The Friendly Classroom

Abstract. We present an overview of a project to integrate present and future technologies into the classroom or lecture hall to enhance teaching for the lecturer, and learning for the class member and those accessing the recorded results of the class. We work within constraints of schedule (a new building coming soon) and the limited patience that most lecturers have with fragile new systems. This paper presents details of two initial stages of this project: a pointer-based system for controlling the technology without requiring a unique device or frequent visits to "the podium;" and a camera calibration architecture and software environment that will permit us to incorporate multi-camera vision applications in our "friendly classroom."

1 Introduction

Enhancing teaching quality by the use of technology has been around since the first human started teaching. From using built-in technology such as voice and hand gestures, teachers have gone on to adopt technology into their teaching programs almost as fast at it could be deployed. Once a technology had matured enough to be easily deployed, learned and maintained, the teaching establishment explored ways to insert it into their programs. This does not mean that all technology was successfully adapted. Indeed, numerous failures have occurred, but at the same time, success stories also abound.

When analyzing the insertion of a technology into the educational context, deployment is and will be an on-going problem, especially as teaching technologies become more complicated. Drawing or writing on paper or on a board takes only a few seconds to learn, but getting a modern podium to work with all its capabilities (lighting, overhead projection, video projection, slide projection, sound amplification, digital media presentation, and internet navigation) can be a serious obstacle to those not technically savvy. In many cases the capabilities of such equipped rooms get underutilized because the lecturer does not want to take the time to learn how to use its myriad options, or because they cannot get something to work, or because the elaborate equipment malfunctions and a knowledgeable support person is needed to fix it.

A second, but steadily growing problem is that of large classrooms. A steady movement has been occurring in the last decades to teach more and more courses in large rooms or auditoriums. In such cases, 100 or more[1] students sit and watch a lecturer perform. Because of the large audience, most lecturers find it difficult or nearly impossible to interact in a meaningful way with their students. This causes the class session to be a one man show, in which the teacher needs better and better acting skills in order to keep his audience awake and focused. Much educational and psychological research has shown that such environments are far from effective in achieving their major goal- that is, to transfer knowledge and understanding from the lecturer to the students [1, 5, 17].

A third and related problem, partially arising out of the large class phenomenon, is the growing expectation that classes be recorded and archived in order to enable students to more closely pay attention during the class session. The logic is that if a student does not feel pressure to write every sound coming out of the lecturer's mouth, they will be able to better attend to what she is actually saying. Later on they can return and browse the lecture again to catch any part they may have missed. These recordings will also benefit distance learning, in which other students, not present at the initial lecture, may learn from the material .In order to implement this various media devices (video cameras, microphones, amplifiers and mixers, and lighting) have been procured and used in many places. A number of problems quickly cropped up. First was the question of how to record such sessions without disrupting the flow of teaching and interaction in class. To be effective, such recording sessions should not force the lecturer to spend more than a few seconds turning the recording process on and later turning it off. This problem is easily solved by hiring media recording technicians, who are knowledgeable, but alas, also expensive. A number of efforts have been made to replace these technicians with automatic recording systems [4, 13, 15], but these have not solved the problem yet. Additional problems also exist: the processing, encoding, indexing and publishing of the materials in such a way that makes them available to the whole student body. It has become clear that this problem is not easily solved in a low cost manner. What is also clear is that in order for automatic recording to have a chance, the cameras and microphone used in the process must be self controlling and correcting, thus enabling a good recording algorithm to successfully capture a teaching session.

In summary, we have decided to focus on the following problems in current use of technology in the lecture room:

- Technology, as deployed in many teaching environments today is too complicated and sensitive. Most potential users are not able to make significant use of the available functionality, and thus, much of the technology is under utilized.
- Technology has not been used enough to strengthen student to student interaction or student problem solving skills. It has mostly been used as a way of presenting materials on the screen- in effect serving as an extension of the age old black board.

[1] There is a growing discussion base about the number of students in a class to be considered large. Some talk about 80 to 100 as the crossover point. Larger examples (i.e. 200-300) abound, while the extreme seems to be a lecture given in front of 1200 students (the course uses 40 TA's!). see: http://www.invisibleadjunct.com/archives/000331.html .

