

# project AVRiL

---

End-of-Quarter Engineering Report

Version 1.0

13<sup>th</sup> Dec 2006

**Advisors:**

Dr. Sohaib Khan

Dr. Umar Saif

**BSc Sproj '07 - Group 7:**

Ahmad Humayun (CE / 2007-02-0236)

Ozair Muazzam (CS / 2007-02-0162)

Tayyab Javed (CS / 2007-02-0130)

Yahya Cheema (CS / 2007-02-0370)

[avril.sproj.com](http://avril.sproj.com)

**Note:**

**This is version 1.0 of the Engineering Report.**

**Since we are following an iterative development process this document is subject to revision as we progress through the project.**

- Section 1: Abstract ..... 5**
- Section 2: Project Block Diagram and Description..... 6**
  - 2.1 Overview of modules / components..... 6
    - Layer 5: Hardware Level..... 6**
      - 1) *Cameras* ..... 6
        - (a) Wide-Angle Camera ..... 6
        - (b) Lecturer Tracking Camera ..... 7
        - (c) Audience Tracking Camera ..... 7
          - i. PTZ Stand..... 7
      - 2) *Microphone Array*..... 7
    - Layer 4: Raw Data Level ..... 8**
      - 3) *Video-Receiver* ..... 8
      - 4) *Camera Controller*..... 8
      - 5) *Audio Processing* ..... 9
    - Layer 3: Software Tracking Level ..... 9**
      - 6) *Slide Transition Detection (also part of Layer 4)* ..... 9
      - 7) *Physical Design Rules*..... 9
      - 8) *Lecturer Tracking*..... 9
        - (a) Motion Tracking ..... 10
        - (b) Background Subtraction ..... 10
      - 9) *Audience Tracking* ..... 10
    - Layer 2: Director Level..... 10**
      - 10) *Direction Mixing* ..... 10
        - (a) Rules Manager ..... 11
        - (b) Mixer..... 11
      - 11) *Encoder* ..... 11
    - Layer 1: Data Compilation Level..... 11**
      - 12) *Presentation Compiler* ..... 11
    - Layer 0: Presentation Level..... 11**
      - 13) *Presentation* ..... 11
  - 2.2 Overview to Diagrams..... 11
- Section 3: System Modules Hierarchy Tree..... 12**
- Section 4: Project Plan ..... 13**
  - 1) *Phase 1* ..... 13
  - 2) *Phase 2* ..... 13
  - 3) *Phase 3* ..... 13
  - 4) *Phase 4* ..... 13
  - 5) *Phase 5* ..... 13
  - 6) *Phase 6* ..... 14
- 4.2 Project Schedule Map ..... 14
- Section 5: Deliverables ..... 15**
  - 5.1 Lecturer Tracking Module interfaced with the required Equipment ..... 15
  - 5.2 Direction Mixing Module interfaced with the Lecturer Tracking Module..... 15
  - 5.3 Slide Transition Detector interfaced with the Direction Mixing Module ..... 15
  - 5.4 Presentation Module ..... 16
  - 5.5 Audience Tracking Module interfaced with required Equipment (Tentative) ..... 16

**Section 6: References..... 17**  
**Section 7: Terms Used ..... 17**  
**Section 8: AVRiL Project Block Diagram ..... 18**

## Section 1: Abstract

*The purpose of this section is to give an overview of the system being developed*

The aim of the project is to design a system that would automate the recording process for university lectures, using video cameras that act intelligently to focus on the instructor, the presentation, the dais or the class of students, whichever is more important at that time.

There is a dire need of cheap, deployable, video systems which are able to record an academic atmosphere un-aided. We aim to develop a system which doesn't require human intervention because of two reasons; employing a dedicated camera team doesn't allow a university to film every important lecture feasibly; two, the visual presence of someone guiding a camera might not allow an instructor to follow his own natural style of teaching.

