressing sufficient interest by continuing to look at it. Novel approaches to player interaction with computer games are made feasible also.

Our system is based on reading the eye-gaze positions as a subject looks at the monitor. The tracking system reads and stores these. Following access to this data, relatively simple operations are carried out, such as determining the subject's dwell time in a restricted area relative to fixed thresholds. Such processing must be carried out effectively in real-time. Based on rules – e.g. sufficient dwell time – signals that emulate mouse clicks are fed back to the computer system observed by the subject.

Implementation

Figure 1 shows the set-up. A subject (or user) sits in front of the presentation monitor. A small infrared camera under the presentation monitor tracks the user's eye-gaze activity.

A few minutes are taken initially to calibrate the particular user and their posture in this one eye tracking system. Shown on the left side of Figure 1 are the two monitors used for positioning and managing the cross hairs relating to point of gaze and a gray level image of what the subject is viewing. Also shown to the right of these monitors are the Control Unit box, and a control workstation (PC). In order to calibrate the subject, a second individual is needed.

The ASL eye-gaze tracking system Control Unit processes the information it receives from all devices connected to it, and calculates eye-gaze position in both vertical and horizontal coordinates as well as pupil diameter and a host of other parameters. This information can be used in real time, receiving it from the RS232 port on the Control Unit which in turn is connected to the serial port of the control PC.

We developed the Visual Mouse application to demonstrate that the stream of eye-gaze data can replace the actions of the physical mouse of the presentation monitor that the subject is observing. Hence mouse operations on the presentation monitor are emulated and are controlled by the subject's visual behavior while observing the monitor.

To show the feasibility of the Visual Mouse, a number of applications were prototyped with the following properties:

- The serial port is used to communicate with the eye-gaze tracking system.
- Information is continuously read from the eye-gaze tracking system.
- The information coming from the eye-gaze tracking system is decoded to get the gaze coordinates in real time.

- The coordinates of the data stream coming from the eye gaze tracker are continuously compared.
- If the subject keeps gazing at a limited area, then a mouse click is made at that position.

The Visual Mouse application uses a number of parameters or settings to allow us to change the state and behavior of the program.

Status of ports is indicated; either stopped or in progress.

A variable is used to indicate that the subject has changed his or her eye-gaze location. This variable is set to 0 each time the program takes new reference coordinates, i.e., every time the subject looks at a position that is not close to the previous coordinates. If the subject keeps on looking at a region around the reference coordinates, the index variable is incremented by 1 with each new coordinate pair falling in that region. A threshold used with this index reference parameter is the value that the index variable must reach to trigger the mouse click. Changing it, we can control the time that the subject must be looking at a region before triggering the mouse click. As the index variable is updated with each new coordinate pair, and coordinate pairs are obtained at 60 Hz, we can easily control the time, in seconds, before the mouse click is triggered by dividing the index reference parameter value by 60.

One further variable is a “tolerance” variable, which sets the width and height of the region around the reference coordinates. The program uses this parameter to decide if the subject is looking at the same place or not. If the difference between the current value and the reference coordinate is less than the tolerance, the program considers that the subject is looking at the same place, does not change the reference coordinates, and increments the index variable. If the difference is more than the tolerance value, the program considers that the subject is looking elsewhere, sets the reference coordinates to the current coordinates, and resets the index variable. The tolerance variable uses the eye tracking system scale, which is established in the calibration of the subject at the very beginning. In our work, the eye tracking system scale is demarcated by the PC screen observed by the subject. The system scale is not inherently limited to a PC screen: to give an extreme example, it could equally well function with a very large wall projection.

All of these system parameters were implemented as slide bars, allowing changed settings to be easily experimented with. Such changed settings could be made by the subject prior to a session, or by a second person at the control system during the course of the experiments.

The Visual Mouse application was developed (see Tinto Garcia-Moreno, 2001) using Microsoft Visual Studio 6.0 Enterprise Edition that includes several tools. We used Microsoft Visual C++ with Microsoft Foundation Class (MFC) as the computer language for the application. The most important class used, EyeSerialComm, uses structures ReceiveData and ReceiveStatus to store and output information. The EyeSerialComm class does all the specific work required for the Visual Mouse application. It handles serial port communication, data decoding and mouse message posting. The Control Unit “serial out” data output port can be set to use either a demand mode or a streaming mode. The Visual Mouse applications use the streaming mode.