- While Technology as used today enables teaching very large classes, not enough effort has been invested in creating a technological infrastructure to enable good quality autonomous capture and publishing of class sessions.

Our work, to be reported below, shows promise in helping alleviate some of these problems. We report on a friendly pointer system that uses knowledge about presentations to enable a lecturer to operate all technical aspects of the room from wherever they stand, without needing to do anything more than to point at hotspots which represent a presentation room function or service. Initial user testing has found the model to be successful. We also report on a model for camera self testing as part of our efforts to create a highly robust input system that can withstand a certain amount of environmental punishment and minimizes the need for maintenance personnel to keep the system running.

2 Relevant Existing Work

2.1 Smart Classes

In most cases, the term “smart class” today will mean a university lecture room or auditorium equipped with an AV podium. This usually includes a multimedia computer, a VCR (and DVD in some cases), an amplifier, a document projector and a video projector. More sophisticated rooms also include a “Smart” white board, centralized lighting controls and an integrated interface through a touch screen.

Figure 1: “smart classroom” podium media centers. Images copyright: www.marquette.edu/ imc/AV/smartpodium.html

Relevant research in this area has revolved around the development of improved interface devices to control the projected computer screen presentation [2, 10, 18, 21, 25]. These devices allow the presenter to control the computer presentation from wherever they are in the room without needing to use a computer mouse. The research in this area explores novel ways of getting mouse like interaction when substituting the mouse with a number of devices: stateless and state-full laser pointers, infrared pointers, and passive sticks with visual capture and localization. Special effort has been put into devising clever ways to enable richer interaction scenarios within the limitations associated with laser pointers and other pointing devices. Examples are identifying state with stateless pointers by the use of dwelling over a spot or by using predefined gestures. The gesture research has opened additional interaction avenues but we feel that the user learning needed for successful use of these models is a barrier to broad adoption.

2.2 Intelligent Rooms

A number of research groups have focused on creating intelligent rooms [3, 11, 12, 20, 23]. Most of these have focused on creating the infrastructure for developing a room with computational intelligence- one that understands the context of what is happening in the room, can identify the participants, and react to their needs and commands, and in the process assist them in carrying out their tasks. Much of the work in these projects revolves around designing, developing and implementing the enabling software architectures that can support and manage such applications [i.e. 3]. Relevant scenarios that have been explored are small group project meetings and operations centers. A few have also focused in the area of enhancing the class room environment which is more relevant to what we are doing, but they too focus on developing the systems for intelligent rooms that can help a lecturer teach by, for example, understanding their teaching plans [i.e. 11]. As will be seen later in this paper, our focus is very different.

The main motivation in all these projects seems to have been on creating novel enabling technologies for tomorrow's applications. By "tomorrow" we mean applications and technologies that will be mature enough for real world use in no less than five years from now. From viewing some of the demonstration videos [i.e. http://oxygen.lcs.mit.edu/videocollaboration.html] we have also concluded that they have not had the time, or have not been able, to focus much on the user experience of such systems. This is understandable since they have focused on the enabling technologies, but nevertheless, the user experience seen in the available demos seems choppy and not naturally flowing, not one that users will endure for long. Some of these problems will go away as the enabling technologies mature, while others will need to have their user experience design reappraised before they can succeed.

3 Our Goals

3.1 Enhancing Teaching and Learning Effectiveness through Technology

Our major goal is to augment the standard university lecture hall with tools and systems to make learning and lecturing more effective. Our focus in not on the room as such, but on what occurs in the room, and how we can enable it to happen more effectively. We have defined a friendly lecture room, one in which the needs of both lecturer and learner have been taken into account and will be deployed in a way which will enable more effective learning to take place. By effective learning we mean that both sides of the equation will be taken care of – the teaching as conducted from the "front"[2] of the class, as well as the student learning done in the "wings".

3.2 Real World Robustness

A second goal of this project is to quickly design, implement, test and iterate subsequent versions of the room in order to develop a system that can withstand the punishment of real world scenarios and users. We are planning to implement four rooms[3] in our new Computer Science building, construction of which should start in 2005. We have a rare opportunity to integrate our lab into the day to day fabric of the department. This gives us a bit over 12 months to develop and implement a four room system which can be embedded in the building and serve as real world test beds. Because of this we have opted to use state of the art (circa 2004) technologies and not bleeding edge technologies which will not be mature enough in time. We will use various mixtures of vision processing, acoustic arrays, sensor fusion and ad hoc networking to implement our design.