AVRiL (Automated Video Recording of Lectures) aims to automatically record, using PTZ cameras, university lectures in order to assist distance learning programs. We are assuming that the system would be used inside auditoriums only, in which only one instructor is teaching at a time. We also assume that the instructor will be standing and in motion most of the time, teaching a group of seated students who face away from the camera. Our final aim is to produce a high quality presentation video which retains a feel of the classroom feel.

## Section 2: Project Block Diagram and Description

*The purpose of this section is to describe the interactions and structures of all the modules of the system*

This section describes the project model we have envisaged in the inception stage. It is written from an engineering stand-point and you might want to refer to the Design Specification (Section 2) for more details on the software components of the system.

### 2.1 Overview of modules / components

This section introduces the components of AVRiL while giving more attention to the engineering aspect of the project. It will describe in detail the roles and interconnections of both the hardware components and the engineering related software components in detail.

At the lowest level we have cameras that feed video to (hardware) digitizer cards. There is also a microphone array which passes sound streams to sound cards for audio processing. The videos are passed above to the Lecturer Tracking Module for detecting the position of the lecturer; and the sound streams are passed above to Audience Tracking Module. All processed video streams are passed to the Direction Module which selects appropriate streams and encodes data into a single video stream. The encoded video is sent to the Presentation Compiler Module, which compiles the video with all other presentation data like slides into a format which can be easily played back on the Presentation Module.

## Layer 5: Hardware Level

### 1) Cameras

The whole process of video capture involves a number of engineering components. The lower hierarchy of the project is pre-dominantly occupied by engineering hardware components. If we observe the data flow of the system envisaged, one can draw a conclusion not only for the Video Capturing modules, but also for the rest of the system, that at the lower level there are engineering systems involving hardware components which feed data to upper layers having software components. Furthermore in the case of Video Capture, there is a feedback system within some hardware and software components, as it will be explained below.

Basically this component just generates raw video and passes it to the layers above to video digitizers. This level also contains the Pan Tilt Zoom (PTZ) Equipment for controlling the Field of View (FOV) of the cameras installed. At the present moment, the whole system is built using only three cameras, but the system is being evolved in a modular fashion after which adding more cameras will just be a matter of adding duplicated modules.

#### *(a) Wide-Angle Camera*

This camera will record video streams of the lecture hall in a fashion to capture the whole dais. This camera will be static (i.e. it will be stationary) so that the system will continuously have a video stream having a complete view of the dais at all times. The

camera might also be initially used to create mosaics of the lecture hall, which will be later used in Background Subtraction in the Lecturer Tracking Module. This camera, like all others we are going to use, gives a raw analog output which has to be fed into a digitizer to convert the output to digital frames.

### ***(b) Lecturer Tracking Camera***

This camera will be installed directly below the Wide-Angle Camera in order to ease the job of both the Camera Controller module and the Physical Design Rules module. The reason for this is that the Lecturer Tracking Camera and the Wide Angle Camera should have focal points which are close to each other, which makes the job easier of modules easier if they are referring to both the video streams of the respective cameras. If all modules work well, this camera should provide a video stream of the tracked lecturer on the dais. This camera is attached to a PTZ stand which aids the control of its motion by software layers above. For proper working of this camera two things are to be insured; that there are no occlusions between this camera and the lecturer; there are no other dynamic objects (in the form of other lecturers or even in the form of, let's say, moving machine parts) on the dais (refer to Requirement Specification, Section 2.4).

### ***(c) Audience Tracking Camera***

This camera is installed facing the audience; in a location where the entire lecture hall can be viewed e.g. directly above the projection screen. It basically works in conjunction with the Audience Tracking module to roughly give video streams of locations where sounds from the lecture hall could be possibly originating from. For proper working of the Audience Tracking Camera it is important to constrain the audience to ask questions one at a time, which also make pedagogical sense (refer to Requirement Specification, Section 2.4). This camera is too attached to a PTZ stand which controls its FOV. Secondly, as mentioned above, this camera too generates analog output which has to be converted by a digitizer.

#### **i. PTZ Stand**

The PTZ Stand is attached to the Audience Tracking Camera and the Lecturer Tracking Camera. The cameras we are incorporating in our system come bundled with the PTZ stands. These PTZ stands can be interfaced with a computer using an RS-232 serial port, through which commands can be sent to change the orientation of the camera itself and also adjust its zoom level to bring the camera to the required FOV. The PTZ Stands are handled by the Camera Controller Module.

