Utrecht University

Geospatial Access To Lifelogging Images in VR
GMT Master Thesis

Author: Kevin Ouwehand (Student ID 6031919)
Supervisors: Wolfgang Hürst & Lynda Hardman

January 30, 2019

Abstract

We demonstrated a proof-of-concept implementation of a map-based system for leisure browsing of geo-tagged lifelogging images in VR. A pilot study was performed, testing quantitative and qualitative aspects by using ten general users as well as the two active lifeloggers who created the dataset that was tested. Our findings show that our map-based approach is useful and applicable to lifelogging data, also due to the high performance of our system, demonstrating its ability to browse very large image datasets in real-time ($n > 50000$). The high entertainment value of our VR system proves our system's applicability for leisure browsing.

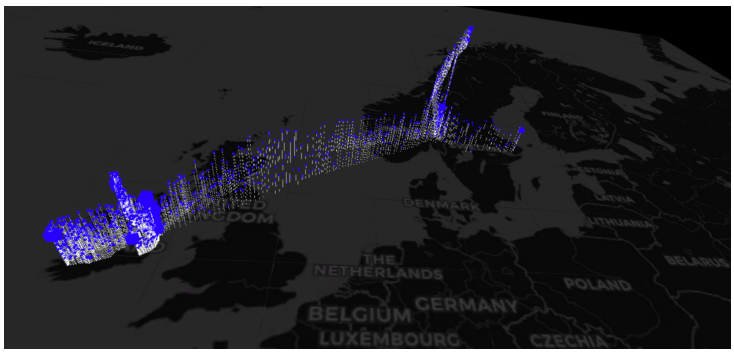

Figure 1: Example screenshot of the program, as seen in VR.

Keywords: Image browsing, Leisure browsing, VR, Geospatial, Geo-tagged, GPS, Lifelogging, Images, Large image sets

Contents

1 Introduction

Over the years, the availability of camera devices and storage capabilities has grown rapidly. This results in users having lots of images, and this is especially true for those who wear lifelogging devices [17]. For lifelogging, people use a small camera attached to their body that takes a picture at a short and regular interval; for example, every 30 seconds. Traditional 2D gallery systems are becoming insufficient to access such image sets, because of their scale as well as their very limited functionality. Lifelogging images also have different characteristics than normal photos. On the one hand, the constant automatic capturing leads to many photos that are, for example, awkwardly framed, blurry, or taken under bad lighting conditions. On the other hand, because they represent one's personal life, people may associate them much more with meta context such as location and time or events when they were taken. In the past 2-3 decades, many researchers have therefore spent their time coming up with various interfaces and systems on how to visualize and work with such large image sets, which will be discussed in more detail in the next section on related work.

However, very few researchers use the potentially infinite space available in VR (Virtual Reality), which seems like a good candidate to deal with the scaling issue of current 2D gallery systems. In addition, anecdotic evidence suggests that many people rarely look back at their old photos. While research has focused on analyzing photo content in order to make collections more structured and thus more easily accessible, related evaluations are often solely focused on performance, but neglect another important aspect that is needed to motivate people to explore their data: using the system should be fun and enjoyable. Thus, there is a clear need for research focusing on photo access systems optimized for leisure browsing and entertaining user experiences.

Alternative interface designs for photo access include map-based visualizations, which are particularly common on smartphones. However, the small screen size of such handheld devices limits the user experience significantly [21, 24]. Our approach addresses this issue by using VR to create an immersive environment with potentially infinite space, that cannot be paralleled on smartphones or any other device with a traditional 2D screen. By using a map-based approach, we aim to address the location meta-context aspect of lifelogging images, as mentioned above. Because lifelogging images are geo-tagged and taken automatically wherever the user goes, the location where they have been taken has an important relevance. For example, people often associate events with locations in their memory. Therefore, we expect map-based interfaces to be particularly useful for lifelogging data.

