

Wall of Memories

Exploring the Claude Nicollier Video Archives

by

Giacomo Alliata

Master's Project final report presented for
the Master in Digital Humanities at EPFL

The logo of EPFL (École Polytechnique Fédérale de Lausanne) is displayed in a bold, red, sans-serif font. The letters are stylized with a blocky, geometric appearance.

Conducted at the Laboratory of Experimental Museology eM+
Under the supervision of Professor Sarah Kenderdine
From May 10th 2021 to September 10th 2021

Wall of Memories

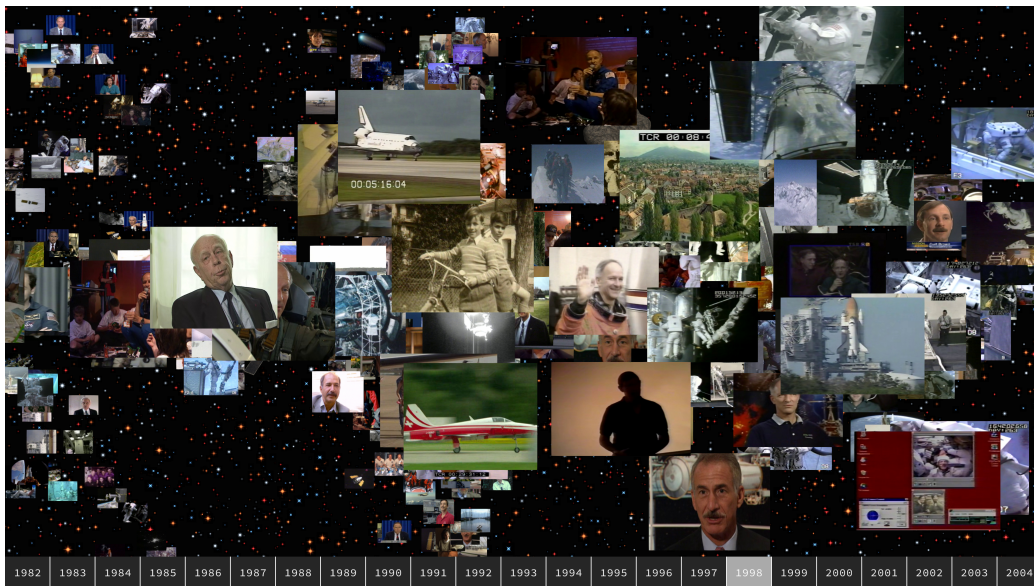
Exploring the Claude Nicollier Video Archives

by

Giacomo Alliata

Link to the video demo: <https://youtu.be/b4qYkm-Xumk>

KEYWORDS: Experimental museology, Interactive, Contemplative, Audiovisual collection, Claude Nicollier, Linear Navigator, Unity



Abstract

English

Wall of Memories is part of the *Cosmos Archeology* exhibition, to be held in Spring 2022 at EPFL Pavilions, Lausanne. The installation aims at exploring the Claude Nicollier Video Archives, a collection of audiovisual

content about the life and career of famous Swiss astronaut Claude Nicollier, digitized and curated by the Cultural Heritage and Innovation Center at EPFL.

A virtual wall of these videos is built in Unity, with a space-like atmosphere, using the Linear Navigator to navigate it. It is a 4K touch screen mounted on a 12m rail fixed to the wall, acting as a window through which visitors can explore this world of videos.

The contemplative nature of the installation invites visitors to experience this audiovisual collection either by letting it pick content randomly, following the Linear Navigator to the corresponding year, or by actively selecting by touch a thumbnail that catches the visitor's eye.

Rather than looking for a specific video or theme, visitors are encouraged to appreciate the whole collection and discover by chance content they can enjoy.

Français

Wall of Memories fait partie de l'exposition *Cosmos Archeology*, prévue pour le printemps 2022 à EPFL Pavilions, Lausanne. Cette oeuvre a pour but d'explorer les Archives Claude Nicollier, une collection de contenu audiovisuel traitant la vie et la carrière du reconnu astronaute suisse Claude Nicollier. Ces archives ont été digitalisées et organisées par le Centre d'Innovation dans les Patrimoines Culturels de l'EPFL.

Un mur virtuel de ces vidéos est généré dans le moteur de jeu Unity, dans une ambiance rappelant l'espace, en utilisant le Linear Navigator pour l'explorer. Ce système est composé d'un écran tactile 4K pouvant se déplacer le long d'un rail de 12m fixé au mur. L'écran fait ici office de fenêtre à travers laquelle les visiteurs peuvent explorer ce monde de vidéos.

La nature contemplative de l'oeuvre invite les utilisateurs à découvrir cette collection audiovisuelle en laissant le système choisir une vidéo au hasard, ou en sélectionnant une image qui suscite leur intérêt et curiosité.

Plutôt que de chercher une vidéo ou un thème en particulier, les visiteurs sont ainsi encouragés à plonger dans la collection toute entière, en découvrant par hasard du contenu qui peut les faire rêver.

Acknowledgments

First of all, I would like to thank Professor Sarah Kenderdine for giving me the opportunity to work at the Laboratory of Experimental Museology of EPFL, developing an installation for the exhibition *Cosmos Archeology*, to be held at EPFL Pavilions in 2022. Her supervision and comments on this work have helped creating an installation I can be proud of.

I would also like to thank the rest of the team at eM+, in particular Sylvain Cardin for his expertise in Unity and for following my work, giving me precious advises and technical help when needed.

A special mention goes to Alain Dufaux, Alain Chardonnens and their colleagues at the Cultural Heritage and Innovation Center at EPFL, for trusting me with the dataset of videos they have digitized and curated, and for answering my questions to better understand it.

Finally, I would like to offer my gratitude to Laurent Novac, for his supervision of the project and valuable inputs on the graphical and UX aspects of this installation.

Contents

1	Introduction	8
1.1	Overview of similar works	9
1.2	The exhibition	12
2	Presentation of the dataset	13
2.1	Digitization and curation work	14
2.2	Final dataset for this installation	15
2.3	Short data analysis of the final dataset	16
2.4	Video format and compression	18
2.5	Measurements with Unity's VideoPlayer	20
2.6	Generating the thumbnails	21
3	Description of the Experience	25
3.1	Intended audience	25
3.2	The serendipity concept	26
3.3	Become your own storyteller	27
3.4	The Linear Navigator	28
3.5	The VideoScreenElement	30
3.6	The virtual wall of videos	31
3.7	The Full Video mode	32
3.8	The background and sound ambiance	33
4	Technical implementation	34
4.1	Unity's framework	34
4.2	Handling the movement of the screen	35
4.3	Interacting with the application	36
4.4	Idle mode	37
4.5	The different elements in the scene	39

4.5.1	The LinearNavigator and the Camera	39
4.5.2	The wall of VideoScreenElements	40
4.5.3	Handling the movement of the VideoScreenElements .	42
4.5.4	The bottom timeline	42
4.5.5	Handling touch and mouse input	43
4.5.6	Generating the background	45
5	Further improvements	50
5.1	Updates to the dataset	50
5.2	Integrating VIRUP data	50
5.3	Finalising the installation for the exhibition	51
6	Conclusion	52
	Bibliography	54
A	Appendix	58
A.1	Compression results	58

List of Figures

1.1	<i>T_Visionarium</i> installation with visitors.	9
1.2	<i>Cloudbrowsing</i> installation by ZKM.	10
1.3	<i>mARChive</i> installation	11
1.4	<i>Campus Chronicles</i> installation by eM+.	12
2.1	Claude Nicollier holding an EPFL flag in space	13
2.2	Number of videos per year.	15
2.3	Principal video creators.	17
2.4	Distribution of video coverage time in minutes.	17
2.5	Distribution of the number of sequences per video.	18
2.6	Distribution of the Unity's VideoPlayer Prepare times.	21
2.7	Keyframes generation test	23
3.1	Views of the Linear Navigator device	29
3.2	Screenshot of the AccelnetProxy application interface.	30
3.3	Wall of VideoScreenElements in Unity	31
3.4	Full Video mode screenshot of the application.	32
3.5	View of the background with just a few videos.	33
4.1	Unity editor view	35
4.2	Visual prompt to interact	39
4.3	Plot of the γ vertical scaling factor	41
4.4	Zoom on the bottom timeline	43
4.5	Trail effect while swiping	44
4.6	Colored selection frame	45
4.7	Reference image for the star field	46
4.8	The four models of sprites used for the star field	47
4.9	Particle System star field in Unity	48

4.10 Satellites and asteroids models used	49
4.11 Satellites and asteroids placement	49
5.1 Timeline painted on top of <i>Campus Chronicles</i> installation . .	51

Chapter 1

Introduction

Datasets of audiovisual content are more and more common, with Youtube being one of the most important social media platform, making video sharing extremely simple. At the same time, broadcasting organisations have been producing and collecting audiovisual content for several decades now, resulting in the creation of large datasets of this kind. Famous examples are the digital archive of Radio Télévision Suisse [1], tracing back to the early 1930s and spanning a vast range of themes. Similarly, worth of mention is the Montreux Jazz Festival archive [2], documenting the history of this festival through videos of concerts and shows that have taken place in Montreux concert halls these last decades.

In this context, the Cultural Heritage and Innovation Center (CHC) have digitized in 2020 the Claude Nicollier Video Archives, a collection of videos related to the famous Swiss astronaut Claude Nicollier’s life and career.

The goal of this Master’s Project in Digital Humanities at EPFL is therefore to build an interactive installation using the Linear Navigator, to explore the Claude Nicollier Video Archives in the *Cosmos Archeology* exhibition, to be held at EPFL Pavilions, Lausanne in 2022.

Challenges involve creating a digital and interactive experience that is innovative and, most importantly, not a classical browse and search system, as it has already been developed online by the CHC. *Wall of Memories* focuses more on proposing an experience where visitors can appreciate the collection of videos as a whole, stumbling upon content they might find interesting or fantasize about and breaking the linear nature of exploring a timeline. Fur-

thermore, due to the length of the videos, proposing a way to navigate these videos is crucial, to not risk visitors only seeing the first minutes and then leaving (as it is unreasonable thinking people will stand up for even an hour watching a video in the middle of an exhibition).

1.1 Overview of similar works

With audiovisual cultural heritage developing further and further, digital humanists, especially in the field of experimental museology, are designing new ways to investigate those collections and offer them to the public in exhibitions and cultural events.

T_Visionarium [3] has been created in 2008 using the UNSW iCinema Centre's Advanced Interaction and Visualisation Environment (AVIE) system [4]. It offers the means to capture and represent televisual information, allowing viewers to explore and actively edit a multitude of stories in three dimensions. 28 hours of digital free-to-air Australian television captured over a period of one week are segmented and converted into a large database containing over 20,000 video clips. These clips are then tagged in order to create new connections. Dismantling the video data in this way breaks down the original linear narrative into components that then become the building blocks for a new kind of interactive television. Figure 1.1 displays the system with visitors inside.



Figure 1.1: *T_Visionarium* installation with visitors.

Cloudbrowsing [5],[6] offers a meaningful example of new multimedia technologies used to empower visitors to explore visual content in an interactive and visual way. Developed in 2009, the Center for Art and Media Karlsruhe (ZKM) proposes to search for a Wikipedia page and the system automatically extracts images to fill the 360° panoramic screen, forming a dynamic collage of images and sounds. The more linear and structured nature of websites is here scattered across the entire screen, slowly moving in an organic way that draws the visitor’s attention, along with the acoustic level evolving in conjunction with the content displayed. Figure 1.2 shows the installation in action.

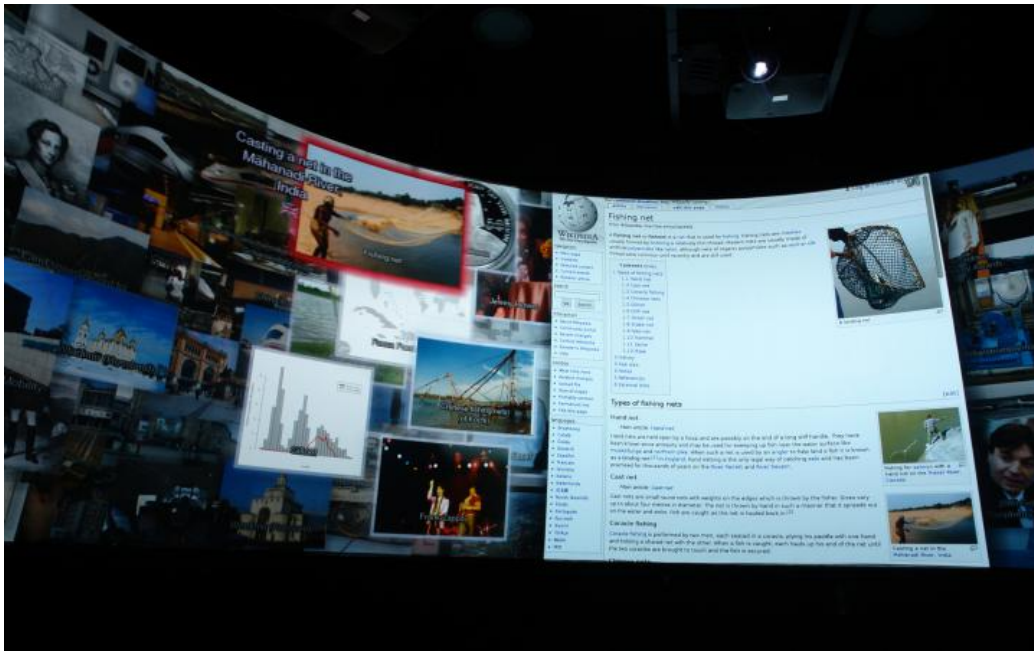


Figure 1.2: *Cloudbrowsing* installation by ZKM.

mARChive [7], the innovative interface to explore Museum Victoria’s collections proposed in 2014, uses the AVIE system to offer interactive access to eighty thousand collection records as a situated experience, inside a 360° three-dimensional exhibition display screen. Figure 1.3 depicts the system in action. This installation proposes to explore the museum collections, each visitor creating their own narrative. Users are invited to select one of the eighteen categories (represented by the eighteen circles of different colors)

and then discover the relative timeline of objects, selecting each item to get more information.