## **2) Microphone Array**

This component is built using a set of microphones facing the audience. The purpose of this module is to detect to a vague extent the location from which sound is originating. It will be used in situations where the audience asks questions from the lecturer and the system tries to estimate (vaguely) where the person, questioning, is sitting. We envisage using multiple

standard microphones placed at strategic locations which can be used to for audio localization. A simple everyday example can be the stereo microphones installed on some video cameras; you could use sound streams from the microphones and tell by the intensity of the sound from the respective microphones to estimate if either the sound is originating from the right or left.

Each microphone will generate its own audio signal which will travel through a standard audio cable and will be input to a sound card. The sound card's output will be handled by the Audio Processing Module, which is directly above in the hierarchy.

## **Layer 4: Raw Data Level**

### **3) Video-Receiver**

As the digitizer cards receive raw analog video signals from the cameras, they convert the signal to digital video streams for processing. These digital video streams in the form of frames are processed by the Video Receiver. At the initial look, one might feel that there is no need for the Video Receiver Module and frames could be directly passed to the Lecturer Tracking Module or the Audience Tracking Module. But the Video Receiver Module is essential for two things: for bringing all frames of video to set common parameters e.g. to bring all frames to a standard brightness level (since different FOV could have different brightness levels); for reducing the frame rate for the Lecturer Tracking Module because it might not be required by the latter to process all frames so the Video Receiver Module will drop some amount of frames and pass the rest to the concerned Tracking Module.

It is important to note the there will be as many Video Receiver Modules as the number of digitizer cards. It is even possible that different instances of the Video Receive Module are run physically on different machines, each having its own digitizer card and its own Tracking Module.

### **4) Camera Controller**

The Camera Controller Module either collaborates with an Audience Tracking Module or a Lecturer Tracking Module to control the PTZ parameters of an Audience Tracking Camera or a Lecturer Tracking Camera (respectively). The Camera Controller takes input from the respective Tracking Module in the form of parameters telling the amounts by which a camera has to be rotated or zoomed. The Module in turn communicates with the serial port to send instructions to the PTZ Stand of the Camera.

If one looks closely, a given camera, the attached Video Receiver Module, its Camera Controller Module, and the Tracking Module (Audience or Lecturer) form a loop back system. The camera passes raw video to the digitizer, whose output is handled by the Video Receiver Module. The frames resulting from the latter are consumed by the specific Tracking Module (e.g. the Audience Tracking Module will be used if we are concerned with an Audience Tracking Camera) for processing. The Tracking Module decides that has the object (being tracked) moved in a fashion which requires the camera to move too. If so, the Tracking Module instructs the Camera Controller Module to make adjustments to the pan, tilt or zoom

of the camera by a given amount. After which the Camera Controller communicates on the serial port and send instructions accordingly to the PTZ stand.

## **5) Audio Processing**

The Audio Processing Module is analogous to the Microphone Array as the Video Receiver Module is to the Cameras. This Module gets its data from the sound cards connected to the microphone array. It does some “preparatory work” on the audio streams and then passes it to the Lecturer Tracking Module. The nature of the preparatory work is of the same nature as that of the Video Receiver, e.g. that if the Audience Tracking Module wants sound streams in a lower bit format than that produced by the Sound Card itself, it will be the onus of this Module to produce audio streams in the required bit rate. It might also be the responsibility of this Module to produce certain processed sounds from certain sections of the stream which might be more useful or more easily processed by the algorithms implemented in the Tracking Module.

