(Semi) Automatic Infinite Entertainment System

Multimedia Computing Project

Marta Carlos
47592
Faculdade de Ciências e Tecnologia, NOVA
m.carlos@campus.fct.unl.pt

Ângelo Duarte
47427
Faculdade de Ciências e Tecnologia, NOVA
amv.duarte@campus.fct.unl.pt

1. INTRODUCTION

With the evolution of technology and the easy access to it, more and more entertainment systems are being developed and put into the market. With this specific area in mind, this project aims to develop a system capable of showing videos based on what it can detect with a camera.

This project is being developed by the means of the Multimedia Computing class and has as its' main goal the detection of faces and display of videos (similar between each other) related to the environment of the detected face. More specifically, the videos will depend on the objects, colors, audio or even rhythms being captured on camera.

Besides the continuous playing of videos, the system also offers an administrator interface. With this interface, it will be possible to specify the videos and when these should be displayed.

2. STATE OF ART

This project arises on the need to make video suggestions based on the user's reactions and environments.

One of the crucial features of this system is face recognition. To understand the environment and make a better user experience, the system needs to understand the amount of users and their movement, something that was already studied and created by [2].

Beyond face detection lies analysing the context with it and thus playing videos accordingly. This context is dependent on different particularities, one being the users themselves, like how interested they seem in the videos. A framework capable of extracting such features like face pose, eye gaze and blinking from video stream was created by [6]. Face-tracking has also been used in combination with games [4].

The administrator plays a big role on the functionality of the system: if the administrator doesn't comprehend how the system works, then no videos will be displayed to the target users.

One of the most important tasks that the administrator will have is assigning videos to events. This will be a semi automatic process where the administrator will count on a video similarity algorithm.

In the area of detecting video similarity, [5] developed a tool capable of performing content-based similarity search using visual and audio features, combined with traditional search methods, over continuous archived video data. Image histograms and spatial-temporal features of videos were also used by [7], that computed a compact video signature and these video signatures are further used for efficient similarity search.

To detect possible video similarities and also help the debugging process, the metadata for each video will be processed once the video is loaded. This implies the reading of different video characteristics.

Another important aspect of the system is the interface. This must have a good visual display of all the videos loaded in the system: there are already many different studied ways on finding the best way to display a big number of videos in one go, depending on the system's main goal.

Considering how the typical one frame thumbnail is often not the most appropriate representation of a video's highlight/importance, a different approach inspired by storyboards was proposed by [1]. With the presented method, one was able to click and drag on different elements of the storyboard scrubs through time.

On the other hand, [3] proposes a video summary as different sized key-frames being displayed like a comic book. This is extremely interesting for the context of this project since one of its' goals is the display of the most important frames of each video (in the playlist) when hovered over.

3. SPECIFICATION

In the initial stages of this project, an initial base for the interface was designed. This design is mainly divided into two modes:

- Administrator mode
- User mode

In this section we shall describe their purposes and the reasoning behind the choices made for each interface specification.

3.1 User mode

The interface for this mode is as simple as it gets: the system retrieves data from the video camera and system events

ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

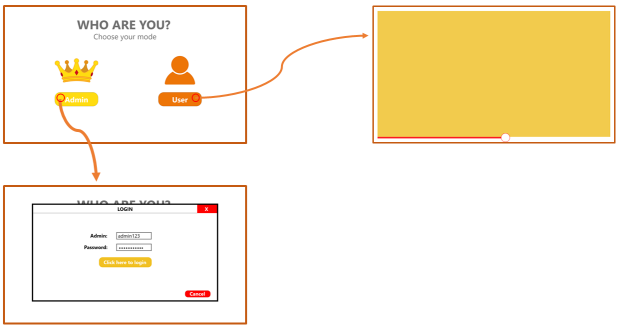

Figure 1: Storyboard of the login page

run it through the algorithm inferring information. Information like the number of faces and user's interest based on face motion and the environment.

With this information and the administrator's defined settings for each video, videos related to the interpreted data are displayed on the screen. The only exception is when no face is detected by the system's algorithm, since this will result in a group of specified videos being played.

3.2 Administrator mode

In this interface is where one will be able to prepare the user experience by loading the videos and configuring the different settings behind the developed system. The home page for this interface will be a typical *playlist*, with a list of thumbnails representing the different videos.

