

# CSCI 576 Multimedia Project

**Instructor:** Parag Havaldar

**Demo date:** Wednesday May 6<sup>th</sup> & Thu May 7<sup>th</sup> 2020, time slots TBD

The course project is meant to give you an in depth understanding of some of the areas in multimedia technology. Since this is a broad field, there can be a variety interesting projects that can be done depending on your interests which can also extend to related and complementary topics that are taught in class.

I have often found that a large project can be successfully accomplished via collaboration. Additionally, working together to design and integrate code can be a rewarding exercise and you will frequently need to work in teams when you set out to work in the industry. Accordingly, please form groups of **exactly two** students. We have started a discussion board to help you make groups, where you may post your preferred language of implementation, availability etc. to choose your partner.

This time I want to suggest a topic that is at the heart of trying to quickly browse, interact with and understand AV content. These days there is no dearth of prolific media content that people create for themselves (video of family, friends, personal events etc, or images of vacations, weddings etc.) or available commercially such as movies, documentaries, sports videos. The assumed way to navigate and browse media content is to browse to a location (folder, http: site) and a list of media files are listed for you – either via a file list, thumbnails, or http based thumbnail pointers. The normal metaphor people follow to view them is by clicking on them and load in a photobrowser or view them in a video player. They can then “linearly” progress (from frame to frame) by playing the video or move from photo to photo. For video, you may also use trick play (fast forward) for watching or randomly access the stream. While all these get the job of “viewing” done, it is very time consuming and does not provide a “wholistic” view of all the content in a location.

You need a way to quickly visualize what the media content is about in the folder or at a location and then make a more “*visually*” *informed decision to explore it*, quickly see relevant details before you decide to invest your time into watching it in more detail. The question then is what kinds of viewing or exploring metaphors can you come up with that allow this. This is a topic of much research and while there are no correct or incorrect answers, surely some might work better than others. Please see description that follows.

***Also – for this project, you are welcome to use any 3<sup>rd</sup> party libraries and can choose your own desirous programming environment/language.***

## Creating and browsing media using “synopsis” images

In this project you will implement a *media synopsis* algorithm that produces a “*synopsis image*” summarizing media content. Synopsis is an ancient Greek word that means “general view” or a “summary view”. As input you will take a path to a folder that contains various visual media elements – video (with audio) and images. Normally all these media elements at one location (folder, http address) should be contextual similar eg video and images of a personal event such as vacation, graduation, wedding. The synopsis image should give you a good flavor and representation of all the media content. Furthermore, you are also tasked to create an interactive player that can interact with this “synopsis” image so that when you click on some location of the synopsis image, this will trigger the corresponding video to play (with audio synchronized) from that contextual location or show the corresponding image.

### INPUTS AND OUTPUTS

Input to your Process: A folder which contains

1. A video file in CIF format (352x288) and a corresponding audio file in WAV format, synced to video.
2. Images in a CIF format (352x288)

More details regarding frame rate and length of video will be given in data description file. You may assume these parameters will be the same for all files used in this project.

Expected Output:

1. A synopsis image (or a hierarchy of images) for the media elements in the input folder. This image will a visual representation of all the “important” parts of the media elements. This can be an offline process (lets call is *CreateSynopsisImage* )
2. You are also required to design and implement an interface that loads the synopsis image and allows to explore the visual content. You should be able click on a location in any “interesting” area in your synopsis image which will result in playing a video (with audio sync) from that contextual location. Or if the “interesting” area came from an image, then show an image(s). Step 1 should also create appropriate pointers/data structures to help the interface index into browsing the A/V content

Example Invocations:

*CreateSynopsisImage.exe locationToFolder* – generates *MySynopsis.rgb*

*ExploreSynopsis.exe MySynopsis.rgb*

After reading through the project, you will understand that no synopsis image is incorrect! But there are informative and descriptive ways to go about creating the image to make is a useful representation of the content that needs to be understood or explored.

## IMPLEMENTATION IDEAS

### *Creating the synopsis image.*

Given a folder, make a list of all video and image elements in that folder. For each video.rgb there will be a corresponding video.wav audio file. The synopsis image can be generated by extracting content from various frames and images and putting them together. One way is to extract “important” frames which may be used to create your synopsis image. The easiest (but not so descriptive) way to extract key frames might be to select one frame from each  $n$  frame (e.g.,  $n = 100$  then select the first of every 100 frames). You may then decide how to create a synopsis image putting parts of these frames together – eg assuming the interesting part is always in the center, then creating a synopsis image by putting together side by side the central part of these frames. This might be good enough to get a first start at understanding the working of this project but for a synopsis purpose, this might not be a good algorithm because every  $n$ th frame may not correctly depict interesting frames in the video.

There are better statistical and heuristic ways to pick key frames instead of every  $n$ th frame. One way to do this is to sample one frame in each scene change, or cuts because all cuts are logically continuous. A more statistical representation may be to choose key frames where the motion information content in the video might be high – such as objects are in motion, or the audio levels are high. This will need you to analyze video frames using motion prediction taught in class or analyze samples of audio for sound levels. You are free to use any frame extraction techniques and smarter techniques will generally lead to a more descriptive summary image. Please see our evaluation policies below. An example synopsis image result of this step might take a form as shown below, where parts of the selected frames are concatenated to create one image that shows you visual aspects of what is in the media folder.