Figure 1.3: *mARChive* installation to explore Museum Victoria’s collection (credits: Volker Kuckelmeister, 2014).

Finally, *Campus Chronicles* [8], built at eM+ in 2019 for former exhibition *Infinity Room 2* [9],[10] curated by Professor Sarah Kenderdine, also uses the Linear Navigator. This temporally distributed, page-turning experience proposes to explore fifty years of press magazines relating EPFL life and events, with issues from *Polyrama* (1970–2006), *Flash* (1973–2016) and *EPFL Magazine* (2016–2018). The Linear Navigator is used as a timeline to navigate the span of years covered, and visitors have the option to select an issue (grouped by year and month) to read it in high resolution, page by page. Figure 1.4 demonstrates a visitor interacting with the application.



Figure 1.4: *Campus Chronicles* installation by eM+.

1.2 The exhibition

Cosmos Archeology is a collaboration between eM+, EPFL Pavilions and LASTRO, the EPFL astrophysics laboratory. It will feature various installations bridging the gap between art and science, showcasing fascinating physical phenomena of our universe and mesmerizing images of stars and galaxies. Rather than creating a science show, the goal is to produce a collection of innovative and mainly digital artworks based on astrophysics data.

It will be featured at EPFL Pavilions, in Lausanne, Switzerland, in 2022. Located at the heart of a vibrant international university, EPFL Pavilions is an amplifier for art, science and society, a meeting place for all disciplines. In the Pavilions, exhibitions and programs evolve in dialogue with scientific innovation and societal challenges, engaging visionary perspectives on our contemporaneity [11].

Chapter 2

Presentation of the dataset



Figure 2.1: Picture from NASA mission STS-46 with Claude Nicollier and the EPFL flag (credits: NASA, 1992).

The Claude Nicollier Video Archives [12] is a collection of 559 videos digitized and curated by the Cultural Heritage and Innovation Center (CHC) at EPFL. It is the result of two large deposits made to the CHC in 2020.

As explained in the official website, the first lot includes the making-of recordings of documentary “Destins” produced in 1998 by Jean-Luc Nicollier as part of his professional activity at RTS. This documentary traces the highlights of the personal life and career of Swiss astronaut Claude Nicollier with European Space Agency (ESA) and National Aeronautics and Space Administration (NASA). This documentary was broadcast for the first time on January 2nd, 1998 on RTS.

The second deposit was made by Claude Nicollier himself and includes a set of more than 380 VHS collected during his professional activities as ESA / NASA astronaut and military pilot in Swiss army. The covered subjects are mainly related to the fields of aeronautics and aerospace. The videos of this deposit were produced between 1990 and 2000 and consist of NASA and ESA productions. The main topics include various subjects as training or conferences around NASA missions: STS-46, STS-61, STS-75 and STS-103, in which Claude Nicollier participated as mission specialist. There are also various television programs from Swiss and international public and private channels.

In total, this collection consists of 559 videos digitized, indexed and augmented with metadata.

2.1 Digitization and curation work

The digitization project started in January 2020, handled directly by the CHC for all supports of type Betacam SP and Digital Betacam, while VHS tapes were digitized by the company Cinetis and its partner in the Netherlands.

Video files have been digitized with highest image and sound quality available in standard definition PAL (720x576) / 25 fps or in NTSC (720x486) / 29 fps. Video files bit rates are constant between 221 and 224 Mb/s. The files are encoded in uncompressed v210 format, with downsampling 4:2:2 and YUV color space. The audio files of the sequences are encoded in 24-bit PCM stereo, at 48.0 kHz sample. Finally, the video and audio are encapsulated in Quicktime containers (.mov), as presented in [13].

Thanks to the curation work, several metadata fields are made available for each video, such as the date and creator of the video. Figure 2.2 presents the number of videos per year. Notice how most videos are between 1990 and 2000, due to the second deposit of VHS tapes consisting mostly of ESA and NASA productions.

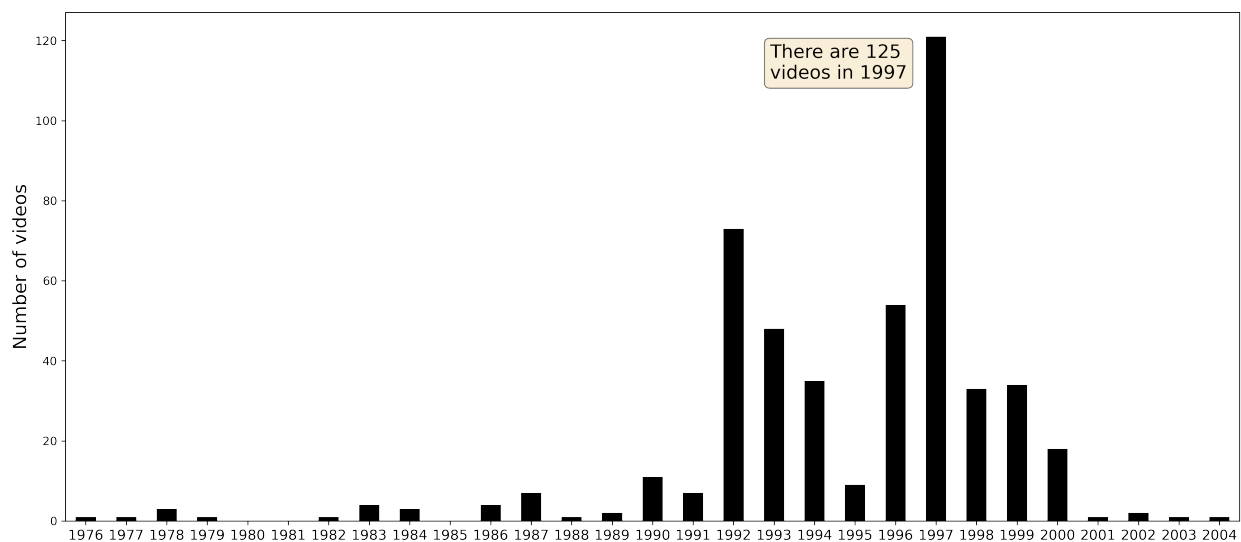


Figure 2.2: Number of videos per year.

Furthermore, videos have been indexed: each sequence is timecoded and a short text describes their content. It is worth mentioning other metadata fields are available, such as a list of subjects like people or places associated to a video. Using these tags to explore the dataset was initially considered, but in the end a more organic and visual experience has been created, rather than a browse and search system, as it is already available on the Claude Nicollier Video Archives website.

2.2 Final dataset for this installation

Out of the 559 videos in the dataset, only 476 were kept for this installation. There are two main reasons for this choice. First, as seen in 2.2, there are for the first years some gaps. Keeping the entire span of 43 years would

have resulted in either blank spaces in the virtual world, or a non-continuous timeline. Due to the linear nature of the device used, none of these options seemed to fit. This resulted in a dataset of 505 videos over a span of 23 years, from 1982 to 2004.

Second, 29 more videos were removed, either because the video or the metadata was actually missing, or because it had been digitized from supports too different from the others, such as CD-ROMs, and had a noticeably different look. In the end, the 476 videos kept for this installation came from four different sources, as outlined in table 2.1.

Type of tape	Number of videos
VHS	345
BETACAM SP	129
DIGITAL BETACAM	1
DVCAM	1

Table 2.1: Final dataset tape types composition.

2.3 Short data analysis of the final dataset

For the sake of completeness, a short data analysis is proposed to better understand the content of the final dataset, and explain some of the challenges encountered.

Figure 2.3 outlines the principal creators. NASA is the most common one, with 211 videos. 53 videos have missing information about the author. The Other category comprehends 34 creators that each have just a few videos (mostly one or two).

Then, figure 2.4 shows the distribution of videos coverage time in minutes. Notice how most of the videos are actually quite long. It would therefore be unreasonable to expect visitors to stand and watch the full videos, hence why a system control to navigate them is necessary. A time scrub bar is chosen for this installation, as it will be discussed later in section 3.

Finally, figure 2.5 presents the distribution of the number of sequences per video. Note that 44 videos are missing sequences information. As one can see, half of the videos have 8 or less sequences, with most actually having only one main sequence. There are also a few special videos with a large number of sequences, up to 176.

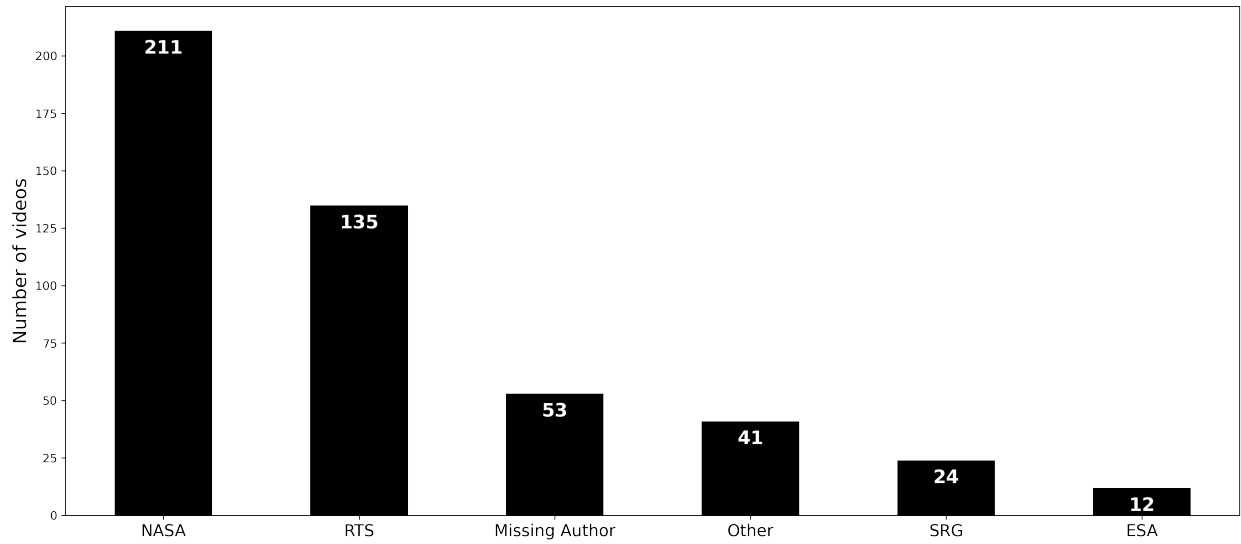


Figure 2.3: Principal video creators.

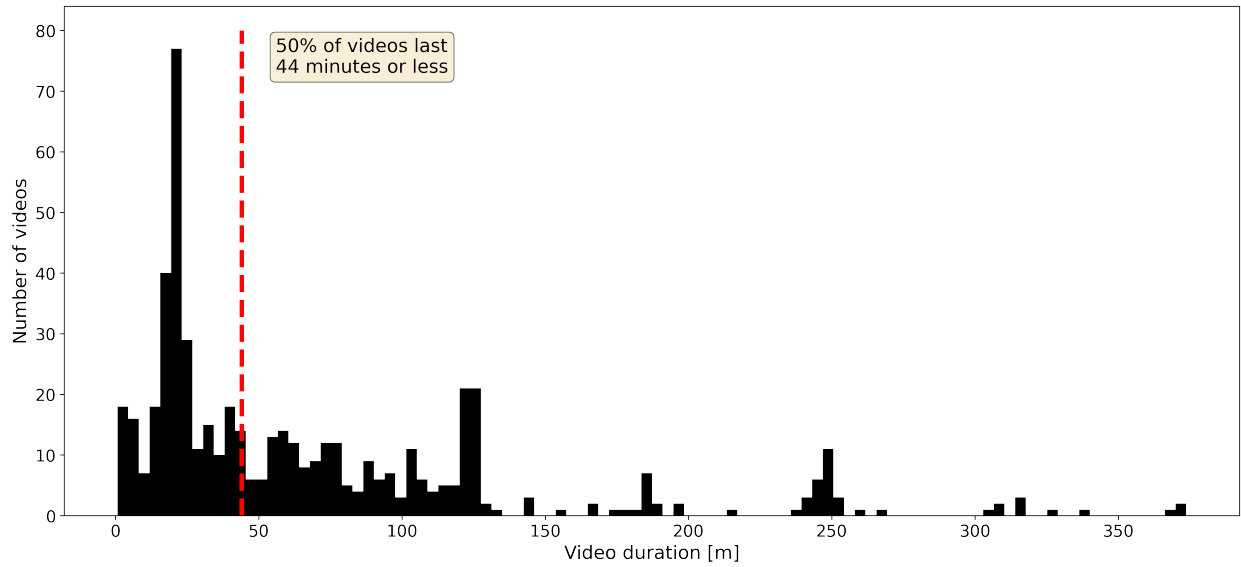


Figure 2.4: Distribution of video coverage time in minutes.