We present a novel approach for leisure access of lifelogging data that combines the benefits provided by large, immersive VR "screens" with the advantages of a map-based representation of these geo-tagged photo collections. A pilot study demonstrates the usability of our system and supports our claims of its usefulness. The major contributions of this work are therefore:

- The proof-of-concept implementation of a complex VR system that enables users to access lifelogging images via a map-based representation.

  Implementing a map-based system for the access of large photo sets in VR is non-trivial. The complexity of the task along with the magnitude of design options require a careful, well thought out, and optimized interaction design and implementation. In addition, the handling of such huge data sets in real-time results in tremendous challenges for system performance. This is not only true for the thousands of lifelog images, but also the high-resolution map that needs to be rendered at various levels of detail ("zooming"). Our system has been demonstrated at the Lifelog Search Challenge 2018 (LSC 2018 [1]), where it was presented to an international audience [19]. Their general reaction to the system suggested that it does indeed provide an entertaining and engaging user experience, and the smooth operation of the system proves the high performance of our implementation.

- The verification of the system's usefulness and usability via a pilot study involving common users as well as two active lifeloggers.

  Using the data from the LSC 2018 [1], we performed a formal pilot study with ten general users plus two active lifeloggers. These two lifeloggers are also the creators of the LSC dataset, as it is their data being used. Using qualitative and quantitative measures, we gained insight about the systems usability as well as subjective user feedback. Our results indicate that the system has indeed a high entertainment value, and that a map-based approach is a warranted way to represent and access lifelog data.

2 Related work

There are numerous ways to create an image browsing system, but (in general) they all have the shared goal of making datasets (with large amounts of images) more organized and retrievable. So far, most of these systems can be categorized as listed below.

1. Query By Example approach (henceforth QBE)
2. Query By Keyword/Category approach (henceforth QBK/QBC)
3. Hierarchical/clustering approach

We will discuss and compare these categories in the next section. Based on that analysis, we will discuss open research areas and present our novel approach in section 3.

Most papers fall into one of the three mentioned categories. However, some methods also have multiple subcategories. The subcategories that we will use will be determined by the final visualization type, for example: 2D, 3D, or VR.

2.1 CBIR/QBE, Image browsing by example/querying

The largest amount of papers fall into this category: Content-Based Image Retrieval (CBIR). For most papers, this means that they use a QBE approach. A nice starting point is the comparison done by Rodden [25], who did a comparison/evaluation study on various similarity-based interfaces for image browsing. His main findings showed that these interfaces were well-suited when the image features (mostly low level, such as color, etc) used by the system, were beneficial to the task being done by a user. However, these systems started lacking when more high-level features (e.g. annotated concepts) were required to perform tasks, because of the gap between low features, such as average color, and high level features such as image concepts. Furthermore, because of the similarity-based approaches, undirected or 'relaxed' browsing is not really possible.

2.1.1 2D visualizations for CBIR

There are a fair number of papers that address image browsing in 2D. For example, Torres et al. [31] use visual structures to group similar images. This is also a paper that uses querying by example, to come up with related images. The resulting images are presented on rings or spirals, to indicate the distance from the query

example. Results indicate that the approach is not significantly superior, however, users preferred this method over traditional 2D image browsers. This is also one of the few papers to address the problem of overlap, which is a problem that will apply to our VR approach as well (presented in chapter 3). If the set of images is too large, overlap will definitely occur, preventing clear presentation of individual images. This was solved by scaling the relative distances between similar images, which is something that our approach might also benefit from.

Another paper is the one by Rodden et al. [26], where images are positioned on a 2D plane based on their similarity to a certain example image or query. Color was used for the similarity feature, and the images were positioned in such a way that a certain direction from the query image represented a certain change in overall color (e.g. more green). This is another one of the few papers that address the problem of overlap, and it is solved by using a discrete 2D space of cells, and filling in images accordingly. This resulted in slightly more space used for the images, but the resulting overview was more effective in presenting an overview of images.