3.3 Self Monitoring and Maintenance

Our third goal is to design a system that is not only robust enough to withstand the punishment of day to day use, but also has the capabilities to maintain itself as much as possible. By this we mean, for example, that camera and audio calibration will be self administered in most cases, with possibilities for self-correction of problems as they occur. The goal is to develop a system that exhibits graceful degradation in the face of real world problems, and can call for help when really needing it.

---

[2] It should be clear from our introduction that we do not think that Frontal teaching is the optimal way to teach, but since this is the predominant model today, we hope to make it more effective.

[3] The new building will house a medium sized auditorium (250 seats), a small auditorium (80 seats), and two small (40 seat) lecture rooms.

3.4 From Goals to action

To achieve these goals we have put together a series of projects, each one focusing on one aspect of the final system. We have designed the projects to be modular in nature, so that they can be integrated into a central management application with hardware abstraction for implementation. These projects are:

- **Friendly Pointer:** This project is focused on building a system capable of allowing the lecturer to use all the devices available in the room without needing to do any hands on device control or room management. The system itself understands the user's requests and orchestrates all the needed services in real time.

- **Camera Self Correction:** This project focuses on developing methods for camera self maintenance. By enabling the cameras to be self monitored and corrected, up to a predefined threshold, maintenance costs are lowered to a minimum, making multiple camera installations possible.

- **Lecturer tracking:** An important aspect of friendly class scenario will be the ability to document and archive the classes taught in the room. Doing this will enable current students to go over any material they may have missed or felt unsure about, and future students to review materials from previous years. In order for this to be pragmatic, the lecture must be recorded automatically, with no need for any technicians present. This project will evaluate existing lecture tracking systems and choose the one most favorable for our specific needs.

- **Sound Pickup:** As lecture halls become larger, sound pickup from the audience becomes more important. This project will utilize a microphone array to enable members of the audience to be heard by the rest of the room when needed.

- **Audience Input:** This project will focus on various ways to enable a large audience to actively participate in a learning session using passive as well as active methods and devices for input. Our aim is to create better, cheaper and more robust solutions relative to those that exist today. Technical and usability tradeoffs will be explored in the search for an optimal solution.

- **Sound Self Calibration:** This project will focus on creating a sound pickup system that will need a minimum of care and maintenance. In a similar fashion to the camera self correction project, this one will explore ways of developing a microphone array which exhibits graceful degradation and self correction.

- **Lecture Recorder:** This project will focus on tools to enable easy lecture recording, archiving and publication. The system will use a mixture of available services in the room to create good looking interactive hyper-videos of a lecture. These hyper-videos will have in built time stamping and links to sections of the lecture, next to links to external materials. Personalization will enable students to develop their own mapping to the generic materials.

- **Lecture Studio:** This project will focus on tools to enable "before" lecture preparation and "after" lecture editing and monitoring of learning materials, discussion groups, archives, and exercises.

- **General Room Monitoring:** This project will focus on tools to enable monitoring class utilization for the department's purposes with the goal of improving and optimizing facility usage.

4 Initial Results

As can be seen, this is an ambitious project, especially when we take into consideration the short timetable available to implement it. Because of this we are developing a number of the projects in parallel and in a modular fashion. Not all of them will be available at the end of 2005, but the rooms will be built in such a way as to enable the easy upgrading of software and hardware as needed.

At the time of writing this paper we have actively been working on three projects: the friendly pointer, camera self correction, and audience input. The first two projects have reached a point where we have tested them and have results to report here. The audience input project will reach this stage in the next three months.

4.1 The Friendly Pointer

Project Goals: The goals of this project[4] are to develop a system that will enable the lecturer to make the best use of the existing technical infrastructure in the lecture room while leaving her free to walk around the room as she saw fit. As stated in the introduction above, in too many cases the lecturer finds the very operation of the myriad devices in the room hindering to their performance as teachers. Not being anchored to a podium enables the lecturer to make more natural and intimate contact with her students, while also enabling her to use all the services in the room in a very unobtrusive way. Doing this necessitates the design and development of a successful interaction model.