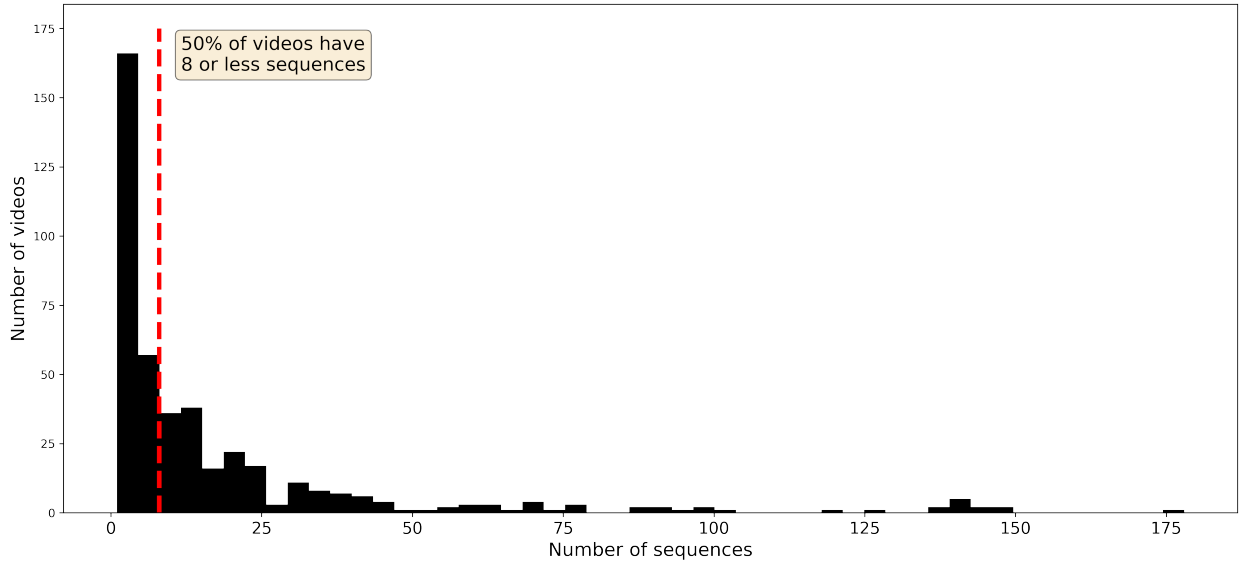


Figure 2.5: Distribution of the number of sequences per video.

2.4 Video format and compression

The collection of videos featured in this installation, in their raw format stored at the Cultural Heritage and Innovation Center, are too large to be stored locally on the Linear Navigator hardware. Thus they need to be compressed while preserving their quality. To this end, the following protocol has been designed.

Height short video excerpts representative of different types of sequences have been selected. Table 2.2 summarises them. Two video compression standards have been tested: H.264 and H.265, using `ffmpeg` to encode the different videos. The Constant Rate Factor (CRF) is used to control the trade-off between preserving video quality and lowering file size, as discussed in [14]. Lower CRF values would result in better quality, at the expense of higher file sizes. The command `"ffmpeg -i input.mp4 -vcodec CODEC -crf CRF output.mp4"` is used in the terminal, with $\text{CODEC} = \{\text{libx264}, \text{libx265}\}$ and $\text{CRF} = \{20, 22, 24, 26\}$ for H.264 and $\text{CRF} = \{24, 26, 28, 30\}$ for H.265. Table 2.3 presents the results of this compression for the first excerpt, where the visual quality is simply estimated by watching the newly-encoded videos.

Table A.1 in the Annex presents the full results. For the visual quality, "original" means the visual quality is unchanged, "ok" means there are only very minor differences that do not affect the overall quality and "less ok" means more important differences start to appear. Note this is of course quite a subjective process, and furthermore many videos already have a low visual quality due to their original support (VHS for instance).

From the results obtained, overall it seems that compressing videos and reducing size even up to 70% does not affect much the visual quality. Hence videos are encoded using the H.265 codec and a CRF of 28. The audio codec is left unchanged to AAC, simply lowering its bitrate to 128 kBit/s.

Video N°	Content	Duration [s]	File size [kb]
1	Video of written document and text on restaurant sign, outside	54	12836
2	Space shuttle flight and journalist presenting news	59	21149
3	Space shuttle flight on blue sky	38	14840
4	Gathering of people outside, good luminosity	34	12823
5	Gathering of people inside, bad luminosity	30	8686
6	Planes flying over lake and mountains	14	5262
7	People inside space shuttle, in space	38	14747
8	Interview of Claude Nicollier, clear and focused on its bust and face	23	3449

Table 2.2: Description of the video excerpts chosen for the video compression protocol.

Codec	CRF	File size [kb]	Visual quality
H.264	20	12666	original
	22	9934	original
	24	7542	ok
	26	5701	ok
H.265	24	6329	original
	26	4825	ok
	28	3728	ok
	30	2947	less ok

Table 2.3: Compression results for the video compression standards tested for the first video excerpt (see Annex A for remaining excerpts results).

2.5 Measurements with Unity’s VideoPlayer on the Linear Navigator hardware

Once the right compression format has been found and the final videos have been provided by the CHC, the loading time of each video in Unity, on the Linear Navigator hardware, can be measured.

In Unity, videos are played using the VideoPlayer component. This takes a video as input, either locally or by url, loads it and plays it, taking care at once of both the video and the audio. Calling the function Prepare() on the VideoPlayer by script loads the video, so that when calling Play(), the video output is immediately available. The time it takes to prepare a video is measured, checking the variable isPrepared, averaging over 10 iterations. Results are presented for the full 476 videos in figure 2.6. One can observe that, except for a few exceptions, most videos take less than half a second to load.

Even though this still is noticeable in some cases, it can be easily fixed by starting preparing the video while it is brought forward (in Full Video mode, as explained in section 3).

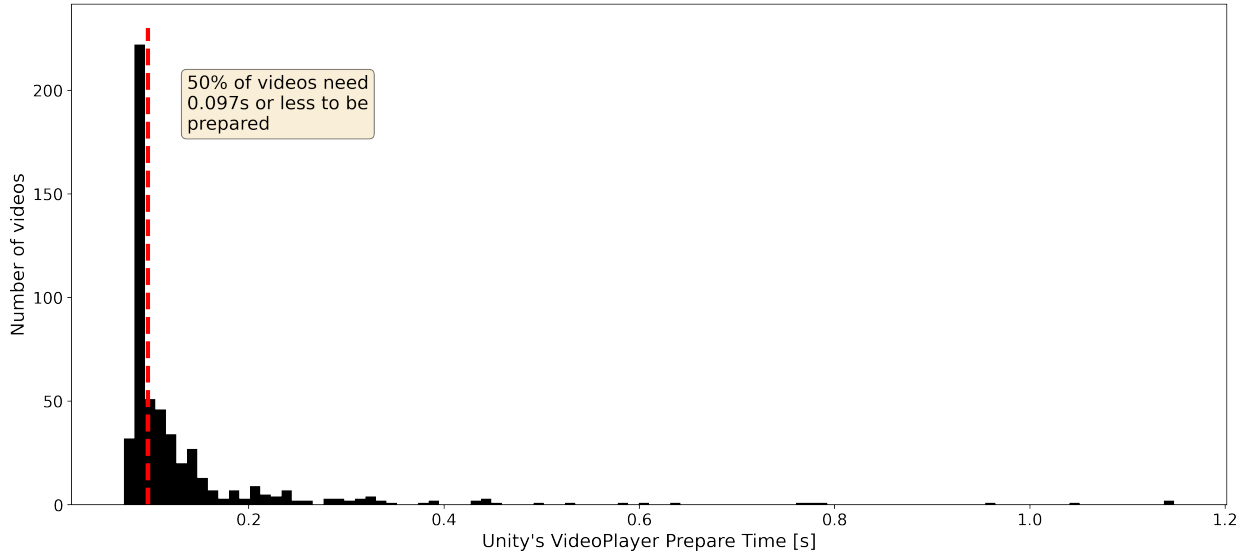


Figure 2.6: Distribution of the Unity's VideoPlayer Prepare times.

2.6 Generating the thumbnails

Since playing hundreds of videos at the same time is not really possible nor pleasing to the eye, one has to find a placeholder to represent each video. Visual summaries of a video can take many forms such as keyframes [15], skins [16], montages [17] and dynamic synopses [18]. In this case a single image, a thumbnail, to represent each video was chosen. Thumbnails are used in most (if not all) video sharing systems, with varying complexities and purposes. From simple placeholder where it is simply the first frame of the video to more elaborate images, often with text messages and complex compositions, like on Youtube, where the goal is to convey the main idea of the video and attract users.

Since the goal of this project was more on developing the actual installation, only a simple procedure has been adopted to generate thumbnails for the videos. For each video, a random frame is proposed and, if it seems good enough, it is saved. Iterating over the videos, often needing to generate just a few frames to find a good one, has taken a few hours for the entire dataset. Even though this simple approach gives good results, some research on the field of video summarization by keyframes extraction has been addressed.

Future improvements might look at applying these methods to the Claude Nicollier Video Archives (either for this installation in particular or for the archive in general).

In the field of video processing and analysis, automatically generating meaningful thumbnails is still an open problem with lots of ongoing research, since it is not only a technical problem but also a more conceptual one. Indeed, what does it mean for a thumbnail to be good? Is it because it conveys well the main theme of the videos? But what if the video talks about many different subjects? Is it because the image is of high quality, sharp and focused for instance?

[15] and [19] propose methods to summarise videos by extracting meaningful representative frames, called keyframes. The main idea is to cut the video in shorter sequences and for each one, extract a meaningful frame, where "meaningful" has to be specifically defined depending on the actual goal. Choosing a characteristic frame can for instance be achieved using a clustering algorithm such as K-clustering, the rationale being that frames of a sequence should be similar and thus clusters of frames of the same sequence should be tightly packed (low within-cluster variance) while each cluster being well separated (high between-cluster variance). The medoid of each cluster, i.e. the most central frame within a cluster, can then be selected as a representative for that particular sequence in the video.

Such an approach was tested in early development, with results presented in figure 2.7. The VGG16 convolutional neural network [20] is used to extract features from a sequence of equally temporally spaced frames from a video sequence. Feature vectors are then passed to a Principal Component Analysis (PCA) algorithm to reduce their dimension to 100 components and finally clustered using k-means, where k is automatically determined using the elbow method with the within-cluster sum of squared errors (WCSS). Mathematically, k is obtained by finding the maximum distance between the curve defined by $WCSS(k)$ and the line passing by $(k_{min}, WCSS(k_{min}))$ and $(k_{max}, WCSS(k_{max}))$. Note this defines the number of subsequences in the input video sequence. The medoid frame of each cluster is selected as the most representative one.

The procedure is implemented in Python, using Scikit-learn [21] implementation of the algorithms mentioned.

As one can observe in the example provided, this approach correctly identifies five clusters of frames and selects a keyframe for each. Whether the output keyframe is indeed the best one is a matter of interpretation, as it seems to produce good results for all clusters except the second one where, arguably, one of the four first frames would have been a better choice.

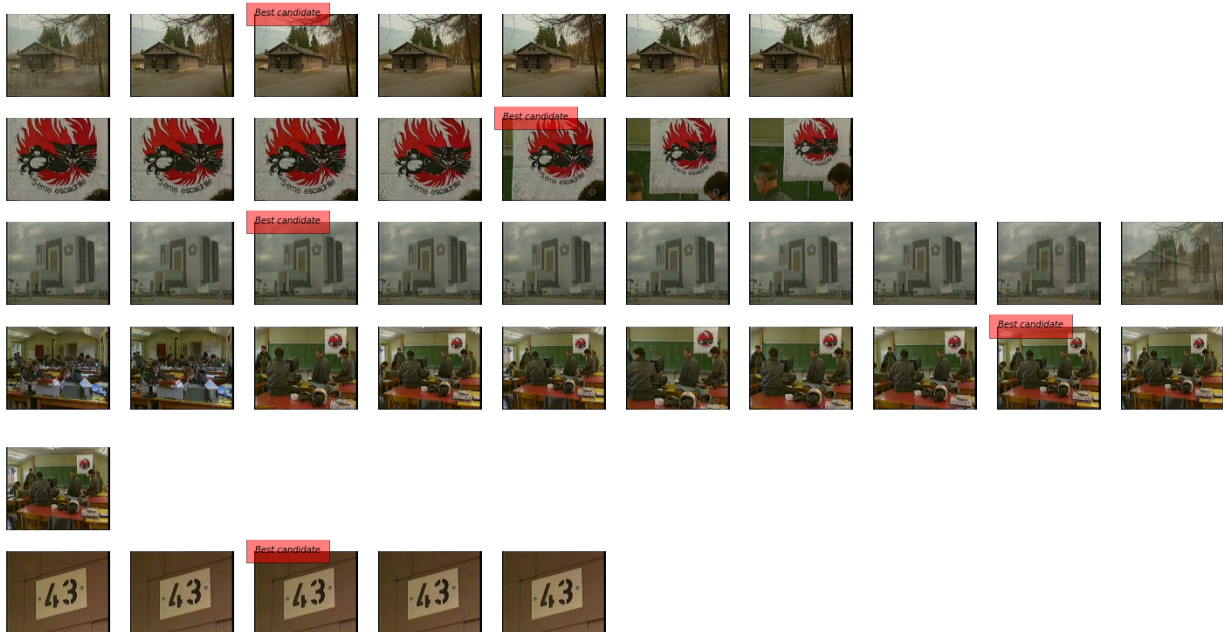


Figure 2.7: Test of the keyframes generation algorithm (keyframes for each cluster indicated) for a sequence of 27 seconds.