## **Layer 3: Software Tracking Level**

### **6) Slide Transition Detection (also part of Layer 4)**

Detects, stores and communicates (to the Direction Mixing Module) the times of slide transitions and the snapshots of the slides themselves. (Refer to the Design Specification Section 2.1.4)

### **7) Physical Design Rules**

This Module is helpful to all the Tracking Modules and the Camera Controller because it contains all the information of the locations and FOV of all the cameras with respect to their physical location inside the lecture hall. All the decisions taken for deciding the parameters by which to move cameras are taken after referring to this Module. (Refer to the Design Specification Section 2.1.1.d)

### **8) Lecturer Tracking**

This Module is being explained in detail in this Engineering Report because we feel that the processes (mostly related to image processing) involved in it are as much a part of the field of engineering as they are to the field of software. The job of this Module is to (1) track the lecturer and her / his movements in the video stream (2) instruct the Camera Controller Module to move the camera accordingly (while referring to the Rules Manager Module) (3) and pass the video stream to the Mixer of the Direction Mixing Module. There are two main section of this Module: the Motion Tracking Module and the Background Subtraction Module. Both these modules work together to give a more accurate location of the Lecturer. (Please also refer to the Design Specification Section 2.1.2)

***(a) Motion Tracking***

The Motion Tracking Module basically works on the concept of optical flow of pixels. If in two given frames the camera remains static, the algorithm in this module will try to judge which pixel has moved in which direction roughly. The big picture that results from this Module is that which object in the first frame has possibly (and vaguely) moved to which portion in the next frame. This Module is helpful in estimating the movements of the lecturer.

***(b) Background Subtraction***

Before this Module can work effectively the Background Subtraction Module will build Mosaics of the complete FOV of the Lecturer Tracking Camera which basically results in a detailed background image of the whole dais (refer to Design Specification Section 2.1.2.a.i). This Module basically works on the principles of image processing that if you have two images of exactly the same FOV, and you subtract a pixel from one picture from the pixel of the same location from the other, an occluding object should appear, if present, in its current location. Same will hold true for a lecturer in a frame standing in the lecture hall. This Module will receive the video frames and the parameters related to the positioning and the FOV of the camera. The job of the Module will be to output the most likely location of the lecturer. The whole Lecturer Tracking Module will then decide (after looking at the results from either or both the sub-modules) if any movement is required from the camera.

**9) Audience Tracking**

The job of this Module is to decide roughly from where the sound is originating in the case when someone asks a question from the audience. The tasks of the Module are (1) to locate the sound from the audience, (2) to instruct the camera to move in the direction to where the sound is coming from, (3) and to pass a video stream to the Mixer of the Direction Mixing Module, if necessary. Filtering out the sound of the lecturer and of echoes in the lecture hall are some of the major challenges in building this module. But this Module takes advantage of the presence of multiple audio streams coming from strategically placed microphones in the lecture hall, to judge if sounds are too close to the location of the lecturer and to concentrate only on those which are not. Once the rough location has been established and the camera controller has been instructed accordingly, a video stream can be sent to the Direction Mixing Module containing shots which hopefully lie close to the person asking the question.

**Layer 2: Director Level****10) Direction Mixing**

The Direction Mixing Module has two tasks to handle (1) to use good direction heuristics [1] [3] [5] to instruct Tracking Modules on Camera motions (2) to use the same heuristics to decide on the best shot to choose from the input of the three video streams. (refer to the Design Specification Section 2.1.5)

**(a) Rules Manager**

The task of the Rules Manager is to basically collect useful information (like changes in slides or question raised from the audience) and to instruct other modules on the given direction rules to create better video streams. The Module can be compared to the mind of a movie director (refer to the Design Specification Section 2.1.5.a).

**(b) Mixer**

The task of this Module is to choose the most appropriate video stream from the input set of three video streams. This Module can be compared to the actions of a movie director (refer to the Design Specification Section 2.1.5.b).

**11) Encoder**

The task of the Encoder Module is to encode the video stream chosen by the Direction Mixing Module into an appropriate video format (refer to the Design Specification Section 2.1.6).

**Layer 1: Data Compilation Level**

(refer to the Design Specification Section 2.1.7)

**12) Presentation Compiler**

The task of this Module is to compile the given video and the presentation slides in to a set format which can be easily played back by the Presentation Module.