Two further issues had to be addressed during implementation. Eye-gaze coordinates are at all times subject to additional small seemingly random displacements (Mountcastle, 1980). The latter resulted in “flickering” of the eye-gaze coordinates. Even though the coordinates given by the eye-gaze tracking system are averaged over a number of values output coordinate “flickering” was quite appreciable. Rather than applying a moving average smoothing filter, we dealt with this issue by finding a good compromise between the tolerance and the index reference parameters. Larger values of the tolerance parameter reduce the flickering effects, but also reduce the resolution of the Visual Mouse. Smaller values of the index reference parameter will generate the mouse click more quickly, which will decrease the effect of flickering. We found our trade-off values of these two system parameters to be robust for the applications to be described below. However we do not exclude the possibility that filtering of the incoming eye-gaze coordinate data stream could well lead to a more automated approach. We are currently studying the statistical properties of empirical eye-gaze coordinate data streams with such issues in mind.

A direct implication of reducing the resolution of the Visual Mouse is as follows. The Visual Mouse works very well when hot links with big icons are involved. Dealing with smaller clickable icon links, however, is troublesome. An example of the latter was our attempt to use the Back button on a regular web browser window, in the context of web surfing. The Back button proved to be too small for the Visual Mouse to operate effectively.

A second issue addressed was related to the accuracy of the calibration procedure. The procedure for calibrating the PC screen for each subject and session used nine points in order to calculate the mapping that relates the subject’s angle of gaze with positional coordinates on the approximately planar PC monitor screen. It is crucial to achieve good calibration of these nine points for positional accuracy of subsequent subject eye-gaze location. Notwithstanding the accuracy with which this is done, there is some decrease of accuracy whenever the subject looks at the PC screen at locations away from the calibration points. One way to avoid the decrease in accuracy in defining eye-gaze positions on a planar screen is to use a greater number of calibration points. A seventeen-point calibration is possible, and will give better results, but it requires appreciably more time to carry out. In summary, we can gain in accuracy of pinpointing eye-gaze locations at the expense of time and effort (and hence subject fatigue) taken in calibration.

All results reported on below used nine point calibration which provided an adequate trade-off between the subject’s work on calibration, and the positional precision of the data consequently obtained.

Large Image Display in Astronomy and Medicine

Support of the transfer of very large images in a networked (client-server) setting requires compression, prior noise separation, and, preferably, progressive transmission. The latter consists of visualizing quickly a low-resolution image, and then over time, increasing the quality of the image. A simple form of this, using block-based regions of interest, was used in our work. Figure 2 illustrates the design of a system allowing for decompression by resolution scale and by region block. It is the design used in grayscale and color compression algorithms implemented in MR (2001). Systems have been prototyped which allow for decompression at full resolution in a

particular block, or at given resolutions in regions around where the user points to with a cursor. Wavelet transform based methods are very attractive for support of compression and full resolution extraction of regions of interest, because they integrate a multi-resolution concept in a natural way. Figures 3a and 3b exemplify a standalone system on a portable PC using cultural heritage images. This same technique can be used to view digitized land-use maps.

Two demonstrators had been set up on the web prior to this Visual Mouse work. The cultural heritage image example shown in Figures 3a and 3b is further discussed at <http://strule.cs.qub.ac.uk/zoom.html>. This image is originally a JPEG image (including compression) of size 13 MB, and with decompression it is of size 1 MB. Decompression of a block is carried out in real time. The compression method used, which supports color, is lossy and is based on the widely used biorthogonal 9/7 Daubechies-Antonini wavelet transform.

A medical (orthopedics) example is accessible at <http://strule.cs.qub.ac.uk/imed.html>. This image has been compressed from 8.4 MB to 568 kB, and again decompression (and format conversion) of limited area blocks is carried out effectively in real time. The compression method used in this case is rigorously loss-less, and supports grayscale images. The multi-resolution transform used is a pyramidal median transform. Further details on the compression algorithms used can be found in Starck et al. (1996, 1998), Louys et al. (1999a, 1999b), Murtagh et al. (1998, 2001a, 2001b).