Main issues with this approach stems from the length of videos in the Claude Nicollier Video Archives. With long coverage times, the number of shots a video could be divided into explodes, and thus this approach would generate hundreds of keyframes. For instance, in the example above, 27 seconds of videos already result in five keyframes. How just one can then be selected, especially for hour-long videos?

All the keyframes could also be combined into a short loop, going through each one of them in order, essentially producing a GIF animation but with dozens of videos on the Linear Navigator screen at the same time, playing all of these animations would not be ideal. Performance-wise, it might be too expensive, and even if it was technically possible, it would result in a very agitated and complex scene, completely off for the contemplative nature of

this installation. Still, automatic video summarization is an interesting topic to explore when dealing with these kinds of issues, namely trying to show a whole dataset while conveying meaningful information on each item, enough so that visitors know what to expect when selecting a video to play.

Chapter 3

Description of the Experience

The idea of this installation is to explore the Claude Nicollier Video Archives, in a very visual and contemplative way. Using the Linear Navigator as a window through which look at a virtual world of videos, visitors are invited to walk along the wall, either following the screen movement or bringing it with them. Once an image catches the eye of a visitor, they can simply touch it to play the video. The idea is not to look for a specific video, or videos on specific themes ([12] already provides these features online) but rather to get a sense of the dataset as a whole, exploring it a little bit at the time. In a metaphorical way, *Wall of Memories* proposes visitors to discover Claude Nicollier's life and career, with this idea of a window through which have a peak in his memories, jumping from one year to another, in a random and more natural way than purely chronological. Furthermore, only thumbnails describe the videos at first, and thus visitors have just a visual cue to get a sense of the content of a video, reinforcing this idea of discovery.

3.1 Intended audience

Since *Wall of Memories* is thought to be part of an exhibition to be held in EPFL Pavilions, at the heart of EPFL campus in Lausanne, the intended audience is quite diverse. It will certainly attract many people directly from the campus, either students or EPFL teachers and staff, but also people from Lausanne and its surroundings, that may or may not have different backgrounds.

In spite of this, the application being mostly visual and revolving on im-

ages and videos, anyone can easily interact with it and experience wonder, without prior knowledge needed.

A case has however to be made for the language. The collection in itself features audio in various languages, mainly French and English but also German, Swiss German and Italian. No attempt to add subtitles has been made and thus not everyone will be able to understand the audio part of the videos but the images by themselves are believed to be interesting enough to compensate.

The only textual component in the installation are the sequence descriptions of the videos, currently only in English. Certainly not essential and generally quite structured, one does not need to be highly proficient in English to understand them but of course visitors with no knowledge of English are to be expected. The CHC has also provided descriptions in French and thus a way to switch between the two languages could be integrated in the visual interface. Further discussions with the rest of the team working on the exhibition might help take a decision here.

Finally, the illusion of a window through which dive into this virtual world of videos is best achieved when standing at arm length from the screen. While the difference is barely noticeable, children in particular might have a different experience (also due to the height of the rail and the screen). On a similar issue, people with difficulties to walk might not enjoy the moving nature of the Linear Navigator.

It is hard to make this kind of adjustments but hopefully the installation will still be enjoyable for every visitor.

3.2 The serendipity concept

This idea of exploring the Claude Nicollier Video Archives in a somewhat random way links back to the concept of serendipity. According to Cambridge Dictionary [22], serendipity is "the fact of finding interesting or valuable things by chance". Here, visitors are invited to dive into this collection of videos and explore it without any real guidance, and thus only by chance will they stumble upon a video that they can find interesting.

This also goes back to the idea that "information is organic", as discussed by Helen McCorry in [23]. Hence why, as discussed in more details later,

thumbnails are arranged in an organic way, slowly roaming around, as if each piece of information, here videos, had a life on their own. Visitors are not looking at a static wall of thumbnails, as if memories were tightly packed and organised, but rather at an organic flow of thoughts. Particularly important then is the Idle Mode of the application, where no one is interacting with it. The whole system will start moving on its own, jumping from one video to the next, scrolling over the wall.

3.3 Become your own storyteller

An important consequence of the Idle Mode (and of what the Linear Navigator offers) is the way one can experience the installation. This collection of videos has a linear nature in itself, each video having a specific date (at least the year). So the basic way of exploring it would be chronologically, starting in 1982 and going year by year up to 2004. However, there is no obligation to follow this route. In fact, the Linear Navigator (in spite of its name) proposes to break this linearity. One can start in 1997 then go to 2002 then back to 1985 and so on. And perhaps the crucial point here is the individuality of this process. Empowering the visitor on an individual level, *Wall of Memories* sets them to be their own storyteller, deciding which year to explore and which video to watch.

Psychologist Jerome Bruner discusses the importance of narrative for its fundamental role in creating and interpreting human culture. In *Acts of Meanings* [24], he presents two characteristics of storytelling that relate directly to museums.

The first is about how people learn. Human beings are natural storytellers: they make sense of the world and themselves through narrative, a form shared both by storytelling and history. From the time they are very young, children learn that the way to integrate their own desires with their family's norms and rules is to construct a story about their actions. This push to construct narrative, Bruner maintains, shapes how children acquire language. And the habit persists into adulthood as a primary instrument for making meaning. These storytelling skills insure our place within human society, and probably imply that information not structured as a narrative is more likely to be forgotten.

Secondly, stories have a point of view. Bruner says that stories can deal

at the same time with the canon — what we have been taught to believe in — and the exceptional, the violations of this canon. He believes that to tell a story is to take a moral stance, even if it's a moral stance against moral stances. Something happens in a story — something is wrong in the world — and its resolution serves to help us sort out our basic values and beliefs.

In *Wall of Memories*, visitors are directly confronted with one story: Claude Nicollier's life and career as a pilot and astronaut. This story is inherently linear: it has a beginning, with Nicollier's early experiences as a Swiss Air Force pilot, then its training to become a NASA astronaut and finally the four space missions he participated in. However, due to the random placement of videos within a year, and the random nature of the jumps from one year to the next in the Idle Mode, this linear story is purposely broken, scattered around. Hence visitors can build their own narrative, piece by piece, video by video. They can choose to select a thumbnail with an aircraft flying over Swiss alps and learn more about Nicollier's early career or one with a view of the Earth from a spacecraft window, with Nicollier and his colleagues presenting their work during a space mission. Visitors become their own storyteller, hopefully finding the experience more enjoyable and remembering more afterwards.

3.4 The Linear Navigator

The Linear Navigator is a 4K touch screen mounted on a 12m long rail on the wall. Screen dimensions (including the black borders) are 1295 mm × 725 mm. The hardware comprehends:

- Intel Core i7-8809G CPU @ 3.10 GHz processor
- 16GB of RAM
- Radeon RX 960 graphic card
- 500GB of memory

A Beosound Stage soundbar by Bang & Olufsen [25] is mounted on its bottom, to drastically increase sound quality. It features:

- 4 x 4" woofer drivers

- 4 x 1.5" full range drivers
- 3 x 3/4" tweeter drivers

with Dolby Atmos sound. Figure 3.1 pictures the device.

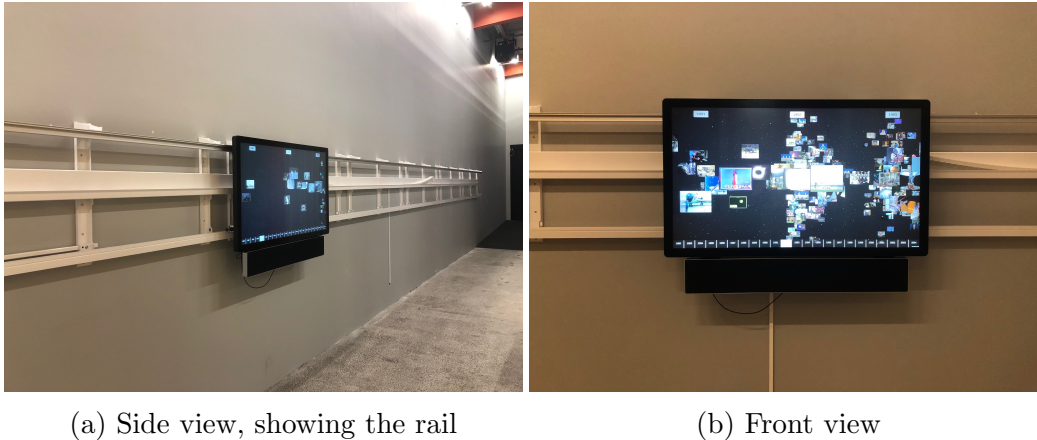


Figure 3.1: Views of the Linear Navigator device, with the sound bar hanging below.

A Copley Motion motor placed behind the screen handles its movement using an inductive sensor to measure its position on the rail and can be controlled with the *AccelnetProxy* application. Figure 3.2 presents its interface. The application has been coded by Nikolaus Völzow during his working period at eM+.

Communication between Unity's framework and *AccelnetProxy* is possible through Open Sound Control (OSC) messages, at a rate of approximately 25 UDP packets per second. The *AccelnetProxy* application sets a motor target that the motor then tries to reach, first accelerating up to a maximum speed and then slowing down towards the end. The application is capable of handling multiple successive calls, even when it is instructed to go in opposite directions. Two small spring-like elements placed on each side of the screen frame stop its movement at the limits of the 12m rail.

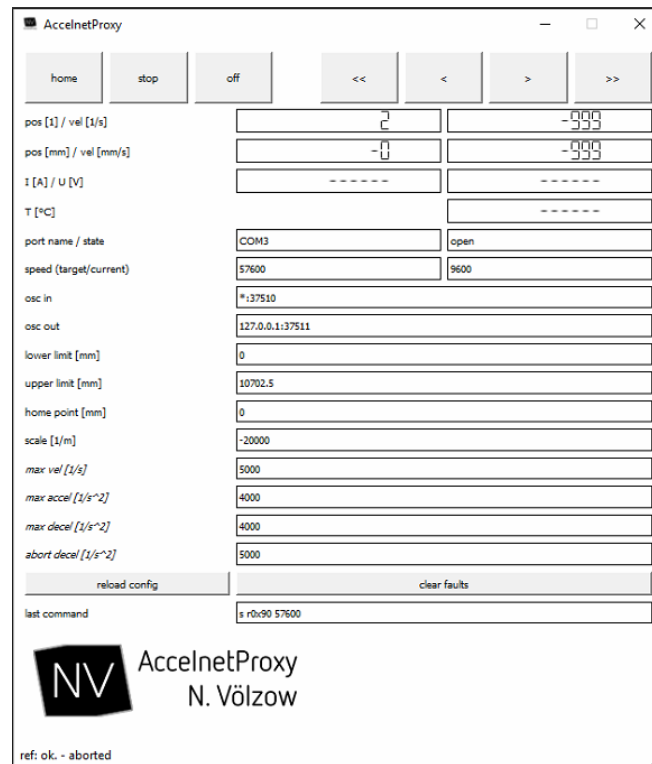


Figure 3.2: Screenshot of the AccelnetProxy application interface.

3.5 The VideoScreenElement

Each video is represented by a VideoScreenElement, which shows a thumbnail of that video and can be selected by touch to bring it forward and play the corresponding video. Due to the high number of videos for certain years (more than a hundred in 1997 for instance), VideoScreenElements are spread out along the depth and vertical axis. This means each year corresponds to a cone-like portion of space in the virtual world, narrow close to the camera and larger the further away it gets. This takes advantage of perspective and ensure that more VideoScreenElements are visible, avoiding some overlap.

To make the scene look more alive and interesting, as well as to make sure each video can be actually seen over time, each VideoScreenElement follows a certain movement, slowly roaming around in the screen plane for small amounts for a long interval and then jumping to a new position, varying its depth, in a rapid movement. The slow roaming around helps build this

contemplative atmosphere while the quick rearrangement of all screens adds an interesting event that might catch the eye of visitors passing by. It also helps redistributing the videos, ensuring more thumbnails are made visible to the user.

3.6 The virtual wall of videos

The virtual world built in Unity is thought as an extension of the physical real world, with the Linear Navigator screen acting as a window through which visitors can explore it. Since the 23 years covered by the videos in the dataset are mapped to the 12m wall, each year covers approximately 52cm on the screen. Thus a time span of three years is always shown on the screen (except for the far left and right of the timeline). Seeing videos of a few years on each side, thanks to the perspective in the virtual world, invites visitors to explore the entire timeline instead of focusing on just a portion of it.

Figure 3.3 demonstrates this wall of VideoScreenElements in the virtual world. Notice how some years, mainly around 1992 and 1997, have a lot more videos and thus the corresponding regions occupied by VideoScreenElements are larger.

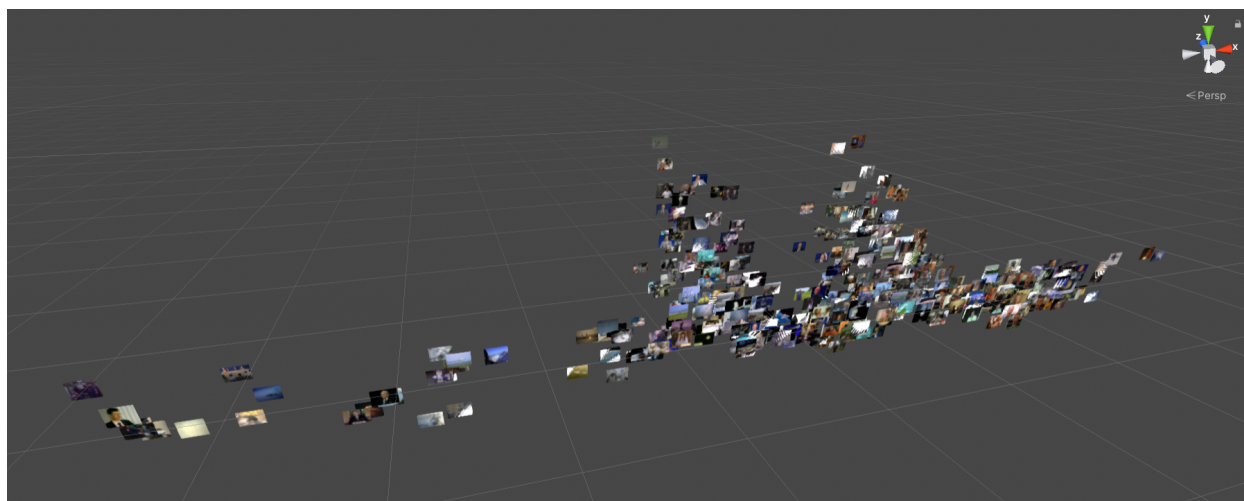


Figure 3.3: Wall of VideoScreenElements in the virtual world, Scene view in Unity.