The system divides videos in two ways:

- By events
- By tags

On the left of the screen, there will always be a sidebar with the list of all possible events and the buttons with three possible actions: add an event, load a video and debug mode. This sidebar will enable the administrator the option to view all the videos assigned to a certain event. From there onwards, the playlist will be separated by tags (visually separated by lines).

Besides that, there are the previously mentioned buttons. The **Add event button** (or *Ctrl + E*) is a pop up to add an event and define the event's details. These details need to be inferred by the algorithm so that the event videos will be played in the User mode, as shown in the Figure 2.

The **Load video button** (or *Ctrl + L*) starts with the selection of a file (to load to the system), followed by a pop up where the tags are defined and one m a pop up showing all the video information (Tags, Metadata, Events), as shown in the Figure 3. The events are automatically filled based on a video similarity analysis, but can be altered.

Debug mode button (or *Ctrl + D*) shows the user their webcam and all the inferred data, enabling the admin to understand in each situation what event event will be played.

The administrator can choose to watch a video by simply hovering the mouse over the thumbnail and clicking on it. When the mouse is hovering on said thumbnail, the 5 different most relevant frames from the video will be played interchangeably in order to help identifying the video's content.

The administrator will be able to create events to assign to videos and establish the parameters associated with the newly created event. Loading videos is pretty simple: the user must choose the video they pretend to upload and tag it accordingly. The tags associated will help the algorithm sort the video to the events it will be associated with.

To support the administrator, we offer two ways one can use to debug the system:

- Easy access to the metadata of each video
- Interface that displays the camera feed and what the algorithm is inferring from the camera

These options give the administrator the power to fully comprehend how the system is working behind the scenes and to quickly identify possible problems.

4. REFERENCES

[1] D. B. Goldman, B. Curless, D. Salesin, and S. M. Seitz. Schematic storyboarding for video visualization and editing. *ACM Trans. Graph.*, 25(3):862–871, July 2006.

[2] P. J. Phillips. Support vector machines applied to face recognition. In *Proceedings of the 1998 Conference on Advances in Neural Information Processing Systems II*, pages 803–809, Cambridge, MA, USA, 1999. MIT Press.

[3] S. Uchihashi, J. Foote, A. Girgensohn, and J. Boreczky. Video manga: Generating semantically meaningful video summaries. In *Proceedings of the Seventh ACM International Conference on Multimedia (Part 1)*, MULTIMEDIA '99, pages 383–392, New York, NY, USA, 1999. ACM.

[4] S. Wang, X. Xiong, Y. Xu, C. Wang, W. Zhang, X. Dai, and D. Zhang. Face-tracking as an augmented input in video games: Enhancing presence, role-playing and control. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 1097–1106, New York, NY, USA, 2006. ACM.

[5] Z. Wang, M. D. Hoffman, P. R. Cook, and K. Li. Vferret: Content-based similarity search tool for continuous archived video. In *Proceedings of the 3rd ACM Workshop on Continuous Archival and Retrival of Personal Experences*, CARPE '06, pages 19–26, New York, NY, USA, 2006. ACM.

[6] Z. Wang, J. Yan, and H. Aghajan. A framework of personal assistant for computer users by analyzing video stream. In *Proceedings of the 4th Workshop on Eye Gaze in Intelligent Human Machine Interaction*, Gaze-In '12, pages 14:1–14:3, New York, NY, USA, 2012. ACM.

[7] C. Zheng and Z. Ming. An efficient video similarity search strategy for video-on-demand systems. In *2009 2nd IEEE International Conference on Broadband Network Multimedia Technology*, pages 174–178, Oct 2009.