The eye-gaze control system can be used to operate these web-based demonstrations. The block sizes are sufficiently large that no precision problems are encountered with these types of applications.

General context for such scientific and medical applications is as follows. New display and interaction environments for large scientific and medical images are needed. With pixel dimensions up to 16,000 x 16,000 in astronomy, which is the case of detectors at the CFHT (Canada-France-Hawaii Telescope, Hawaii) and the UK's Vista telescope to be built at the European Southern Observatory's facility at Cerro Paranal, Chile, it is clear that viewing "navigation" support is needed. Even a digitized mammogram in telemedicine, of typical pixel dimensions 4500 x 4500, requires a display environment. Our work concerns therefore both image compression, and also a range of other allied topics – progressive transmission, views based on resolution scale, and quick access to full-resolution regions of interest.

Ongoing work by us now includes the following goals: (i) better design of web-based display, through enhancing these demonstrators (including more comprehensive navigation support for large image viewing, and a prototyping of eye-gaze controlled movement around three-dimensional scenes); and (ii) support for further Visual Mouse interaction modes. Chief among interaction modes is a "back" or "return" action based on lack of user interest, expressed as lack of gaze concentration in a small region.

Eye-Gaze Control of Multiple Video Streams

Figure 4 illustrates the new approach to interacting with multiple streams of multimedia data. Shown are a number of presentations from a recent workshop, each with a streaming video record of what was discussed and presented. If the observer's eye dwells sufficiently long on one of the panels, the video presentation is displayed in the panel. If the observer's interest remains as measured by eye-gaze on the panel, the video continues to display. At any time, the observer's interest may wander. If his or her eye dwells sufficiently long on another panel then the previously playing video is replaced with a name-plate, and a video stream now plays in the new panel. An HTML OBJECT element is used to insert an ActiveX component into the HTML document as well as all of the necessary information to implement and run the object.

A typical session is indicated in the following, and a short video record of this has been made available (see <http://strule.cs.qub.ac.uk/~fmurtagh/eye.html>). In succession, we have

1. Initial set-up.
2. Eye-gaze tracking camera.
3. Loss and regain of tracking.
4. What the subject is viewing.
5. Remote control of the eye-gaze tracking camera.
6. Calibration screen, and calibration procedure, in operation.
7. Attempt at small-size button activation.
8. Multiple video stream example in operation.
9. Orthopedics example in operation.

Conclusion

Performance of the systems described in this paper has been very good. We have found both latency and obtrusiveness to be minimal in this system. Work is continuing with other subjects, and other types of web and image display content to further assess the ergonomic aspects of the eye-mouse. We will report on this further work in due course.

Robust use of the eye-tracking equipment in an arbitrary environment is still needed. It will be a while before a little camera on your home TV or PC is able to track your eye and act on your behavior, but we are moving towards that objective. To facilitate this objective, a portable installation is desirable, in which the only required hardware is a small infrared eye tracking camera, and the software can be hosted by a TV set-top box or handheld PC.

Leading applications of this approach to human-computer interaction are required, and we have discussed a number of proposals in this article. Following Goldberg and Schryver (1995), we have developed a system around the needs for region of interest decompression and display in the context of large image interpretation and analysis in science and medicine. Secondly we have developed a system for the viewing of multiple video streams.

In future work we will enhance these prototypes, and we will investigate how we can proceed towards mobile and robust use of this innovative technology. We also intend to study dependence on display device, e.g. screen resolution, and display transfer function (raster scanned and digitally addressed CRTs, flat panel displays, and so on).