An interesting paper by Combs et al. [14] tackles the question of whether zooming improves the image browsing experience. Their 2D system has a query section, and a results section. Users can zoom in the results section to determine how many images are visible (and inherently, at what size), as well as zoom in to view one image in full. Their (statistically significant) results show that zooming does indeed improve image browsing, compared to various other image browsers. It can be argued that the same concept of zooming to deal with scaling etc. can be used in 3D, and hence, VR.

2.1.2 3D and VR visualizations for CBIR

An example of a CBIR system that uses a 3D space for graphical representation, is [22]. Again, the user selects an example image as the initial query, and then similar images are returned based on image features such as color, texture etc. A separate feature ranking is used, so the user can give more or less importance to some features. The 3D space is then used to position the similar images based on their similarity with regards to the features of the original image compared to the similar ones, taking into account the feature ranking. Optionally, clustering can be enabled to show clusters of similar images, which can then be expanded to show those images. This is an example of a more relaxed browsing system, instead of one meant for performance based on some tasks. Although no results were presented, the approach of the system is extendable/applicable to VR as well.

Another CBIR system that uses a 3D space for visualization, is the one by Schaefer [27]. A 3D sphere is used to represent the HSV color space, with the rotational axis representing color (hue), and the tilting axis representing brightness (value). A hierarchical system is used to show more images when zooming in. The user can also change the hierarchical tree structure (in real time) to modify the close placement of dissimilar images, as a correction. Selecting a cell fills the sphere again with all images from that cell. The method was tested qualitatively on 4500 images, since, as mentioned before, a standard test set and standard set of tasks does not exist yet. Most test subjects preferred the HSV navigation, but no quantitative results are shown. An unique feature of this paper is their combination of QBE with a hierarchical system. After 3 levels of the hierarchy, potential access to roughly 23 million images is possible, indicating the potential for large scalability (even so because of their O(n) approach for populating the HSV sphere). They even made a VR version [28], but it was merely presented and thus untested.

2.2 Query By Keyword (QBK)

Yee is one of the few who used a multi-faceted approach to create an website-like image browsing system that uses keywords as a search option, instead of an example image [33]. Users can search for images based on hierarchical keywords and categories. Relevant categories and keywords (as well as some images) are shown for the resulting images, thereby giving an overview of the relative structure of the image database. His qualitative results show that his category-based approach is more preferred and flexible than a simple QBE system. It also helped the test subjects learn more about the image database itself (a database with images and descriptions on art). Users preferred his system, even though it was an order of magnitude slower than the QBE baseline system.

Khanwalkar et al. also used a multi-faceted approach, as they introduced a VR system that used a multi-dimensional metadata model to allow for navigating large image datasets, based on various links between images[20]. Images were interlinked in a graph structure by time, location, people, and concepts. The actual browsing system consisted of 2 navigational methods, but for both methods, the images were wrapped around the user, as if inside a cylinder. The first method used the graph structure directly, showing relevant images, as well as the image metadata properties (location, time, etc). Users could navigate the large dataset by using the edges of the graph (the interlinks such as location, time, etc) to navigate to other categories of relevant images (e.g. from a specific location to specific people). The second method used a pre-defined hierarchical structure based on the image concepts (people, sports, etc), and allowed for hierarchical access to the

dataset. The main advantage is that this paper uses a multi-faceted approach for finding (relevant) images (with positive results from users for that aspect), however, they do not use the potentially infinite space available in VR.

2.3 Other image browsing systems

There are more image browsing systems other than CBIR or QBE. Schaefer et al. [23] did a short comparison on various such systems. They identified 2 types of image browsing: horizontal, and vertical. Horizontal image browsing is the navigation within a single plane of visualized images, whereas vertical image browsing is the navigation using a hierarchical or otherwise relational structure. Four more techniques to aid in image browsing were identified: panning, zooming, magnification and scaling. Panning is used to move around the resulting set of images, whereas zooming is used to change the visualized scale of the resulting set of images. Magnification is used to enhance the size of 1 or more images that are shown, whereas scaling is used relatively the same like zooming, by scaling the sizes of all images. Unfortunately, no results were discussed. Schaefer et al. did correctly identify the need for a standard set of images and tasks, as well as a baseline, to assess the performance of various image browsers.