3.7 The Full Video mode

When a video is selected, the VideoScreenElement passes in Full Video mode. It is brought forward, closer to the camera and thus occupying a large portion of the screen, and an overlay adds information about the video and a time scrub to navigate it, as shown in figure 3.4. On the top left, the video number, the author and the date are indicated. Note that not all items have information about the author, and for some only a year is available. Under the video, a short description of the sequence currently playing is available.

On the bottom, a time scrub with timecode indicates the current video time and gives visitor a way to navigate the video, moving back and forth as they want. Note that when a video is launched, it starts at the frame corresponding to the thumbnail rather than the beginning. Indeed visitors that select a thumbnail are likely to want to watch the sequence relative to that image.

Finally, a transparent grey background is added to the whole screen behind the video playing to dull down what is happening in the back, helping visitors focus on the video playing in Full Video mode.



Figure 3.4: Full Video mode screenshot of the application.

3.8 The background and sound ambiance

Linking to the general theme of the exhibition and to the particular content of the Claude Nicollier Video Archives, the background is a static star field based on Unity particle systems. To make it more interesting and varied, four different kinds of stars are randomly generated, varying their sprite, size and color.

In addition, 3D models of satellites orbiting around the wall of VideoScreenElements, at varying speeds and depths, are added. Finally, some asteroids also populate the scene. Figure 3.5 shows the result.

For the sound ambiance, a calm and space-like atmosphere music is suitable. The track chosen is *Cosmic Glow* by Andrewkn [26], available on Pixabay royalty-free music hub. It would play at normal volume when no video is selected, and fade out slowly as a VideoScreenElement transitions to Full Video mode. This would ensure there is always sound coming out of the installation, to make it more appealing to visitors passing by.

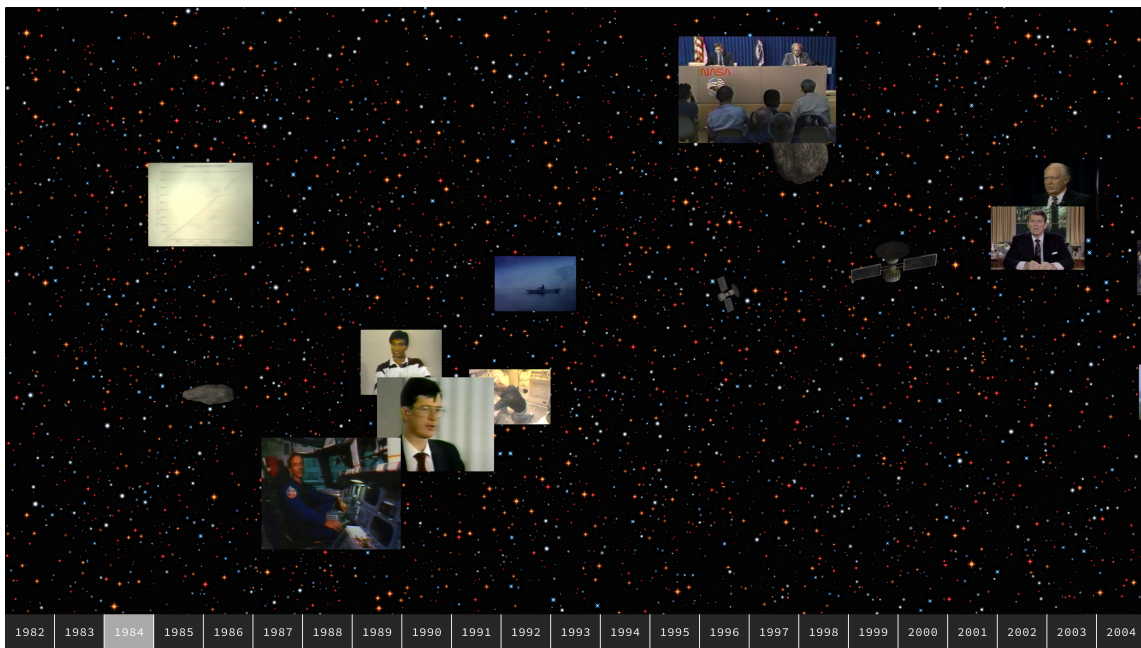


Figure 3.5: View of the background with just a few videos.

Chapter 4

Technical implementation

4.1 Unity's framework

According to the official Unity website, "Unity is the world's leading platform for creating and operating interactive, real-time 3D content, providing the tools to make amazing games". Widely used in the video-gaming industry, Unity's framework offers a range of tools that are also quite suitable to create interactive experiences for museological settings, as shown by the different installations previously created at eM+.

The user-friendly visual interface allows to quickly create a scene, placing various gameobjects¹ and tweaking their appearances. Figure 4.1 shows the Unity editor. Adding components² to those objects, either from Unity's comprehensive list of ready-to-use elements or by coding a new script, in C# or Javascript, offers near-infinite possibilities to modify and control their behaviour while playing.

Consider for instance the case of the Unity VideoPlayer. It is a ready-to-use and user-friendly built-in component that allows to play videos in a scene, by simply assigning the video file and choosing a target display, in this case a simple Quad element. It also handles the audio part of the video file, either by directly playing the soundtrack or by assigning an external audio source in the scene (thus offering more control over the sound). Unity

¹Base class for all entities in Unity Scenes.

²Base class for everything attached to GameObjects

handles preparing the video in an efficient way, so that loading times are kept low.

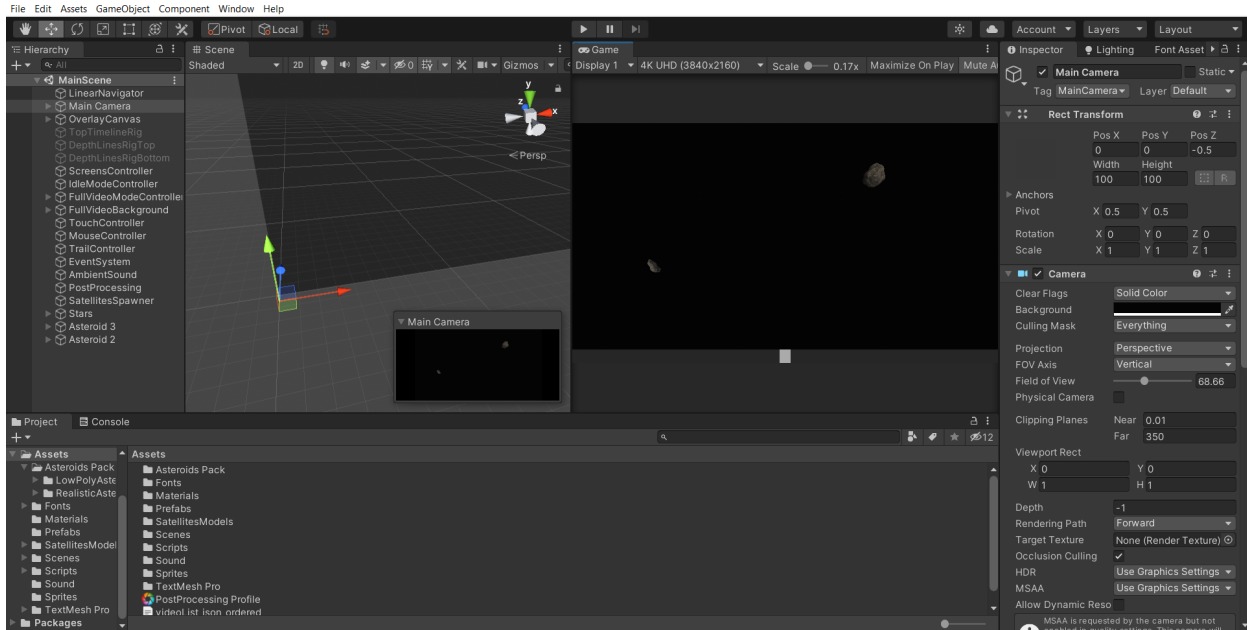


Figure 4.1: Unity editor, with the hierarchy, scene view, game view and inspector (top, from left to right) and the project files (bottom).

4.2 Handling the movement of the screen: AccelnetProxy

The first (and perhaps most important) element of this project is handling the movement of the Linear Navigator. As discussed earlier in section 3.4, the motor is controlled by an external application, AccelnetProxy. Communicating with the *Wall of Memories* application is possible through Open Sound Control (OSC) messages. Those OSC messages are represented by two elements: an address string for the motor controller device and a list of value(s) being sent to or from that device. In practice, the LinNav component in the Unity application outputs messages with a motor target that the motor controller tries to reach, and listens to messages giving the motor

position along the horizontal wall.

In order to get an illusion of a window in the virtual world, a near-perfect 1:1 mapping needs to be obtained between the movement of the screen on the horizontal physical wall and the movement of the camera in the virtual world. This helps to have a realistic sense of depth. To achieve this, the game camera field of view is empirically set to exactly mimic the view an adult would have through a window by standing at approximately half a meter from the screen (a distance considered due to visitors being able to touch and interact with the screen).

Once this is set, it is fundamental to ensure a smooth movement of the camera, following the movement of the screen. Since only approximately 25 OSC messages per second are exchanged between the two applications, the motor position must be filtered to avoid camera jumps in the virtual world. This is achieved with the OneEuroFilter [27],[28], a commonly-used algorithm to minimize lag and jitter when tracking human motion that also proves to efficiently filter out noise in this system, by tweaking the parameters.

4.3 Interacting with the application

The Linear Navigator having a touch screen, no external control system is needed. Users can interact by touch with the scene with two types of finger movement. A simple tap on one of the images will launch the corresponding video in Full Video mode, bringing it as close as possible to the scene camera. If the video is from a different year than the current one, the Linear Navigator will move to that year. The time the video takes to move forward and start playing is proportional to the years difference, so that both the Linear Navigator and the video reach their destination approximately at the same time. A colored frame appears around the selected video, to highlight it while it transitions to Full Video mode. Once a video is playing, a scrub bar on the bottom shows its progress and allows the user to jump to any given time. Note that the video starts at the frame corresponding to the thumbnail.

By keeping a finger on the screen and moving it in a swipe motion, the user can move horizontally in the scene, the bigger the swipe the quicker the

screen moves (up to its maximum velocity, reached for a swipe long a quarter of the screen width). When swiping, a trail is left on the screen, as a visual feedback on the user interaction. This also helps the user understand the possibility of swiping when they start touching the screen to interact with the system.

A timeline is also present on the bottom of the screen, so that users can instruct the system to directly go to a certain year. Note that each successive interaction with the timeline will simply override the precedent. In a similar way, swiping while the screen is moving to a certain year will simply reset its movement to follow the swipe.

4.4 Idle mode

One of the most important aspects of an interactive installation for an exhibition is its behaviour when no one is interacting with it. With plenty of others beautiful experiences planned for the exhibition, there will be periods of time where no visitors are actively engaging with *Wall of Memories*. It would be a shame if the system would simply be still, silently waiting for someone. Furthermore, a seemingly dead installation would probably appear less interesting and engaging.

It is thus crucial to make sure something is always happening, so that visitors notice the installation and are attracted to it. As previously discussed, the movement of the VideoScreenElements in the scene is perpetual, creating a continuous organic flow. This already helps making the installation feel more alive. However what is characteristic of *Wall of Memories*, compared to another screen-based installation, is the ability of the Linear Navigator to move across the wall. The big screen starting to move on its own, jumping from one year to another, is the kind of visual event in the exhibition that will, hopefully, catch visitors attention.

This reflects the ideas of *resonance* and *wonder* defined by Stephen Greenblatt [29], applied here to a more modern and digital installation:

By resonance, I mean the power of the displayed object to reach out beyond its formal boundaries to a larger world, to evoke in the viewer the complex, dynamic cultural forces from which it has emerged and for which it may be taken by a viewer to stand. By wonder I mean the power of the displayed object to stop the viewer in his or her tracks, to convey an arresting sense of uniqueness, to evoke an exalted attention.

It is the sudden movement of the Linear Navigator when it jumps to a new random year, with its mechanical sound, and the quick rearrangement of VideoScreenElements on the screen that are supposed to make visitors *wonder* and bring them to the installation. Then, when exploring it and discovering new videos, *resonance* can take place, with space missions reports of NASA, filmed directly in space shuttle cabins, Claude Nicollier and his fellow astronauts floating around in no gravity, the Earth visible through the shuttle windows.