Interaction Design: The friendly pointer system enables this to happen by allowing the lecturer to point at physical representations of a wanted service, where ever they

[4] For an in-depth look at the interaction design aspects of this project see, http://www.cs.huji.ac.il/~kirk/Room_Interface_Design_CHI05.pdf

might be in a room and when ever the wanted to. Once the system registers the lecturer input it will put into effect all the needed management functions needed to put that service into operation. Thus, when the lecturer points at a PLAY hotspot the system will do the following: set the input on the video projector to the VCR, dim the lighting, and set the volume level so that the VCR soundtrack is on and hearable. Clicking later on STOP will return the system to the state it was in *prior* to PLAY (video input set back to what it was before, light turned back on, sound input set to lecturer, etc).

Implementation: The system was implemented using a simple $20 web camera (640x480 @ 15 frames per second), a controllable mini-DV camera (which served as the VCR in this case and was controlled over an IEEE-1394 “firewire” cable), a PIC based microcontroller which controlled a relay circuit to control the 220V lighting, a controllable video projector (using control codes over an RS-232c connection), a simple $5 laser pointer, and a Windows workstation which served as the central controller. Video input was passed to an image processing module, which identified the location of the red spot in the scene. Once a red spot was identified as hovering over a hot spot, the appropriate actions linked to that hotspot were put into effect. Thus, each hotspot had a script associated with it, and the control application saved previous states to enable returning to them. Device control was enacted over serial (to the microcontroller and the projector) and IEEE-1394 cables (to the miniDV camera).

Testing: Testing was run using two groups: both groups gave a simple lecture using AV equipment and a teaching scenario. The test group (N=5) were given a short explanation of the friendly room, were then shown a short demo of the system in action, and were then allowed to acquaint themselves with the laser pointer and hotspots for two minutes. They were then asked to use the laser pointer and teach a short class using a pre-designed scenario (i.e. turn the lights off, bring up a PowerPoint presentation, go to slide 4, go back to slide 2, Play (the video), Pause, Stop, and return to the Main menu). The control group (N=5) were asked to do the same scenario, only in a regular podium based class room, having to operate all the technical aspects of the room (lighting, PowerPoint, VCR, Video Input on projector) manually. Both groups were then questioned as to their feelings of control, ease of use, and relative ease of use compared to a mouse. After the initial stage the groups were asked to do the same scenario, only in the other class. Thus, Test subjects were asked to teach in the regular room, and the control subjects taught in the friendly room. They were then questioned again, this time also being asked to compare the ease of teaching the scenario in the friendly room and the regular room.

Results: The results of this study (see table 1 below) showed us that subjects found the pointer-based room much easier to use than the regular podium-based room. The ratings for ease of use of the test group were significantly higher than those of the control group (9.2 versus 6.9 ($F(1,8)=32$, $p<0.01$)). When we checked for the *relative* ease of use of the rooms to each other we found an even stronger result: The friendly room was deemed to be almost twice as easy to use as the podium based room (mean relative ease of use rating = 1.8, where 1 represents *equal* ease of use), while at the same time the podium based room was rated as almost half as easy to use relative to the friendly room (mean relative ease of use rating for the podium task = 0.59). Stating this result using other words shows that the podium room was deemed to be almost twice as *hard* to use.

Interestingly enough, these results were obtained even though the friendly room showed more user action errors (1.4 vs. 0.4, $F(1,8)=8.3$, $p<0.05$) than the podium based room. Additionally, the results for feeling of control was also seen to be somewhat higher in the podium based room (9.4 vs. 7.8, although this was not statistically significant). These results seem to be caused by the inherent difficulties in pointing that are associated with the laser pointer. When the laser is off, the user does not know where it will point when turned on. Thus users go through a self correction process when pressing on the laser button: they point in the general direction they would like to hit, then click on the laser button, then visually acquire the red dot, and then fix their pointing accordingly. This can easily cause pointing mistakes and make the users feel somewhat less sure of themselves during its use.