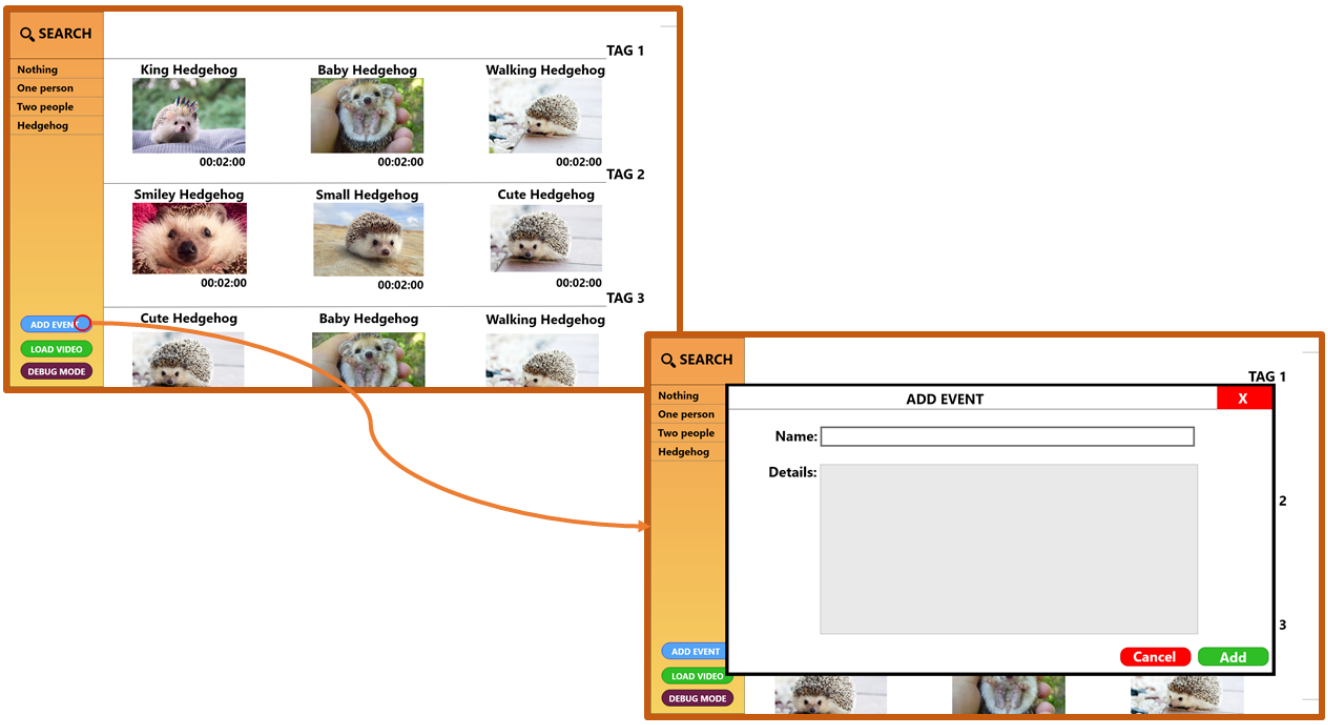

Figure 2: Storyboard of creating a new event

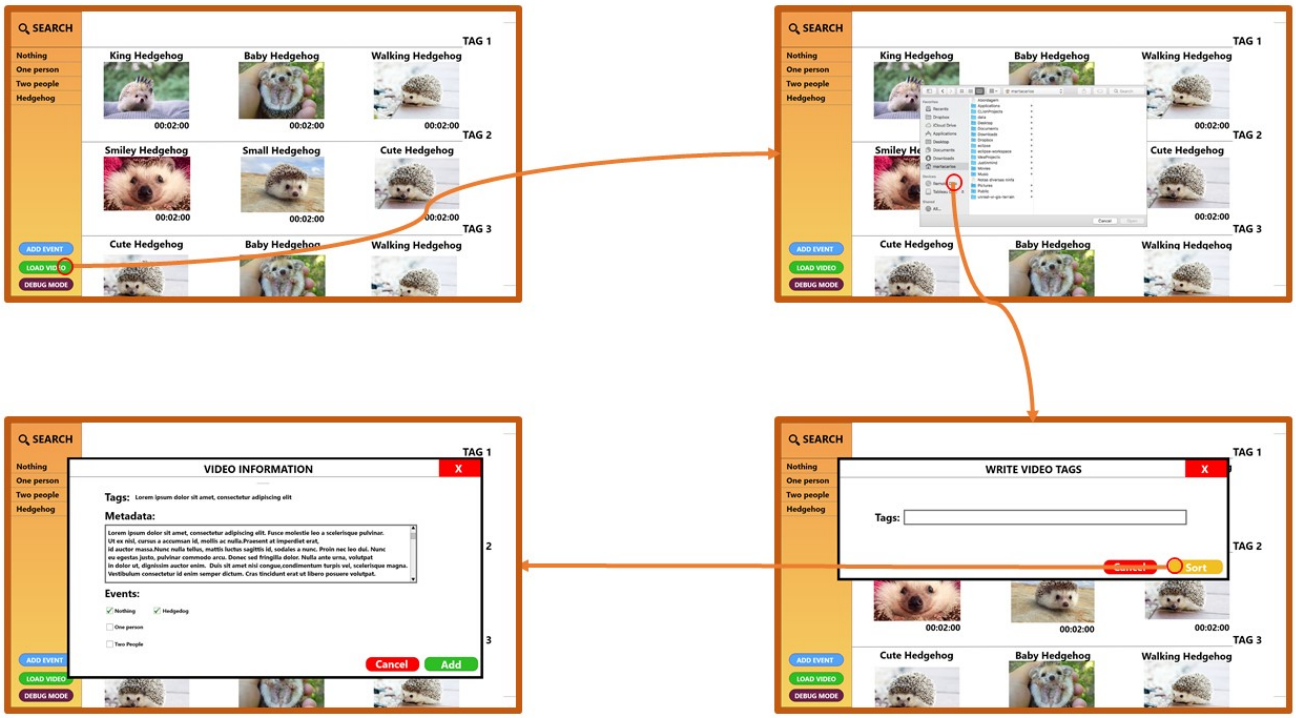