Yang was one of the first to do a direct comparison of a QBE system against a map approach [32]. A hybrid approach is presented where images are presented on a 2D plane. Those images are in fact example images for a QBE system. A self-organizing map is used to cluster the images on the plane. This method tries to tackle the problem of QBE not being an undirected browsing system, as well as the limitation of being dependent on the quality of the resulting images based on the initial query example image. His results showed that his map approach is better than normal QBE, with test subjects finding more images faster, but not with less queries.

Finally, Duane et al. created a prototype VR system for efficiently accessing lifelogging photos, and were the first to create a lifelog access tool in VR, using the same VR hardware that this paper used [16]. Their initial pilot study for that system revealed a very interesting trend: user performance seemed mostly unaffected when comparing their VR system to an almost identical, traditional PC system [15] . Interestingly enough, they also used almost the same dataset that was used by this paper, except it was the previous version of that dataset (NTCIR12 [2] instead of NTCIR13 [3]). Their system presented the lifelogging images based on temporal aspects, and allowed the user to filter the images based on image concepts. Images were presented on a flat wall, and extended in two directions ‘seemingly infinite’ (if enough images were shown). Users could query

the lifelogging data by selecting a date and time range, and some concepts (e.g. car, people, etc). This research is the most related so far, however, it does not use the spatial (location) aspects of the dataset, and it is also not designed for relaxed image browsing.

Furthermore, Marijn Mengerink is, at the time of writing, researching a map-based system for image browsing in general, however it is not yet published. His research includes implementing various types of map visualizations and interactions, as well as evaluating them. Our work is related to his, however we are focusing specifically on lifelogging data, and only use one map visualization and type of interaction. Finally, the most direct relation between our work and his, is the implementation of his system, as our implemented system is an offshoot of his implementation, but in a different direction and with many changes.

2.4 Open research areas

In short, very few papers propose a system usable for relaxed or undirected browsing, as most of them are about performance or accomplishing some task. Unfortunately, no standard list of tasks, and no standard database of images (or baseline data) seems to be available to be used for comparison. The sole exception is the NTCIR lifelogging dataset and tasks that are made available [2, 3].

A handful of papers exist that claim to provide an image browser in VR, however, they are not the Virtual Reality systems that we have come to know of in the past few years. Instead, they use a regular 3D virtual environment, visualized on a normal monitor. Furthermore, of the papers that actually do use VR, none of them use the spatial aspect of image data, and also none focus on relaxed image browsing.

Finally, very few papers use the concept of zooming, even though it is shown that it improves the image browsing experience of users significantly, especially when dealing with large image databases [14]. The concept of zooming can, and should, also be used to retain a sense of overview, such that users do not get lost in the system and the overwhelming amount of images.

3 Research approach

Having identified open research areas and current research limitations, we will improve on this by using multiple aspects from multiple papers. An undirected, 'relaxed' image browsing system will be presented and verified as a proof of concept, where geolocation metadata is used to visualize and position images on a map of our planet Earth. A VR headset will be used to look at the map and the images, improving the immersion and experience of undirected image browsing.

3.1 Map-based approach

Multiple map visualizations are possible, but we stuck to a flat, 2D map, since most people are familiar with the setup and layout of such maps, such as when using physical maps or online variants, e.g. Google maps. Even though this limits our map visualization space to '2D' in VR, it has the more significant benefit that users will not have to learn and adjust for moving in the third dimension, while standing still in real-life and wearing the VR headset. Such a contradiction in presented movement (via the headset) versus actual and/or expected movement by users, is one of the main reasons of VR motion sickness [18]. Of course, motion sickness is not beneficial to user experience, and in order to minimize or even eliminate that, we opted for this type of map.