References

1. ASL Applied Science Laboratories, Bedford, MA. <http://www.a-s-l.com> (2000).
2. A.H. Gee and R. Cipolla, "Non-intrusive gaze tracking for human-computer interaction", in *Proc. International Conference on Mechatronics and Machine Vision in Practice*, Toowoomba, Australia, IEEE Computer Society, 112-117 (1994).
3. J. Gips and P. Olivieri, "EagleEyes: an eye control system for persons with disabilities", presentation at *Eleventh International Conference on Technology and Persons with Disabilities*, Los Angeles (1996). EagleEyes homepage <http://www.cs.bc.edu/~eagleeyes>
4. J.H. Goldberg and J.C. Schryver, "Eye-gaze contingent control of the computer interface: methodology and example for zoom detection", *Behavior Research Methods, Instruments and Computers* **27**, 338-350 (1995).
5. J.H. Goldberg and X.P. Kotval, "Computer interface evaluation using eye movements: methods and constructs", *International Journal of Industrial Ergonomics* **24**, 631-645 (1999).
6. H. Istance and P. Howarth, "Keeping an eye on your interface: the potential for eye-gaze control of graphical user interfaces", *Proceedings of HCI'94* (1994). Paper at <http://www.cms.dmu.ac.uk/~hoi/hci94/hci94.html>
7. M. Louys, J.L. Starck, S. Mei, F. Bonnarel and F. Murtagh, "Astronomical image compression", *Astronomy and Astrophysics Supplement* **136**, 579-590 (1999a).
8. M. Louys, J.L. Starck and F. Murtagh, "Lossless compression of astronomical images", *Irish Astronomical Journal* **26**, 119-122 (1999b).
9. V.B. Mountcastle, Ed., *Medical Physiology*, Vol. 1, Mosby (1980).
10. F. Murtagh, J.L. Starck and M. Louys, "Very high quality image compression based on noise modeling", *International Journal of Imaging Systems and Technology* **9**, 38-45 (1998).
11. F. Murtagh, M. Louys, J.L. Starck, F. Bonnarel and M. Farid, "On-demand delivery of large compressed images in astronomy: computational requirements", *SPIE Proceedings*, Vol. 4477, SPIE Symposium, San Diego, July/August 2001 (2001a).
12. F. Murtagh, M. Louys, J.L. Starck, F. Bonnarel and M. Farid, "Compression of grayscale scientific and medical images – principles, environments, evaluation", *Electronics & Communication Engineering Journal*, submitted (2001b).
13. MR, Multiresolution Software Environment, <http://www.multiresolution.com> (2001).
14. F. Quek, "Non-verbal vision-based interfaces", keynote talk, *IWHIT'95 – International Workshop on Human Interface Technology '95*, Aizuwakamatsu, Fukushima, Japan. Paper at <http://vislab.cs.wright.edu/Publications/nonverbal.html> (1995).
15. J.L. Starck, F. Murtagh, B. Pirenne and M. Albrecht, "Astronomical image compression based on noise suppression", *Publications of the Astronomical Society of the Pacific* **108**, 446-459 (1996).
16. J.L. Starck, F. Murtagh and A. Bijaoui, *Image Processing and Data Analysis: The Multiscale Approach*, Cambridge University Press (1998).
17. F. Tinto Garcia-Moreno, Eye Gaze Tracking System Visual Mouse Application Development, Report, Ecole Nationale Supérieure de Physique de Strasbourg

(ENSPS) and School of Computer Science, Queen's University Belfast, 77 pp. (August 2001).

18. Guang-Zhong Yang, L. Dempere-Marco, Xiao-Peng Hu and A. Rowe, "Visual search: psychophysical models and practical applications", *Image and Vision Computing* **20**, 291-305 (2002).

Figure Captions

Figure 1. General layout of the prototype set-up is shown here. Significant components include the presentation monitor and the eye-gaze tracking camera below the monitor. Arrayed from left to right behind and to the left of the subject are two monitors, one on top of the other, forming part of the eye-gaze tracking system, the eye-gaze tracking system Control Unit, and a PC that hosts the control software.

Figure 2: A design for compression of a large image, by block, and supporting five resolution levels. At each resolution level, a display window is superimposed at a given position. At the lowest resolution, the window covers the whole image. Wavelet and other multiresolution transforms use such a pyramidal data structure.

Figure 3a. A digitized image shown at full size and reduced resolution.

Figure 3b. A block from Fig. 3a shown at full resolution. This block is extracted and decompressed from the compressed image file. A mouse click on the low resolution image controls this retrieval and decompression.