Figure 3: Storyboard about how to load a videos to the system

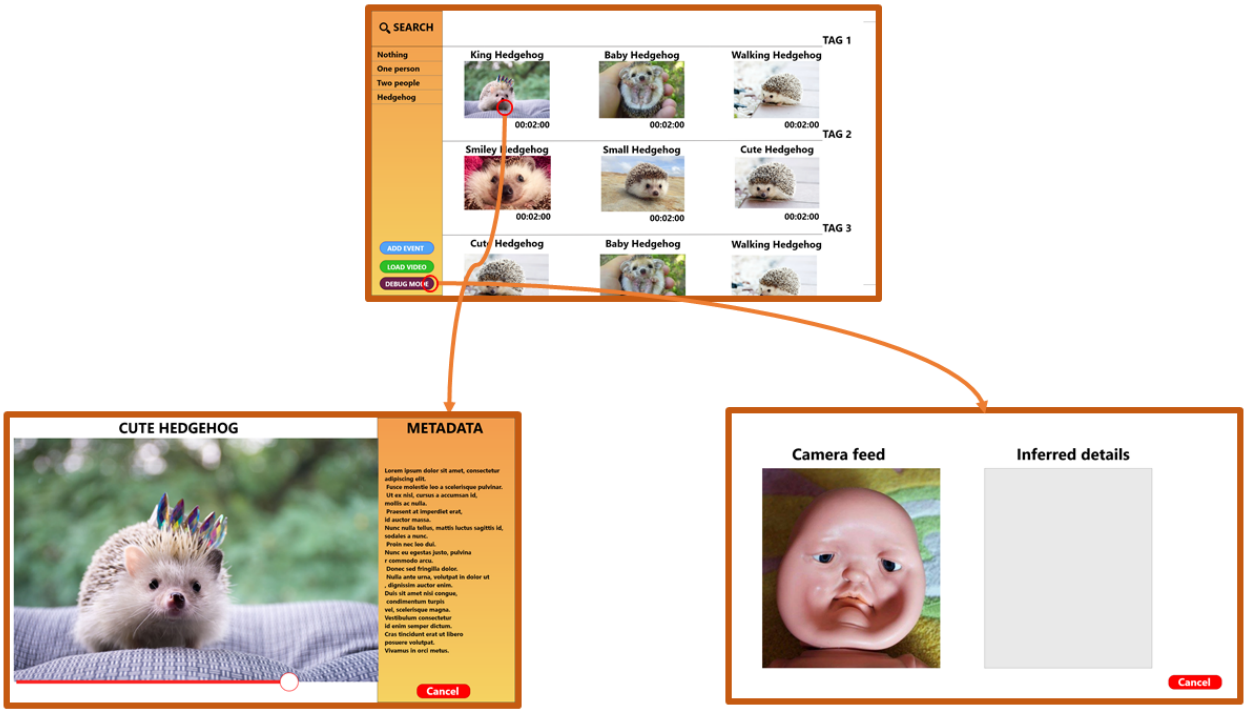

Figure 4: Storyboard about the debug possibilities