Finally, it is worth mentioning that the Idle Mode is by default active, and is interrupted when users interact with the application. However, after a certain delay, if no interactions are registered, it starts again, silently (i.e. without warning). This ensures that the Linear Navigator stays in front of visitors when they are interacting, because if for instance they click on the timeline to go to a certain year, they would not want to change destination year midway. At the same time, when visitors leave and no one is interacting anymore with the screen, the system needs to resume the Idle Mode, to start moving again and attracting new visitors. Incidentally, if someone is standing in front of the screen for a certain time without interacting, simply admiring the thumbnails and their movement, they will get the pleasant surprise of the Linear Navigator starting to move on its own.

This behaviour was different in the *Campus Chronicles* installation discussed in section 1.1: the Idle Mode was similar but users would get asked to interact after a certain delay, and if still no interactions were detected, the system would start moving. The choice here was to avoid that the Linear Navigator would leave while reading a page (and thus not necessarily touching the screen). However, such a visual prompt, of which a mock-up for *Wall of Memories* is proposed in figure 4.2, is found to break the immersion of the user during the experience and thus rejected.

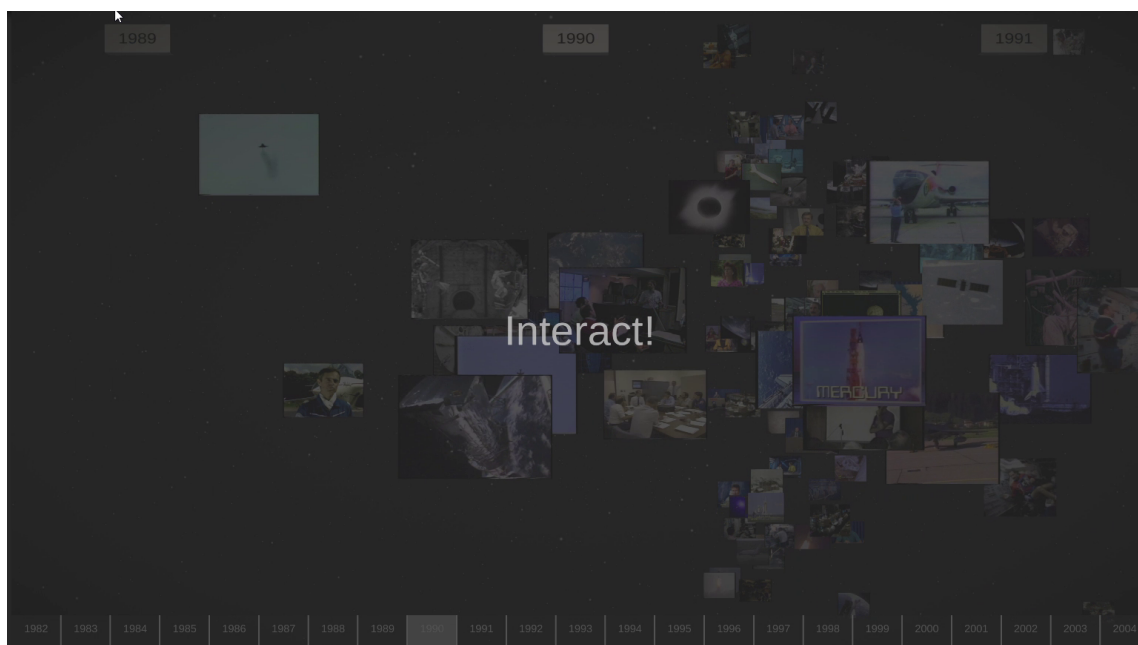


Figure 4.2: Visual prompt mock-up to interact before starting the idle mode (old design).

4.5 The different elements in the scene

Conceptually, it is better to hand control of a certain functionality to a certain element in a scene, rather than having control of all functionalities attached to the same gameobject. Here, a somewhat abstract and conceptual explanation on how the scene is built and how the various components interact between them is provided.

4.5.1 The LinearNavigator and the Camera

The LinearNavigator gameobject handles the communication between the Unity application and the AccelnetProxy, via OSC messages, as explained in 4.2. To do so, the gameobject implements the OSC component and the LinNav component. The former provides OSC functionalities, based on the .NET Visual C# library for the Make Controller Kit, while the latter controls the motor. This is also where the OneEuroFilter is used, to interpolate the motor position.

For development purposes, the LinNav and CameraController handle two modes, whether the application is running on the actual Linear Navigator or on another system. Indeed, since the camera movement is based on the motor position, it needs to be simulated if there is no motor position to read. In the latter, the camera movement is simulated by simply interpolating between the current position and the target position, computed as a function of the target year. The `Mathf.Lerp()` function is used, as a simple linear movement is enough for development purposes.

Otherwise, the motor position along the physical wall is mapped to the virtual wall length, so that the camera follows the Linear Navigator screen.

4.5.2 The wall of VideoScreenElements

When the application starts, the wall of VideoScreenElements is initialized. In pseudo-code, the simple algorithm to populate the wall is defined in 1, where `VideosOfYear(year)` returns the videos of the input year, `GenerateRandomPosition(year)` generates a random position within a volume assigned to each year (a portion of the wall) and `CreateVideoScreenElement(video, position)` instantiates a VideoScreenElement with the given video at the input position.

Algorithm 1 Wall of VideoScreenElements Initialization Algorithm

```
year ← 1982
while year ≤ 2004 do
    listVideos ← VideosOfYear(year)
    for each video ∈ listVideos do
        position ← GenerateRandomPosition(year)
        CreateVideoScreenElement(video, position)
    end for
    year ← year + 1
end while
```

The position of each VideoScreenElement is randomly generated within a volume assigned to each year. The horizontal position $xPos$ along the x axis is given by equation 4.1:

$$xPos = year * \frac{WallLength}{NumberOfYears} + randomShift \quad (4.1)$$

The position $zPos$ along the z axis is uniformly sampled between $z = 0$ and $z = WallDepth$. Finally, the vertical position $yPos$ along the y axis is given by an uniformly sampled vertical position between $-WallHeight$ and $WallHeight$, multiplied by a scale factor γ given by equation 4.2, where γ_{Global} is a constant defining the maximum vertical spread:

$$\gamma = \begin{cases} \gamma_{Global} * (1 - \frac{WallDepth - zPos}{WallDepth})^2 & \text{if } zWall \geq ScaleZStart \\ 1 & \text{if } zWall < ScaleZStart \end{cases} \quad (4.2)$$

In practice, γ is large when the VideoScreenElement is far away ($zPos$ close to $WallDepth$) so that the elements are spread out on the vertical direction and more of them are visible. For small z values (smaller than $ScaleZStart$), the scale factor is simply set to 1 so that all VideoScreenElements are not in the middle (y coordinate close to 0).

Figure 4.3 plots γ with $\gamma_{Global} = 4$ and $ScaleZStart = 1$ over the range $[0, 2]$.

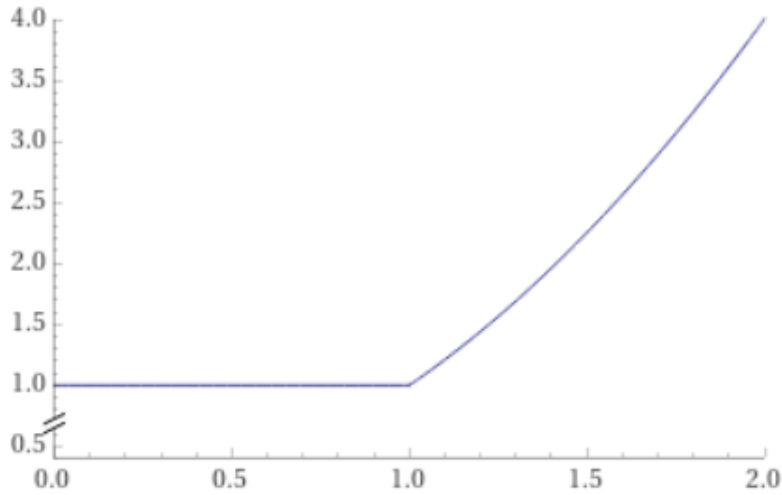


Figure 4.3: Plot of the γ factor with $\gamma_{Global} = 4$ and $ScaleZStart = 1$ over the range $[0, 2]$ (notice the cut on the vertical axis low values).

4.5.3 Handling the movement of the VideoScreenElements

As mentioned before, the VideoScreenElements have each a movement on their own, forming a very organic flow. This is, in C#, handled with Coroutines, looping them for a continuous and infinite movement. Conceptually, the movement has two steps: first, it roams around, slowly and around the same point; second, it jumps quickly to an entirely new position within the volume of the wall assigned to the corresponding year of the video. Jumps of all the VideoScreenElements are synchronized so that visitors can experience a quick rearrangement of the whole scene, a quite appealing visual event.

For the first part, it interpolates between the current position and the target position, computed taking the current position and adding a small vector in the (x,y) plane (the plane of the screen) using C# `Random.insideUnitCircle()` multiplied by a scale factor. For the second part, the jump target destination is computed following the same procedure outlined in section 4.5.2.

Note that, when in Full Video mode, as soon as the roaming part is finished, the Coroutine is exited, to avoid having a big rearrangement of the thumbnails in the background that would distract visitors from the video playing.

4.5.4 The bottom timeline

As mentioned before, on the bottom of the Linear Navigator screen lies a timeline that allows visitors to select a year and have the system moves towards it. The timeline listens to touch (or mouse) input and maps the horizontal position along the timeline to the range of years.

When a new year is set, the timeline calls the relevant function to close the Full Video mode, if a video is playing, and instructs the LinNav component to move to the target position along the physical wall.

Finally, a visual element, lighter than the timeline, indicates the current year, following the camera position along the virtual wall of VideoScreenElements, as indicated in figure 4.4.



Figure 4.4: Zoom on the bottom timeline (current year is highlighted).

4.5.5 Handling touch and mouse input

For development, both mouse and touch controls have been taken into account, even though in the final installation only touch will be necessary. Fortunately, Unity’s Input system handles both cases in a similar way, so that the two systems work, at least conceptually, in the same way. To explain the procedures applied, only touch will be referred to, as for the mouse input, it will essentially be the same.

There are two kinds of input visitors can use to change the behaviour of the application: a simple tap with one finger or a swipe motion.

The first step in handling users input is thus to differentiate between the two. A simple check is applied by computing the euclidean distance Δ between the initial position of the touch and the updated position when the touch has moved or has ended. This is easy to check with the Unity’s Touch structure, describing the status of a finger touching the screen. This provides the current position of the touch as well as its phase (whether it has just started, ended or moved).

Δ is then compared with two distances: `minDistancePixelToSwipe` and `maxDistancePixelToTap`. The former controls how much a user has to move the finger on the screen to detect its movement as an actual swipe, to avoid accidental movements, while the latter differentiates between a tap and a swipe (as even with a tap there might be a small Δ). To better check for a tap, its duration is also compared to a time interval to tap, meaning that holding down the finger in the same position longer than this interval will not count as a tap.

Once the difference between a swipe and a tap has been determined, the touch control system will either instruct the LinNav component to move according to the swipe length or check whether a target object (i.e. a Video-ScreenElement) has been selected.

The swipe amount is computed as the percentage of the Linear Navigator

screen the finger has run through, multiplied by a factor of 4. In practice, this means the maximum swipe velocity is reached for a swipe distance of a fourth of the Linear Navigator screen, in either directions. The direction of the movement of the Linear Navigator is the same of the swipe motion, so that users can literally "bring the system with them" by walking in a direction while keeping a finger on the screen. To provide users with a visual feedback, a trail effect is added, as demonstrated in figure 4.5.

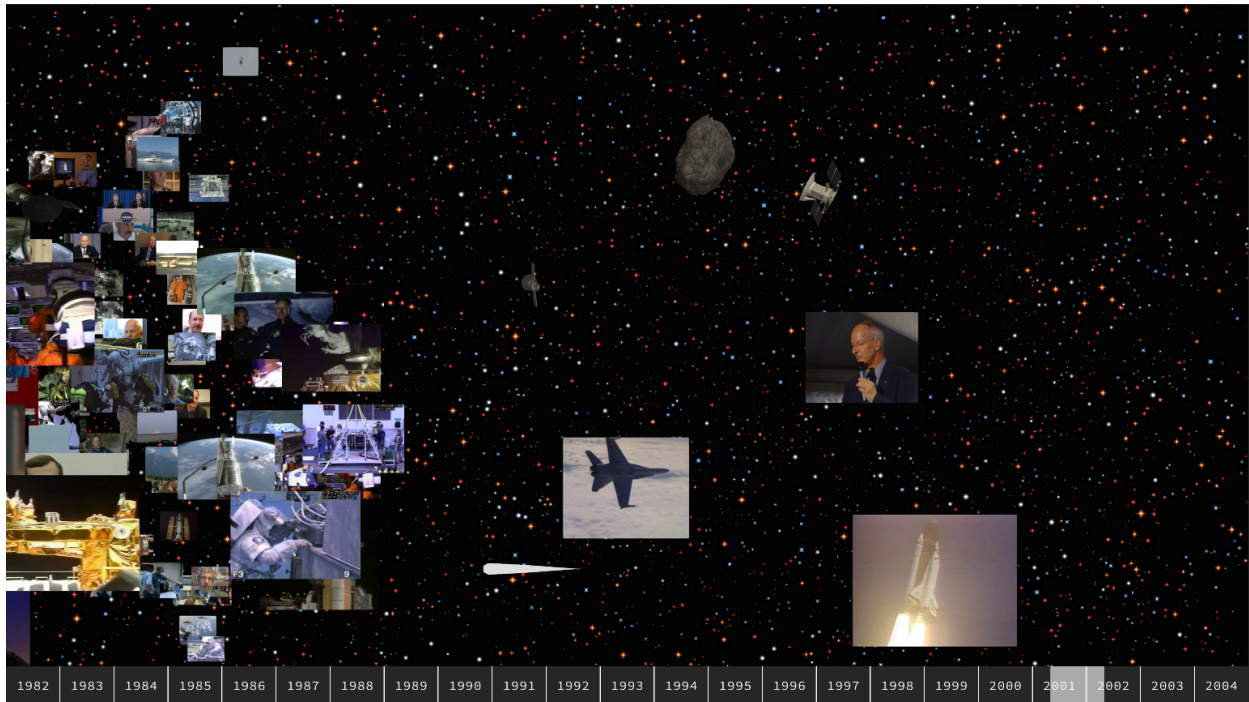


Figure 4.5: Trail effect while swiping (in this case, the screen moves to the left).