You will also need to create a metadata file with your synopsis image. This should have logical pointers from areas (or even) pixels of your image to the actual video/image content on your folder. This is necessary for exploration so that you can interact with your synopsis image. Eg – click on an area in the image should enable you have enough information so that you can play/show the right content from the right time (in case of video). We leave the design/fileformat of this representation to each group, you may

choose a structure that makes sense to your design. *The generate summary image as a “visual” guide to understanding and exploring what the media elements are about.*

The above simpler synopsis image definitely works but also generates a long and less descriptive synopsis. A more involved though better way could be - after extracting all the important frames, you could create a summary image by selecting interesting sub areas in all frames and putting them together. For this process, you could use a variety of techniques for this (eg context based sub sampling, face detection, seam carving etc.) – these are limitless. You can also treat this problem as an optimization problem, please refer to [2] [3] for more details. Again, please see our evaluation policies below to determine which techniques to use.



### ***Exploring the synopsis image.***

Create a user interface that loads the synopsis image and displays it. The interface should also have an area to display the video/image. You should also have interactive interfaces (eg button widgets) to - *play*, *pause* and *stop* in case of a video. An example interface is shown below. This should explain the expectation of explorative interactivity, but you are welcome to create your own layout and design. The beauty of the interface is not important but having the ability to explore as described here is important.

In this example interface, the synopsis image is displayed at the bottom for users to explore the media elements. Once the users click on a position of the synopsis image, your interface should locate the place in the original video or image and display it in the display area. If video, it should display the corresponding frame. You may choose to play the video automatically from the corresponding frame or have the user to hit play, pause, stop to interact with it. If it clicked area points to an image, there is no playback necessary. and start to play from there.

DISPLAY HERE  
VIDEO (at corresponding frame)  
OR IMAGE

PLAY

PAUSE

STOP



### ***PROJECT EVALUATION***

The generation of your synopsis image should normally take a longer time and will necessarily be done offline. When your *ExploreSynopsis* program is launched, it should load the synopsis image file along with the metadata pointers and provide a user interface that allows you to explore the synopsis image interactively. An example of such an interface is shown above. Clicking on areas of the synopsis image should show the corresponding video being playing (with audio synchronized) at the corresponding frame so show the corresponding image.

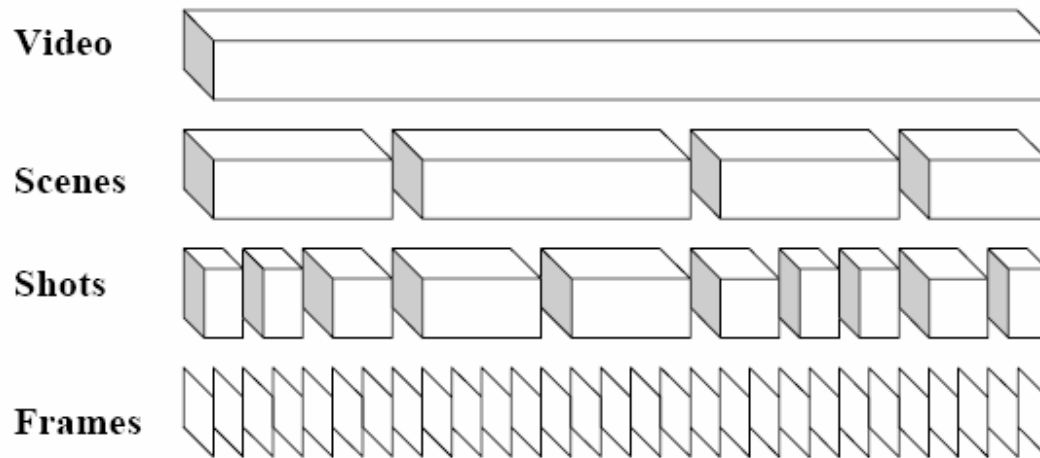
Your project will be evaluated on

1. The quality, completeness and continuity of your summary image
2. The correctness of how you handle the interactivity – whenever you click on a position in the summary image, it should start playing the video at that location, or show the corresponding correct image.
3. The correctness of audio video synchronization in case of a video.
4. The ability to answer questions related to your implementation and the theory around it.

### **REFERENCES**

- [1] [http://www.cs.princeton.edu/gfx/pubs/Barnes\\_2010\\_VTW/index.php](http://www.cs.princeton.edu/gfx/pubs/Barnes_2010_VTW/index.php)
- [2] <http://www.wisdom.weizmann.ac.il/~vision/VisualSummary.html>
- [3] [http://www.cs.princeton.edu/gfx/pubs/Barnes\\_2009\\_PAR/index.php](http://www.cs.princeton.edu/gfx/pubs/Barnes_2009_PAR/index.php)

It might help to understand the anatomy of a video:



- **Frame:** a single still image from a video, eg NTSC - 30 frames/second, film – 24 frames/second
- **Shot:** sequence of frames recorded in a single camera operation
- **Sequence or Scenes:** collection of shots forming a semantic unit which conceptually may be shot at a single time and place