Figure 4: An example of eye-gaze control of a streaming video record of a workshop held in Computer Science, Queen's University Belfast, on 29-30 January 2001. User interest, as measured by eye-gaze of sufficient duration on a description panel, causes the streaming video to be activated. If the user continues to watch that panel, the streaming video continues. If the user switches his/her attention to another panel, the streaming video in the initial panel is halted and is started in the new panel of interest.



Figure 1.

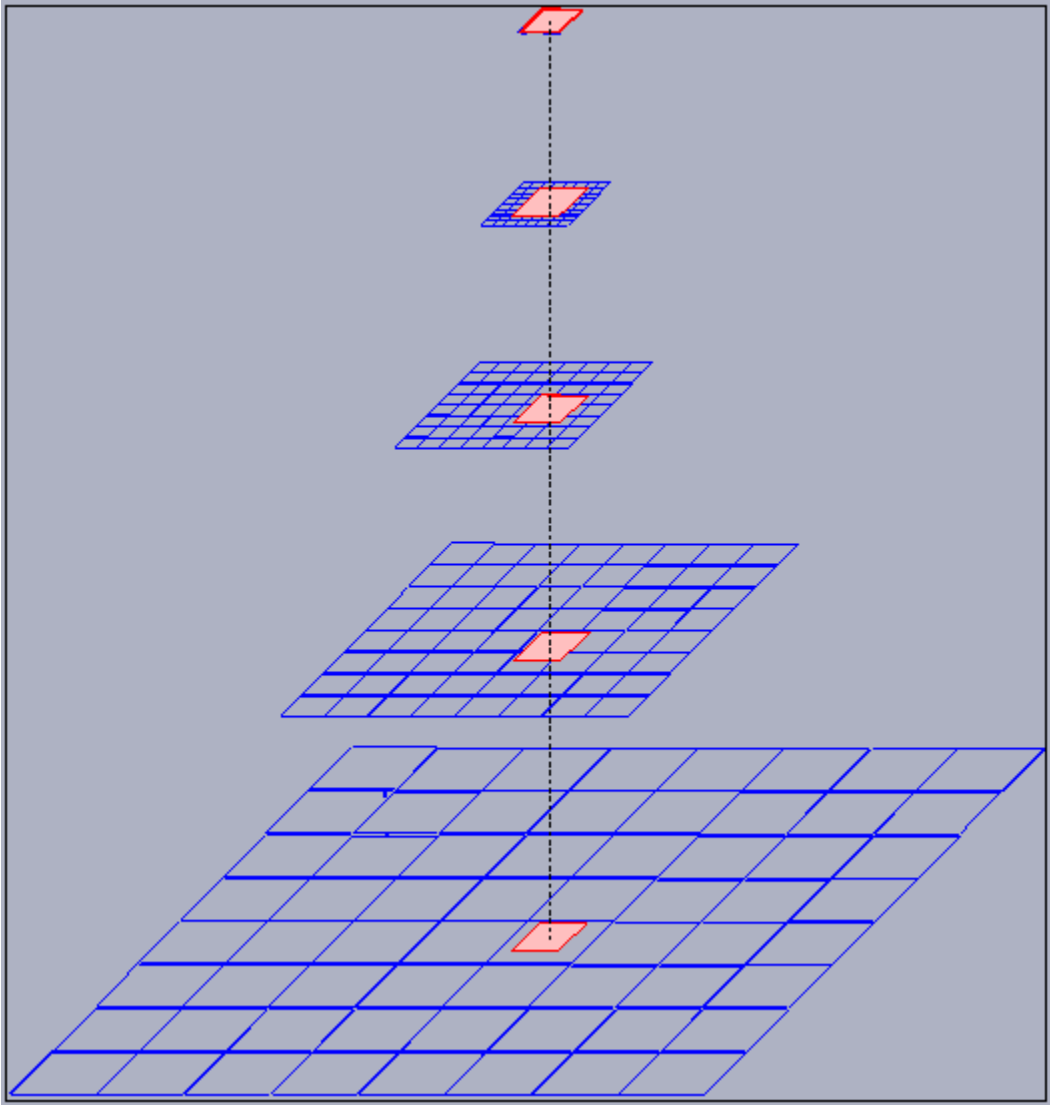


Figure 2.