To navigate around the map, teleportation and zooming out will be used. It will also be used to deal with the issue of scaling, by allowing the user to choose what images will be visible at any time, while still retaining an overview of his position on the map and the images around him. Of all possible (geo-)navigational methods, this one proved the most intuitive and least nausea-inducing when compared to other methods such as flying, when tested by the researchers. This was also confirmed by preliminary testing of Marijn Mengerinks research, as mentioned in section 2.3.

3.2 Dataset

In order to test the system with appropriately labeled images, we use the LSC 2018 dataset [1]. This dataset contains the geo-tagged images that we need for our system, as well as high-level annotated concepts classified from automated computer vision programs (e.g. 'car', 'water', 'airplane', etc). It contains over 45 days of data from two active lifeloggers, and is actually the NTCIR-13 (NII Testbeds and Community for Information access Research) Lifelog dataset [3].

All lifelogging images were taken 45 seconds apart, and contain images of the lifeloggers from the moment of waking up, until going to sleep, resulting in about 1500 images per day at most. All images are also GPS-annotated, meaning that they have either a named geolocation, GPS coordinates, or both. For privacy reasons, the exact GPS coordinates for the images with locations labeled as "HOME" and "WORK" are removed (and thus not used in the system), and all faces on all images are blurred. The dataset actually contains more information than listed here (e.g. biometrics [1, 3]), but it was not used for this system.

Since this dataset contains actual, real-life lifelogging data, it should be reasonably representative of common lifelogging data, and therefore be a representative dataset for our research. Furthermore, to our knowledge, it is also the only dataset available of geo-tagged lifelogging images (excluding the LSC datasets of other years), but it is a 'standard' dataset nonetheless, which could be used to compare our approach to other systems.

However, due to our map-based approach and the relative repetitive nature of most people's lives, the dataset might not contain enough images at different locations or with different contents. The majority of the images will likely be of the lifelogging user performing ordinary every-day tasks, such as eating food, going to work, working etc, which could all be at relatively the same location or close by. This could lead to a very large concentration of images at only a few locations, which would pose a serious limitation for our map-based approach.

When examining the dataset, it turned out that roughly half of the images were labelled as "HOME" or "WORK" for both users, and were therefore not used in the system due to their lack of GPS coordinates. For both users, it turned out that the majority of their images were still in their respective home country, with ordinary contents such as driving to work, eating food, etc. Fortunately, the lifelogging data of the first user was reasonably spread out, having made two trips abroad. For the second user, having only a single travel abroad, a similar scenario was encountered, as most images were labelled as "HOME" or "WORK".

Therefore, this distribution of image locations and the actual image contents may impact the evaluation of our system when testing with local test subjects. Given the complete lack of affinity to the LSC dataset, local test subjects may or may not enjoy the system as much as the actual owners of the dataset, thus impacting the experience negatively.

3.2.1 Clustering

When dealing with lifelogging data, it is likely that many images will be at similar locations, as mentioned in the previous section. Therefore, the actual dataset is put in a cluster hierarchy, to handle these scaling issues for such a large dataset. By clustering images that are close by, we limit the number of actual image locations visualized on the map, and allow for fine-grained access. This cluster hierarchy contains 5 levels, listed below, and is also depicted in image 2, from root to bottom:

1. Base cluster, containing all images.
2. Clusters based on location, containing all images at a certain location and those close by.
3. Clusters based on day, containing all images at a certain date.
4. Clusters based on hour, containing all images at a certain hour.
5. Clusters based on intervals of ten minutes, containing all images in a 10 minute interval.