	Action Errors *	Feeling of Control	Ease of Use rating	Ease of Use Relative to the other system (equal =1)
Test (N=5)	1.4	7.8	9.2	1.8
Control (N=5)	0.4	9.4	6.8	0.59
H1: Test will not be different from control	P<0.05	P<0.072 Not Significant	P<0.05	P<0.01

Table 1. Results of User Testing of the Friendly Pointer. *: Action errors: A user causing the system to put into action something that the user did not mean to do.

Conclusions: The results show us that even though the laser pointer is not seen as a very accurate control device, showing more action errors than the mouse [21], and giving the users a somewhat degraded feeling of control, these handicaps are clearly disregarded when they are offered within the context of a system that gives them enough added value. Thus, being able to easily control the room from wherever they stood gave the users such a strong feeling of control and reduced their stress that they were happy to accept the problems inherent with the laser pointer. All subjects voiced their clear motivation to have such a room at their disposal when teaching.

4.2 Camera Self Correction

Project Goal: The goal of this project was to enable the long term placement of multiple cameras in the various friendly class rooms. If pervasive computing efforts are to succeed, the amount of hands on maintenance by technicians must be minimized. If this goal is not achieved, such environments will not be cost effective in the long run and will find themselves out of action. In this project we have developed an algorithm for camera self testing and self correction. This algorithm will enable the system to identify changes in the camera angle of view and to self correct for these changes up to a defined threshold of accuracy. If this threshold was crossed, the system will raise an alarm to call for a maintenance person to go out into the field and re-adjust the camera. Additional goals are to create a “Plug and Play” algorithm that does not necessitate any calibration before being put into use, and to be light enough to run in real time next to additional processes in the system.

The Algorithm. To satisfy these demands we developed an algorithm based on the following stages:

a. Find feature points in the scene
b. Track each feature point.
c. Calculate the tracking confidence for each feature point
d. Analyze the relative motions of all the points
e. Calculate the correction motion for the points

Finding Feature Points. We need to select many points, so that even if significant parts of the image changes we still have enough good points to track. The points should be spread all over the image to get better robustness against background changes. We also need to make sure that each point has enough strong gradients in its neighborhood so we are able to track it well.

We divided the image into 40 x 20 pixel cells. For each pixel in a cell we calculated the sum of the image gradients in a 9 x 9 area around it, and pick the one with the strongest gradients (See figure 2). To achieve better real time performance this stage can be done in advance, and once in a few frames we can update the feature points for another part of the image. We can also reduce the size of the original image and do all the calculations on the smaller image using fewer calculations.

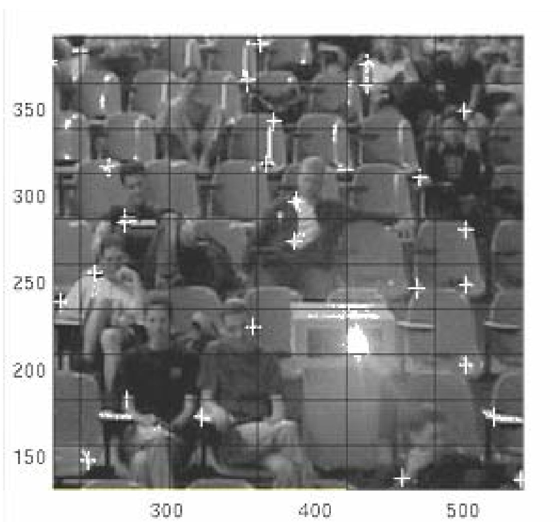

Figure 2. Dividing the image to cells (black grid) and find the best feature in each cell (white cross)

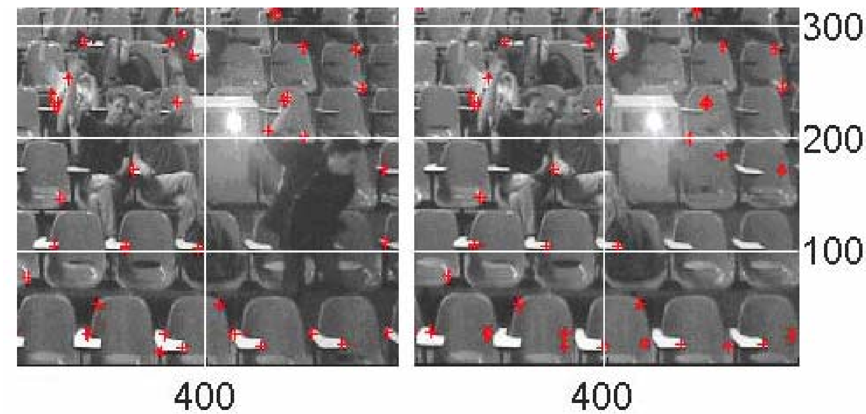

Figure 3. Tracking results, as you can see some points drifted away, in the next stages we will try to filter out those points