**Layer 0: Presentation Level**

(refer to the Design Specification Section 2.1.7)

**13) Presentation**

The task of this Module is to read the output generated by the Presentation Compiler Module and to be able to replay it at any given time.

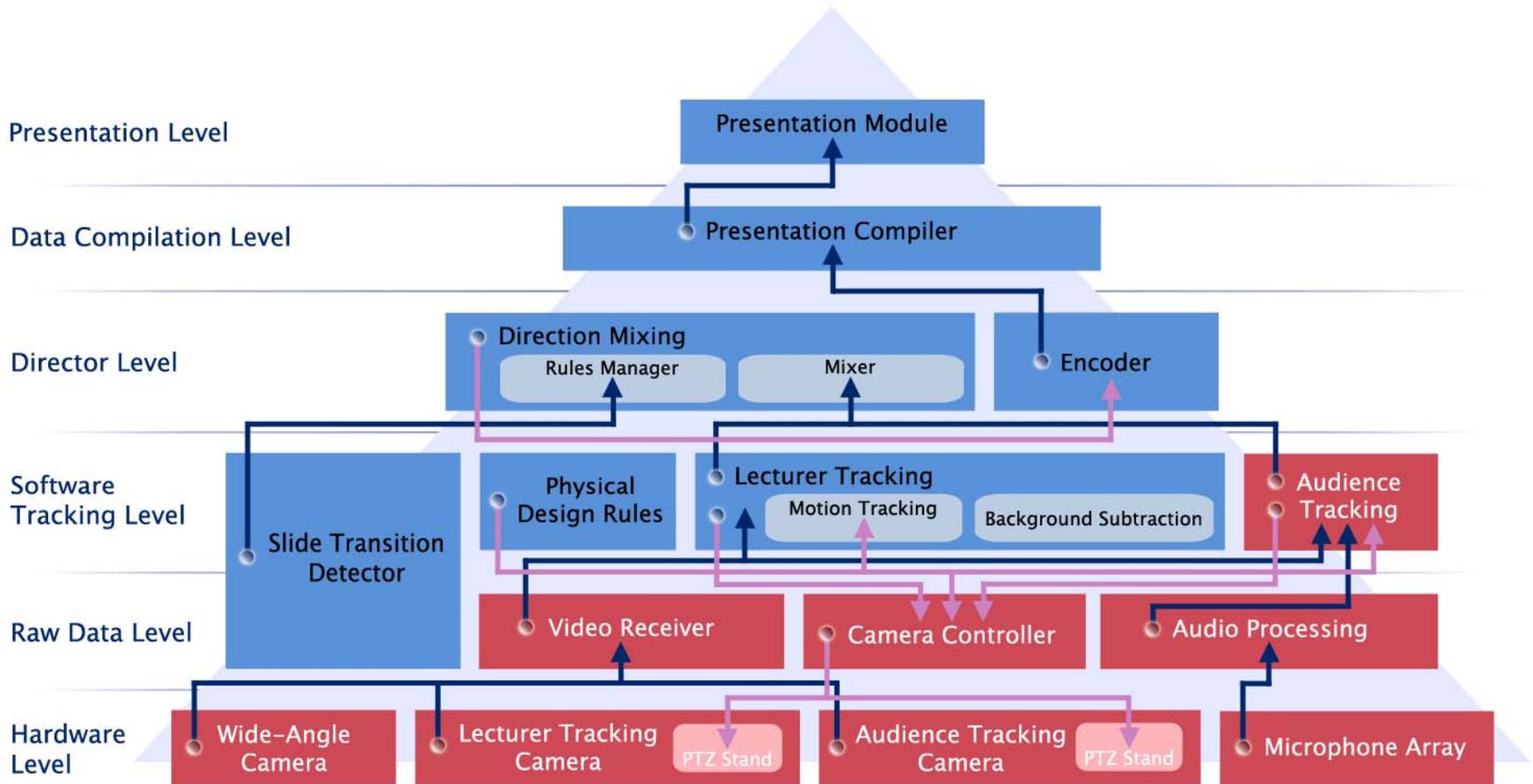
**2.2 Overview to Diagrams**

The first Diagram attached (Diagram 1) shows the hierarchy of the system according to the data flow in the system. Section 2 closely follows this diagram to explain the interfacing and objectives of each of the modules. The main Diagram attached (Diagram 2) shows actually how the system is integrated and what modules run where. Both Diagram 1 and Section 2 of the document closely derives from the ideas depicted in this diagram as it explains the inter-links and the functioning of all the hardware attached and all the modules present running across various machines.