On the other hand, if a tap is detected, the Unity Raycast system is used to check whether the tap has selected or not a `VideoScreenElement`. This is a rather common technique to check for interactions in 3D virtual world, as outlined in [30],[31]. Following [32] taxonomy for selection techniques, a touch tap on a `VideoScreenElement` functions as both Indication of Object and Indication to Select, while a colored frame appearing upon selection functions as Feedback, illustrated in figure 4.6.

Raycast are particularly useful in this kind of setting because the objects

users are trying to interact with are simple 2D Quad element floating around in the virtual world. Since the Raycast stops at the first gameobject it intersects, there is no risk to tap on a certain thumbnail and accidentally play a video from a VideoScreenElement behind it. The system only needs to check whether the gameobject it intersects has a VideoScreenElement component, and that it is not already playing in Full Video mode.



Figure 4.6: Colored blue frame providing feedback on the selection of a video.

4.5.6 Generating the background

As mentioned before, the background is a combination of a star field, 3D models of satellites orbiting around the wall of VideoScreenElements and static asteroids rotating on themselves.

The star field is created using Unity ParticleSystem. Particle systems are common techniques in 3D rendering software and video game engines. It is a graphical technique that simulates complex physically-based "fuzzy" effects,

often used to render smoke or fire effects for instance. The idea is to spawn over time many particles, each with a certain sprite, and controlling various parameters such as size, velocity, lifetime or physical properties.

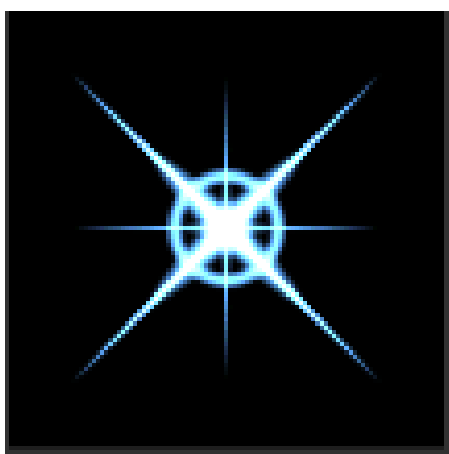
Usually this means new particles are continuously created while existing ones are destroyed, but to generate a star field, this behaviour is disabled. Instead, all particles are spawned at the start of the application and their lifetime is set to infinity. They are also static, as it would not really make sense for stars to move or change size over time.

Once the basic idea is set up, it is a matter of customization, changing the particles appearance. The main elements to focus on are the sprites used, their color, their emission intensity and their size, as well as defining the 3D region in the virtual world in which to spawn the particles. This region has of course to be far away and large enough to cover the entire screen while not covering other objects in the screen, thus simulating the star field. Figure 4.7 shows a reference image used.

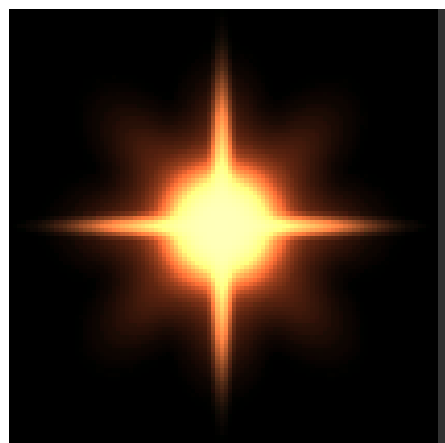


Figure 4.7: Reference image used for the star field (credits: NASA)

In Unity, the Particles/StandardUnlit shader is used for the particles, set with a combination of sprites and color for the Albedo and the Emission maps. Since reflections on the particles are not needed, the Unlit shader is preferred to the Particles/Surface shader, as it is faster. Figure 4.8 presents the four kinds of star models used, with varying sprites and colors. The sprite models are then used in the Unity ParticleSystem, spawned with random sizes within different similar ranges for each model.



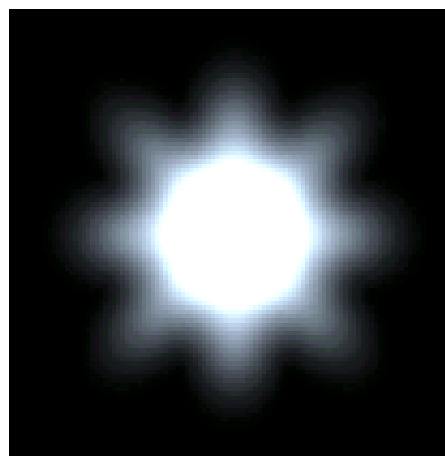
(a) Blue star



(b) Orange star



(c) Red star



(d) White star

Figure 4.8: The four models of sprites used for the Particle System star field.

Figure 4.9 demonstrates the final result, without the other elements in the scene.

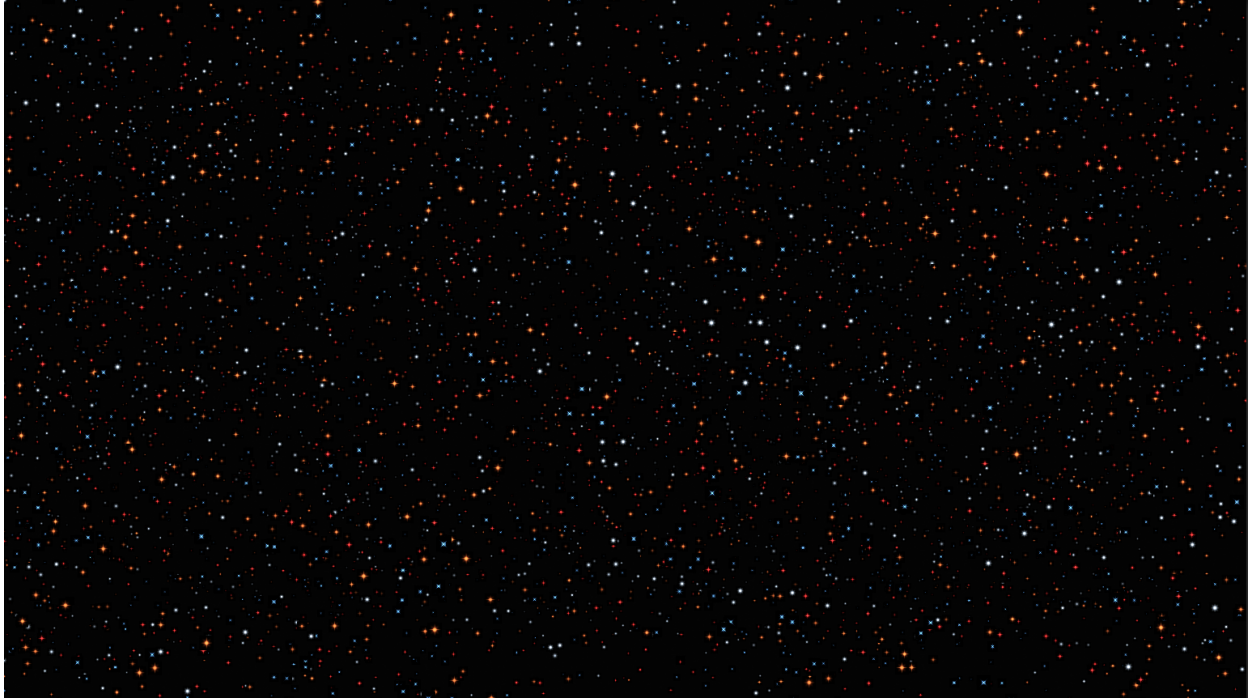


Figure 4.9: Particle System star field in Unity, without the other elements in the scene.

In addition to the star field, 3D models of satellites orbiting around the wall of VideoScreenElements, at varying speeds and depths, and static asteroids rotating on themselves are added. Figure 4.10 shows the satellites and asteroids models used and figure 4.11 demonstrates their placement in the Unity Scene View.

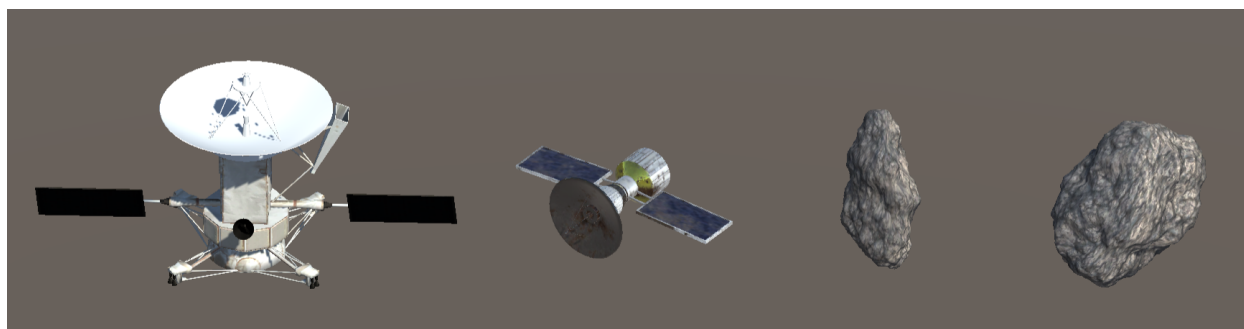


Figure 4.10: Satellites and asteroids models used for the background.

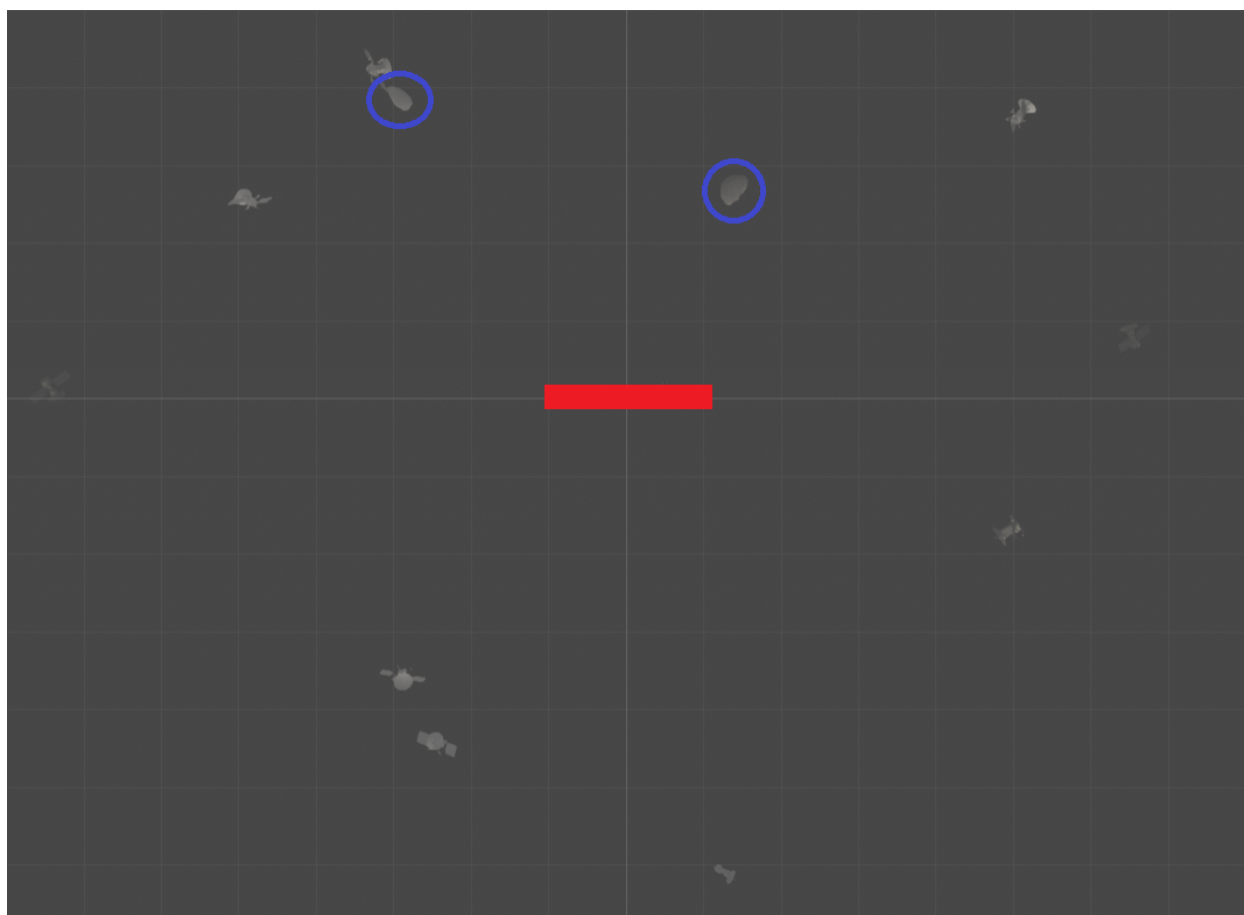


Figure 4.11: Scene view in Unity with 10 orbiting satellites and 2 static asteroids, circled in blue (wall of thumbnails in red).