Calculating the Tracking Confidence. After we tracked each point we need to analyze the results of the tracking and make sure we haven't drifted away. The most common cause for drifting is when the destination image is too smooth so we don't have a significant difference between the desired points to its neighbors. To avoid this problem we calculate the sum square difference between the source image and all its neighbors in the destination image (1) and the tracking confidence is then calculated (2). We then select all the points that returned a tracking confidence level above 2. This means that we only keep the points that are an obvious local minimum of the error function so we are convinced that the result is unambiguous.

$$D_{ij} = \sum_{x,y} (curI(x,y) - newI(x+j, y+i))^2 \qquad (1)$$

$$Confidence = \frac{D_{00}}{\min\limits_{i,j=\pm 1}(Dij)} \qquad (2)$$

Calculating the Image Correction. In this stage we need to decide which points to use to calculate the correction, and how good our solution is. To do so we use the RANSAC algorithm [9]. We do an iterative process. In each step we calculate the mean and variance of all the good feature points, and discard all the points that are more than one standard deviation away from the average. We continue with the process until we discard more than half of the points or the standard deviation drops below 0.1. In each iteration i we calculate a correction confidence score (3):

$$Confidence = \frac{N_i}{N_0}(1.1 - std(M_i)) \tag{3}$$

Where N_i is the number of the good points in iteration i, N_0 is the initial number of good points. M_i are the motions of all the points in step i.. After this process we can use the average of the remaining points as the suggested correction. The calculated confidence can be use to determine the quality of our solution – a low confidence means that the motion of the camera was too large for us to correct and we'd better "Call for Help".

Results. To test the algorithm we took two different images of a class taken in the same camera position. We randomly selected a motion and shifted the second image according to it[5]. We ran the algorithm on the two images and compared the result to the shift we did. To better simulate a real world situation we select two images with significant changes between them. (See figure 4)

Figure 4 Test images; note the differences between the images.

We repeated the test 10,000 times with motions uniformly distribute between -150 and 150 pixels for each direction. The results show that for motions that were smaller than 40 pixels we have no error (See figure 5a), and the error percentage increases dramatically when the motion is larger. The reason for this is that the tracking uses a 5 level pyramid[6] so a motion of 40 pixels in the lower level is more than one pixel in the highest level –causing the Lucas & Kanade [19] tracking algorithm to become less accurate. We can increase the accuracy of the algorithm in two ways.

[5] Note that displacing the second image was used as a simulation of camera movement. This is not true camera motion but enables us to run many test very quickly and compare the results we get against the real motion we put in place. Future work will compare real camera motion against this.

[6] Pyramid: a stack of images where each one is half the size of the one below it.

- Increasing the number of levels in the pyramid which theoretically doubles the tracking range. The problem with this solution is that the additional level will be too small and we won't have enough data to work with, making this option not very practical.
- Do exhaustive search on the highest level: For each feature point we search in a 5x5 area in the highest level of the pyramid and select the position that gives us the minimal error (we use the Sum Square Difference as the error function) and use it to initialize the Lucas & Kanade tracker. The problem with this option is that the tracking becomes computationally heavy and it may be too slow for real time performance, especially if we need for it to work along side other applications.

We have found that the current accuracy is conceptually enough for our needs at this point. Future work will test the utility of our threshold and develop a solution with a wider correcting window if needed.

Conclusions: In spite of the fact that the algorithm shows too many errors when the motion is above the 40 pixel threshold, the algorithm can accurately detect this problem. When looking at the correction confidence (See figure 5b) we can see that we always get a low confidence when we had an error. From these result we can conclude that our algorithm is in fact capable of automatically correcting for small camera motions within the 40 pixel correcting window, and that the algorithm is capable of choosing whether to automatically correct the errors or to alert a maintenance personnel to come and adjust the camera when it has exited the correcting window.