Page 12: System Modules Hierarchy

Page 18: Project Block Diagram

### Section 3: System Modules Hierarchy Tree



**LEGEND**

**Blocks**

- Engineering related H/W & S/W Components
- Software Components

**Lines / Directions**

- Data transferred to Layers above
- Data transferred within or to Layers below

## Section 4: Project Plan

*The purpose of this section is to give a tentative schedule of how the system is going to be developed*

After our research we have divided the project work into 6 phases. These phases just act as logical blocks for the progress of the project. Although each of these phases might not have modules completely related to each other, they have modules adjusted in the order which makes sense in development. We have also looked at data and structural dependencies between modules.

All the blocks have been divided between the four group members, and at many places the modules have been handed to multiple members wherever we feel that the development would be more effective if the work is divided within a group of people. The purpose of keeping a “Cover Period” (January, Week 3) almost in between the development process is to let all members brush up the developed modules and to cover all unfinished modules. Each module in the phase is marked by a colored line (the color used is an attribute of the phase) denoting the length. The phases and the respective modules attached to them are:

### 1) Phase 1

1. Implementation of **Camera Controller Module** and interfacing with the **PTZ Stand**
2. **Motion Tracking Module** (*Lecturer Tracking Module*)
3. **Background Subtraction**
4. **Mosaic Building** (completion of *Lecturer Tracking Module*)

### 2) Phase 2

5. **Audio Processing Module**
6. **Video Receiver Module**

### 3) Phase 3

7. Implementation of DSP Algorithms in **Audience Tracking Module**
8. Integration of **Tracking Modules** with other modules
9. Building of **Microphone Array**

### 4) Phase 4

10. **Slide Transition Detector Module**
11. **Physical Design Rules Module** (integration with *Tracking Modules* and *Camera Controller*)
12. **Rules Manager Module** (and integration with *Slide Transition Module*)

### 5) Phase 5

13. **Mixer Module** (and integration with the *Tracking Modules*)
14. **Encoder Module**
15. **Direction Mixing Module** (integration with the *Encoder Module*)

6) Phase 6

16. Presentation Compiler Module

17. Presentation Module

4.2 Project Schedule Map

	Ahmad	Ozair	Tayyab	Yahya
<b>Dec</b> Week 1				
Week 2	③	①	③	②
Week 3				
Week 4		⑥		
<b>Jan</b> Week 1	④		④	
Week 2	⑤	⑤	⑥	⑥
Week 3	<b>Cover Period</b>			
Week 4	⑦	⑦	⑧	⑧
<b>Feb</b> Week 1				
Week 2	⑨	⑨	⑪	⑪
Week 3	⑩	⑩	⑫	⑫
Week 4		⑬	⑫	⑫
<b>Mar</b> Week 1	⑭		⑬	⑬
Week 2			⑮	⑮
Week 3	⑯	⑯	⑯	⑯
Week 4	<b>TESTING OF MODULES</b>			
<b>Apr</b> Week 1				
Week 2				
Week 3				
Week 4				

## Section 5: Deliverables

*The purpose of this section is describe the nature of the end-modules and equipments that is going to be subject to evaluation*

All the deliverables are in the form Modules given in Section 2. But as a basic objective we plan to deliver a prototype which can be used for recording of lectures with three cameras, a number of interfaced machines, a microphone array all running with minimum human intervention. We would like to emphasize at this point, that the system being developed will be a prototype which would demand further improvement and development rather than being end-product which is readily deployable. Secondly, as we see the project having great research perspective, we plan to deliver modules implemented after extensive research. A corollary would be that the algorithms implemented might have some research value in the field of automated video recording of lectures.