Chapter 5

Further improvements

5.1 Updates to the dataset

Due to the modular nature of the virtual world built in Unity, it is straightforward to update the collection of videos, if the CHC were to add more content over the following months, or modify it. In particular, sequence textual descriptions might need to be changed to have a less sturdy kind of texts.

Similarly, would subtitles for the video be produced, integrating them to the visual interface of the Full Video mode could be interesting. This could indeed increase accessibility to the installation, especially when videos are not in a language understood by visitors, or for people with difficulties hearing. This could however pose certain challenges, as simply overlaying the subtitles on the video is not always going to be seamless, due to some videos already having textual elements. Furthermore, textual elements are already present in the visual interface, with the textual description below the video. One could however simply make the case of replacing the sequence description with the subtitles.

5.2 Integrating VIRUP data

One of the key reasons why *Cosmos Archeology* came to be created was to showcase the incredible work that resulted in the VIRUP system [33], a project conducted at the Laboratory of Astrophysics LASTRO of EPFL.

Upon discussion with Laurent Novac, the idea is to integrate part of the astrophysical data used by VIRUP into *Wall of Memories*. Further development might therefore look at replacing the randomly-generated star field with actual data of stars and constellations in our universe, scaled to fit the background of this installation.

5.3 Finalising the installation for the exhibition

Even though *Wall of Memories* is functionally complete, some elements still need adjustments and revisions, in particular to make it coherent with the rest of the exhibition. This refers mostly to the 2D design (choice of colors, fonts, UI style) and the sound design.

These have been addressed during development, to propose an independent installation ready on its own, but once the rest of the exhibition is ready and overall work on the whole arrangement can begin, changes to *Wall of Memories* are to be expected.

Furthermore, the actual setting of this piece in the exhibition space as to be discussed. At the moment, the project is to paint on the physical wall, on top of the Linear Navigator rail, the timeline, similarly to how it was previously done for *Campus Chronicles*, as demonstrated in figure 5.1. This would of course depends on nearby installations. A brief biography of Claude Nicollier might also be beneficial, to put the installation in context.



Figure 5.1: *Campus Chronicles* in the *Infinity Room 2* exhibition space, with the timeline painted on top.

Chapter 6

Conclusion

For this Master's Project in Digital Humanities, conducted at the Laboratory of Experimental Museology, under the supervision of Professor Sarah Kenderdine, at EPFL, a digital and interactive installation called *Wall of Memories* has been developed. Featuring in the upcoming exhibition *Cosmos Archeology* at EPFL Pavilions in 2022, it proposes to discover the Claude Nicollier Video Archives, a collection of videos regarding Swiss astronaut Claude Nicollier.

Rather than proposing a simple (and common) browse and search system, the idea of this installation is to create a contemplative experience where visitors can appreciate the collection of videos as a whole, and discover new content by chance. Serendipity and the exciting process of exploring the virtual world built in Unity, with only visual cues in the form of thumbnails on the videos, are key concept of this project. Similarly, breaking the inherently linear nature of a timeline and taking a twist on the Linear Navigator, *Wall of Memories* induces users to jump from one year to another, thus becoming their own storyteller, building an unique narrative piece by piece, one video at the time.

From a technical point of view, this project has been a great opportunity to get a more thorough understanding of Unity's framework and obtain experience in the domain of experimental museology. This encompasses both concepts specific to this installation, such as the initialisation of the wall of VideoScreenElements and handling their movement, or setting up the touch system control for users to interact; as well as more general concepts related

to the creation of digital and interactive experiences. A better understanding of audiovisual data, such as video formats, has also been acquired.

From the humanist side, taking a collection of audiovisual content and researching ways to offer them to the public has been extremely rewarding. Digital Humanities cover of course a very large domain of disciplines and research fields but the way one can use modern technologies to shape the way visitors experience artworks and cultural heritage is a fascinating topic that I wish to pursue. This project has indeed certainly confirmed my wishes to start a career in the field of experimental museology, contributing to a new generation of digital projects to (re)discover cultural heritage.

Bibliography

- [1] Radio Télévision Suisse. *Les archives de la RTS*. URL: <https://www.rts.ch/archives/>. (accessed: 29.08.2021) (Go back).
- [2] Alain Dufaux and Thierry Amsallem. “The Montreux Jazz Digital Project: From preserving heritage to a platform for innovation”. In: *Journal of Digital Media Management* 7 (4 2019), pp. 315–329 (Go back).
- [3] Neil Brown et al. *T-Visionarium project description*. URL: http://icinema.unsw.edu.au/projects/t_visionarium/project-overview/. (consulted: 08.08.2021) (Go back).
- [4] Matthew McGinity et al. “AVIE: A Versatile Multi-User Stereo 360° Interactive VR Theatre”. In: *Proceedings of the 2007 Workshop on Emerging Displays Technologies: Images and beyond: The Future of Displays and Interacton*. EDT '07. San Diego, California, USA: Association for Computing Machinery, 2007, 2–es. ISBN: 9781595936691. DOI: 10.1145/1278240.1278242. URL: <https://doi.org/10.1145/1278240.1278242> (Go back).
- [5] Bernd Lintermann et al. *Cloudbrowsing, an interactive Installation for Panorama Screen*. 2009. URL: <https://artsandculture.google.com/asset/cloudbrowsing-an-interactive-installation-for-panorama-screen-bernd-lintermann/sAGDQ4-XWR93cg?hl=en>. (accessed: 29.08.2021) (Go back).
- [6] Bernd Lintermann et al. *Cloudbrowsing official project description page*. 2009. URL: <https://zkm.de/en/project/cloudbrowsing-project>. (accessed: 06.09.2021) (Go back).
- [7] Sarah Kenderdine and Timothy Hart. “mARChive: Sculpting Museum Victoria’s Collections”. In: *MW2014: Museums and the Web* (Feb. 2014) (Go back).
- [8] Laboratory of Experimental Museology eM+. *Campus Chronicles video demonstration*. URL: <https://www.youtube.com/watch?v=xjp9laq-V-4>. (consulted: 03.09.2021) (Go back).
- [9] Laboratory of Experimental Museology eM+. *Infinity Room 2 official website*. URL: <https://archiveweb.epfl.ch/infinityroom2.epfl.ch/#/>. (consulted: 03.09.2021) (Go back).

- [10] Laboratory of Experimental Museology eM+. *Infinity Room 2 virtual tour*. URL: <https://spaces.3dvrspaces.com/3dvr/artlab-infinity-room-2/fullscreen/>. (consulted: 03.09.2021) (Go back).
- [11] EPFL Pavilions. *EPFL Pavilions website*. URL: <https://epfl-pavilions.ch/>. (consulted: 08.09.2021) (Go back).
- [12] Cultural Heritage and Innovation Center. *Claude Nicollier Video Archives*. 2020. URL: <https://mjf-database.epfl.ch/claudenicollier/browse.html>. (accessed: 29.08.2021) (Go back).
- [13] Cultural Heritage and Innovation Center. *Claude Nicollier Video Archives Digitization Process*. 2020. URL: <https://mjf-database.epfl.ch/claudenicollier/about.html>. (accessed: 29.08.2021) (Go back).
- [14] Werner Robitza. *CRF Guide (Constant Rate Factor in x264, x265 and libvpx)*. 2017. URL: <https://slhck.info/video/2017/02/24/crf-guide.html>. (accessed: 20.07.2021) (Go back).
- [15] Jingjing Meng et al. “From Keyframes to Key Objects: Video Summarization by Representative Object Proposal Selection”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 1039–1048. DOI: 10.1109/CVPR.2016.118 (Go back).
- [16] Michael Gygli, Helmut Grabner, and Luc Van Gool. “Video Summarization by Learning Submodular Mixtures of Objectives”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015, pp. 3090–3098 (Go back).
- [17] Min Sun et al. “Salient Montages from Unconstrained Videos”. In: vol. 8695. Sept. 2014, pp. 472–488. ISBN: 978-3-319-10583-3. DOI: 10.1007/978-3-319-10584-0_31 (Go back).
- [18] A. Rav-Acha, Yael Pritch, and Shmuel Peleg. “Making a Long Video Short: Dynamic Video Synopsis”. In: vol. 1. July 2006, pp. 435–441. ISBN: 0-7695-2597-0. DOI: 10.1109/CVPR.2006.179 (Go back).
- [19] Aditya Khosla et al. “Large-Scale Video Summarization Using Web-Image Priors”. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition*. 2013, pp. 2698–2705. DOI: 10.1109/CVPR.2013.348 (Go back).

- [20] Karen Simonyan and Andrew Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. 2015. URL: <http://arxiv.org/abs/1409.1556> (Go back).
- [21] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830 (Go back).
- [22] Cambridge University Press. *Definition of serendipity from the Cambridge Advanced Learner’s Dictionary and Thesaurus*. 2021. URL: <https://dictionary.cambridge.org/dictionary/english/serendipity>. (accessed: 29.08.2021) (Go back).
- [23] Helen McCorry. “Museums, the Web and the Serendipity Facilitator (in Conference Proceedings of Delivering Diversity; Promoting Participation)”. In: *MDA information* (2001), pp. 133–135 (Go back).
- [24] Jerome Bruner. *Acts of meaning*. Harvard University Press, 1990 (Go back).
- [25] Bang & Olufsen. *Beosound Stage soundbar official description page*. URL: <https://www.bang-olufsen.com/en/fr/speakers/beosound-stage>. (consulted: 06.09.2021) (Go back).
- [26] Andrewkn. *Cosmic Glow*. URL: <https://pixabay.com/music/ambient-cosmic-glow-6703/>. (consulted: 12.08.2021) (Go back).
- [27] Géry Casiez, Nicolas Roussel, and Daniel Vogel. “1€ Filter: A Simple Speed-Based Low-Pass Filter for Noisy Input in Interactive Systems”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2012, pp. 2527–2530. ISBN: 9781450310154. URL: <https://doi.org/10.1145/2207676.2208639> (Go back).
- [28] Géry Casiez, Nicolas Roussel, and Daniel Vogel. *1€ Filter implementation in various languages*. URL: <http://cristal.univ-lille.fr/~casiez/1euro/>. (consulted: 01.09.2021) (Go back).
- [29] Stephen Greenblatt. “Resonance and Wonder”. In: 1991. Chap. 3, pp. 43–56 (Go back).

- [30] Ferran Argelaguet Sanz and Carlos Andujar. “A Survey of 3D Object Selection Techniques for Virtual Environments”. In: *Computers and Graphics* 37.3 (May 2013), pp. 121–136. DOI: 10.1016/j.cag.2012.12.003. URL: <https://hal.archives-ouvertes.fr/hal-00907787> (Go back).
- [31] Doug A. Bowman and Larry F. Hodges. “An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments”. In: *Proceedings of the 1997 Symposium on Interactive 3D Graphics*. I3D '97. Providence, Rhode Island, USA: Association for Computing Machinery, 1997, 35–ff. ISBN: 0897918843. DOI: 10.1145/253284.253301. URL: <https://doi.org/10.1145/253284.253301> (Go back).
- [32] Doug A. Bowman, Donald B. Johnson, and Larry F. Hodges. “Testbed Evaluation of Virtual Environment Interaction Techniques”. In: *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*. VRST '99. London, United Kingdom: Association for Computing Machinery, 1999, pp. 26–33. ISBN: 1581131410. DOI: 10.1145/323663.323667. URL: <https://doi.org/10.1145/323663.323667> (Go back).
- [33] Laboratory of Astrophysics LASTRO. *VIRUP : The Virtual Reality Universe Project*. URL: <https://www.epfl.ch/labs/lastro/public-outreach/virup/>. (consulted: 06.09.2021) (Go back).

Appendix A

Appendix

A.1 Compression results

Table A.1: Compression results for the video compression standards tested

Excerpt N°	Codec	CRF	File size [kb]	Visual quality
1	H.264	20	12666	original
		22	9934	original
		24	7542	ok
		26	5701	ok
	H.265	24	6329	original
		26	4825	ok
		28	3728	ok
		30	2947	less ok
2	H.264	20	21802	original
		22	17462	original
		24	13490	original
		26	10157	original
	H.265	24	10459	original
		26	7933	original
		28	5946	original
		30	4432	original

Continued on next page

Table A.1: Compression results (continued)

Table A.1 – continued from previous page

Excerpt N°	Codec	CRF	File size [kb]	Visual quality
3	H.264	20	15685	original
		22	12827	original
		24	10230	original
		26	7929	original
	H.265	24	6889	original
		26	5417	original
		28	4170	original
		30	3192	ok
4	H.264	20	13326	original
		22	10636	original
		24	8287	original
		26	6373	ok
	H.265	24	8053	original
		26	6233	original
		28	4768	original
		30	3654	ok
5	H.264	20	8646	original
		22	6806	original
		24	5197	original
		26	3919	ok
	H.265	24	4917	original
		26	3681	original
		28	2759	original
		30	2093	ok
6	H.264	20	5044	original
		22	3726	original
		24	2576	original
		26	1769	original
	H.265	24	2432	original
		26	1705	original
		28	1221	original
		30	917	ok
Continued on next page				

Table A.1: Compression results (continued)

Table A.1 – continued from previous page

Excerpt N°	Codec	CRF	File size [kb]	Visual quality
7	H.264	20	14782	original
		22	11968	original
		24	9345	original
		26	7023	original
	H.265	24	8749	original
		26	6652	original
		28	5033	original
		30	3826	original
8	H.264	20	3259	original
		22	2602	original
		24	2042	original
		26	1593	original
	H.265	24	2006	original
		26	1588	original
		28	1276	original
		30	1054	original