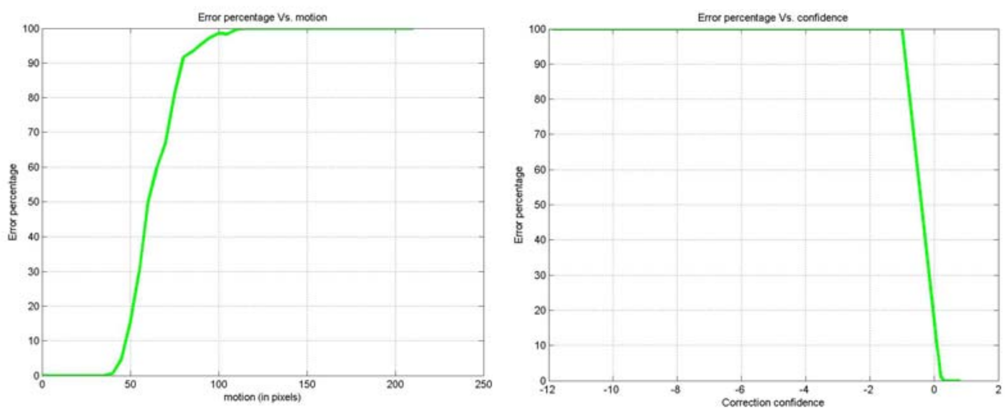

Figure 5. a. Error percentage Vs. motion. b. Error percentage Vs. Confidence.

5 General Conclusions

This paper has reported on our initial work within the "friendly class" project. We have focused on tackling a number of problems inherent in the present day classroom teaching scenario. Our aim is to enable more effective teaching and learning within the classroom as well as after the class is over. The results reported here show initial promise in being able to carry this out. The friendly pointer was shown to be an easier to use pointing device for operating all the devices in the lecture hall, thus freeing the lecturer from being tethered to the podium and enabling them to move around and engage in a more intimate and dynamic interaction with their students.

An additional benefit to be reaped by such a system is that it will enable less technologically inclined lecturers to make use of what until today they might have avoided learning and using, greatly increasing the set of courses that can benefit.

The camera correction project is a first step in the process of creating an input system for classroom based pervasive computing that can withstand the ravages of time and the environment. It is clear to all in this field that unless such models are successfully developed, this form of pervasive computing will not succeed in gaining traction, since the alternative will be systems that need frequent and expensive maintenance. The results of this project have shown that simple hardware can become more robust and need less maintenance when coupled with properly designed software. However, this is only a start on the full design effort required to ensure maintainability of such solutions.

6 Future Work

Apart from progressing as far ahead as possible within the limited timescale available to us at this stage (approximate 12 months), the friendly class project will also look at the results reported here and feed them back into the design and testing process. A number of questions come up which need to be answered as best as possible before the system can be put on autopilot in the class. Some of the questions pertain to the best architecture to use for controlling all aspects of the friendly class. We will explore alternative methods and vocabularies for signaling and controlling across the various devices and services making up the room. Other questions focus on the interaction models and hotspot representation in the room. We will develop or utilize self testing and correction algorithms that will generate a wider window of self-correction for our input devices, making them even more robust and cost effective. We anticipate that in future extensions and technologies which we include in the "friendly classroom," we will see the same fusion of interaction design needs and autonomous computing substrate that must be developed, as is exemplified in the two parts of the project reported here.