### **5.1 Lecturer Tracking Module interfaced with the required Equipment**

This deliverable includes the following equipment and modules:

- A. Wide Angle Camera (Section 2.1.1.a)
- B. Lecturer Tracking Camera with PTZ Stand (Section 2.1.1.b)
- C. Video Receiver Module (Section 2.1.3)
- D. Camera Controller Module (Section 2.1.4)
- E. Lecturer Tracking Module (Section 2.1.8)

As a stand-alone deliverable, it should be able to track a lecturer on a dais with the PTZ tracking camera and produce video with good tracking fidelity.

### **5.2 Direction Mixing Module interfaced with the Lecturer Tracking Module**

This deliverable also includes the Encoder Module since it is directly related to it as the output of the Direction Mixing Module is used as input by the Encoder. The deliverable includes:

- A. Direction Mixing Module (Section 2.1.10)
- B. Physical Design Rules Module (Section 2.1.7)
- C. Encoder Module (Section 2.1.11)

As a stand-alone deliverable, if given slide transition times and three video streams: one of the tracked lecturer; one of the wide-angle camera; and the audience video stream, it should be able to decide which video stream should be directed to the encoder (and the rest will be discarded). The decisions this deliverable makes should be based on “good direction heuristics”.

### **5.3 Slide Transition Detector interfaced with the Direction Mixing Module**

This deliverable just includes:

- A. Slide Transition Detector Module (Section 2.1.6)

This deliverable should be able to interact with the Direction Mixing Module in order to give both timestamps of slide transitions and snapshots of the slides themselves.

### **5.4 Presentation Module**

This deliverable includes two modules:

- A. Presentation Compiler Module (Section 2.1.12)
- B. Presentation Module (Section 2.1.13)

This deliverable as a stand-alone, if given slide snapshots, slide transition times and an encoded video, the deliverable should be able to compile it in some native or proprietary format. This format should be easily re-playable by the deliverable into its native GUI, which should give the look and feel of the lecture hall environment

### **5.5 Audience Tracking Module interfaced with required Equipment (Tentative)**

This deliverable involves three modules

- A. Microphone Array (Section 2.1.2)
- B. Audio Processing Module (Section 2.1.5)
- C. Audience Tracking Module (Section 2.1.9)

This deliverable as a standalone should be able to complete two objectives: it should be able to decide if someone is asking a question from the audience or not at a given time; if so it should be able to provide a video stream which closely points to the person asking the question. *This deliverable is made tentative because it has concepts of DSP which are quite involved inside the field, and the concepts used in implementing such a Modules are still being researched upon.*

## Section 6: References

- [1] Bianchi, Michael (Foveal Systems), Automatic Video Production of Lectures Using an Intelligent and Aware Environment; 2004 ACM
- [2] Cruz, Gil (Bellcore), et al., Capturing and Playing Multimedia Events with STREAMS; 1994 ACM
- [3] He, Lei-wei (Microsoft Research), et al., The Virtual Cinematographer: A Paradigm for Automatic Real-Time Camera Control and Directing; 1996 ACM
- [4] Kameda, K. (Kyoto University), et al., CARMUL: Concurrent Automatic Recording for Multimedia Lecture; 2003 IEEE
- [5] Liu, Qiong (Microsoft Research), et al., Automating Camera Management for Lecture Room Environments; 2001 SIGCHI
- [6] Mukhopadhyay, Sugata (Cornell University), et al., Passive Capture and Structuring of Lectures; 1999 ACM
- [7] Rui, Yong (Microsoft Research), et al., Videography for Telepresentations; 2003 SIGCHI
- [8] Yu, Bin (University of Illinois at Urbana-Champaign), et al., Video Summarization Based on User Log Enhanced Link Analysis; 2003 ACM

## Section 7: Terms Used

- PTZ** – Pan Tilt Zoom.
- FOV** – Field of View
- Digitizer** – Also called a Frame Grabber. Converts analog input of cameras to digital frames.
- Static** – Cameras that have a constant FOV

# Section 8: AVRIL Project Block Diagram