Figure 3a.



Figure 3b.

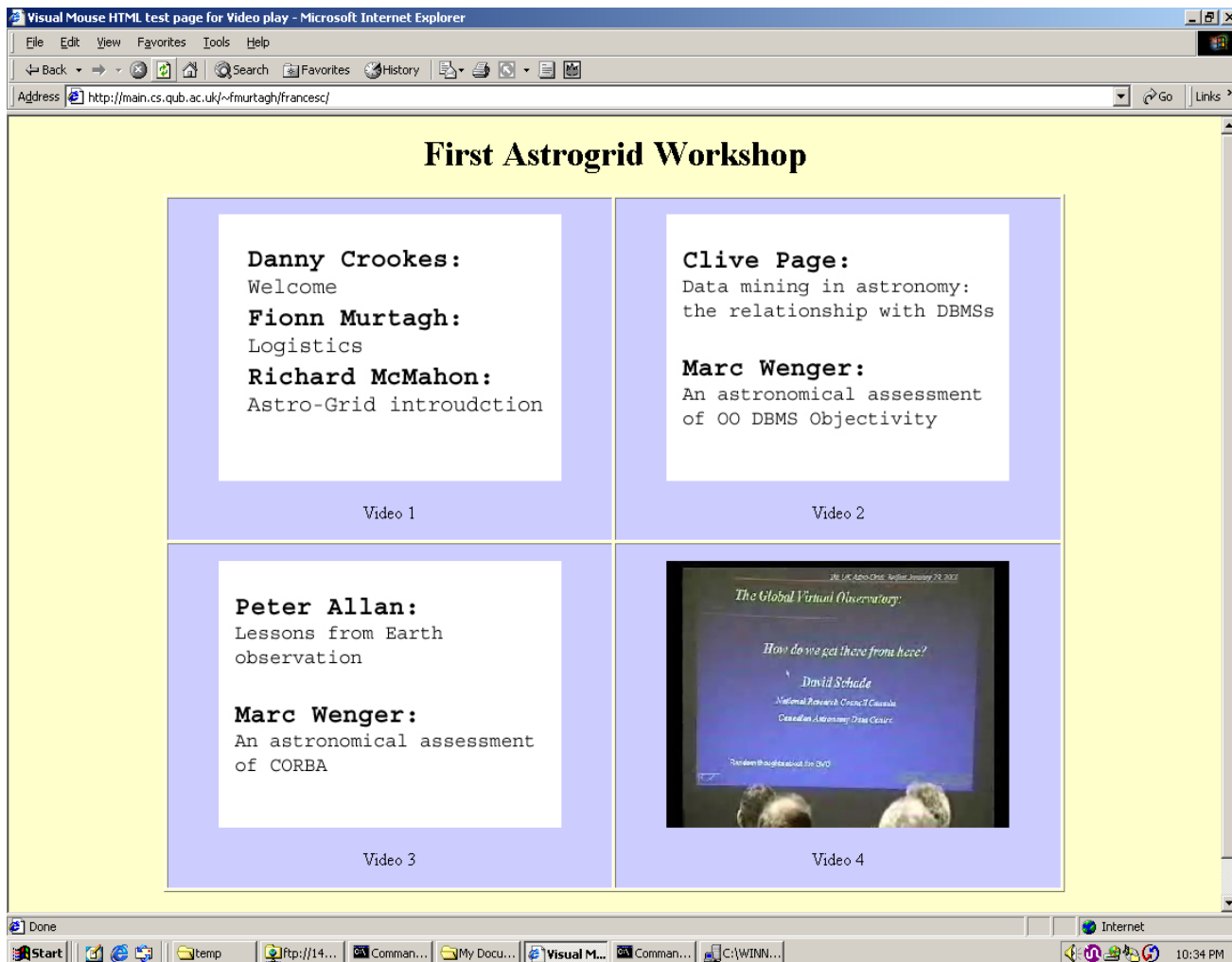


Figure 4.