References

1. Betts and Shkolnik - The Behavioral Effects of Variations in Class Size: The Case of Math Teachers (EEPA, Vol. 21, No. 2, Summer 1999)
2. Cavens, D., Vogt, F., Fels, S., and Meitner, M. (2002): Interacting with the big screen: pointers to ponder, ACM SIGCHI '02 Conference on Human Factors in Computing System, Companion Proceedings,p.181-182.
3. Coen, M., Phillips, B. , Warshawsky, N., Weisman, L., Peters, S., and Finin, P.. Meeting the Computational Needs of Intelligent Environments: The Metaglue System. In 1st International Workshop on Managing Interactions in Smart Environments (MANSE'99), pp.201--212. Dublin, Ireland, December 1999.
4. Digital Lecture Recording Service. University of Canberra, Australia. http://www.canberra.edu.au/icts/av/lecstream-students.html
5. Does Size Matter? Class Size and Student Achievement (ERP Research Watch, June 1999)
6. Dufresne, R., Gerace, W., Leonard, W., Mestre, J. and Wenk, L. Classtalk: A Classroom Communication System for Active Learning. Journal of Computing in Higher Education, 7, 3-47, 1996
7. Ehrmann S., Asking the Right Question: What Does Research Tell Us About Technology and Higher Learning? American Association of Higher Education
8. Faculty Use of SMART Classrooms at San Diego State University. Report for the Senate Instructional Technology Committee, 2000 http://id-www.ucsb.edu/detche/news/sdsu/jfrazee1.pdf
9. Fischler M. A., Bolles R. C.. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comm. of the ACM, Vol 24, pp 381-395, 1981
10. Flachsbart, J. , Franklin, D. , Hammond, K., Improving human computer interaction in a classroom environment using computer vision, Proceedings of the 5th international conference on Intelligent user interfaces, p.86-93, January 09-12, 2000, New Orleans, Louisiana, United States
11. Franklin, D., Budzik, J., Hammond, K.: Plan-based interfaces: keeping track of user tasks and acting to cooperate. IUI 2002: 79-86
12. Franklin, D., Hammond, K.: The intelligent classroom: providing competent assistance. Agents 2001: 161-168
13. Friedland, G. , Jantz, K. , Knipping, L. : Towards Automatized Studioless Audio Recording: A Smart Lecture Recorder, Technical Report B-04-14, Faculty of Computer Science, Freie Universität Berlin, September 2004.
14. Friedland, G., Knipping, L. , Schulte, J. , and E. T. a. E-chalk: A lecture recording system using the chalkboard metaphor. International Journal of Interactive Technology and Smart Education, 1(1), 2004.
15. Honey, M., McMillan Culp, K., Spielvogel, R. Critical Issue: Using Technology to Improve Student Achievement. North Central Regional Technology in Education Consortium, 1999
16. ilecture recording system, Murdock University, Perth, Australia: http://www.murdoch.edu.au/teach/ilecture/recordingtips.html
17. King, J. The Impact of Class Size on Instructional Strategies and the Use of Time in High School Mathematics and Science Courses (EEPA, Vol. 21, No. 2, Summer 1999)
18. Kirstein, C. and Muller, H. (1998): Interaction with a projection screen using a camera-tracked laser pointer, in Multimedia Modeling; MMM '98 Proceedings, p.191-192

19. Lucas B. and Kanade T. An Iterative Image Registration Technique with an Application to Stereo Vision. International Joint Conference on Artificial Intelligence, pages 674-679, 1981.
20. Oh D., Fox H., Van Kleek M. , Adler A., Gajos K., Morency L., and Darrell T.. Evaluating Look-to-Talk: A Gaze-Aware Interface in a Collaborative Environment. In Proceedings of CHI 2002. Minneapolis, MN, April 2002.
21. Peck, C. (2001): Useful Parameters for the Design of Laser Pointer Interaction Techniques, in Proceedings of CHI 2001, p.461-462
22. R.R. Hake, Interactive-engagement vs traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses, Am. J. Phys. 66, 64- 74, 1998
23. Scholz, J., Grigg, M., Prekop, P. and Burnett, M. (2003). Development of the software infrastructure for a ubiquitous computing environment - the DSTO iRoom. In Proc. Workshop on Wearable, Invisible, Context-Aware, Ambient, Pervasive and Ubiquitous Computing, Adelaide, Australia. Conferences in Research and Practice in Information Technology, 21. Johnson, C., Montague, P. and Steketee, C., Eds., ACS. 169-176.
24. Ushijima, K., Construction and practice of a new educational environment with lecture recording system. AINA 2004, 18th International Conference on Advanced Information Networking and Applications. Pp.12-15. 2004
25. Xiang, C., Balakrishnan, R., VisionWand: interaction techniques for large displays using a passive wand tracked in 3D, Proceedings of the 16th annual ACM symposium on User interface software and technology, p.173-182, November 02-05, 2003, Vancouver, Canada