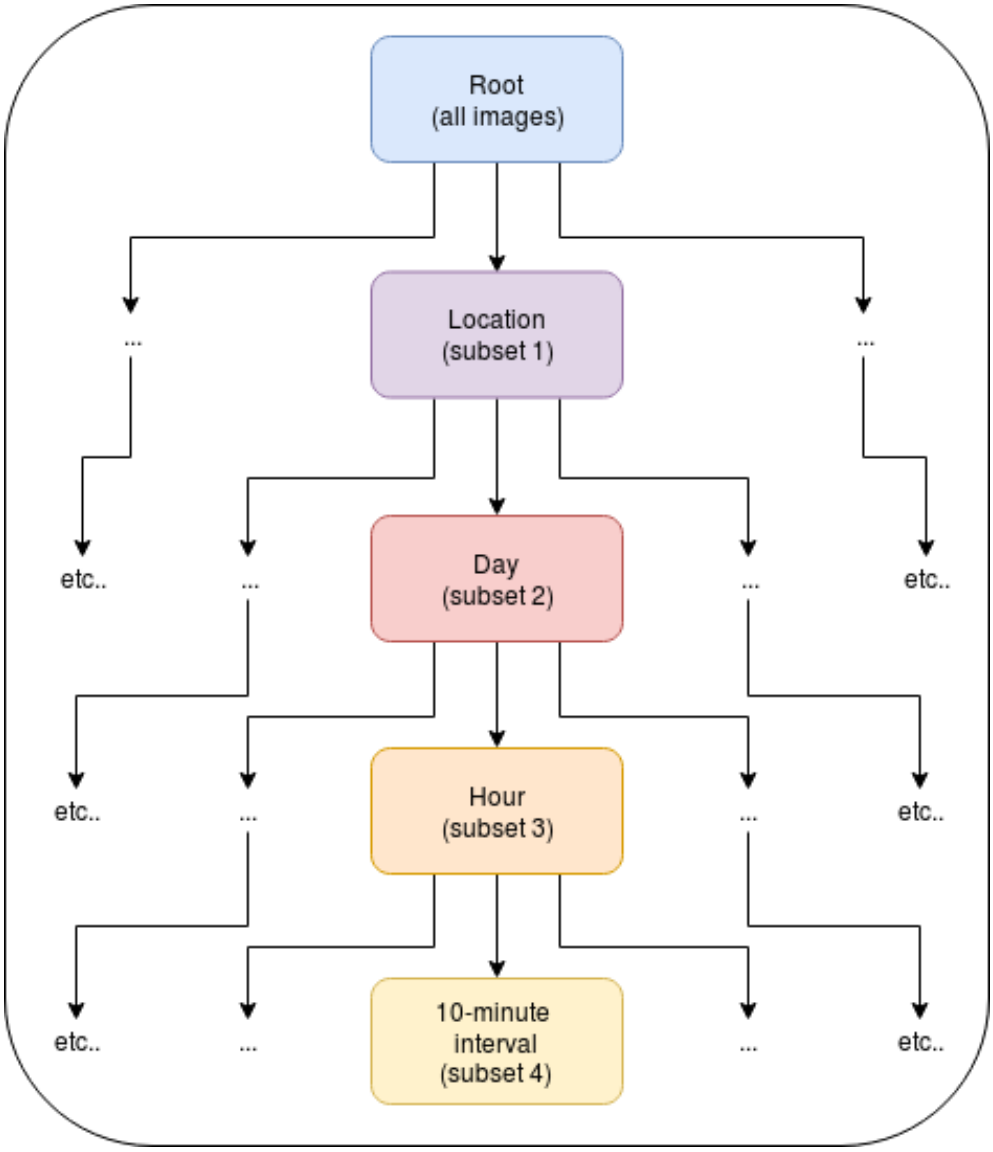

Figure 2: Overview of the clustering hierarchy, indicating that each cluster can have multiple child or subclusters, or even none. The size of each subcluster (the number of images) is at most the size of the parent cluster. Also, images in child clusters cannot contain images that are not present in its parent cluster, so subset 2 $\subseteq$ subset 1, etc. Of course, if a cluster has 2 child clusters, then the size of the two child clusters are always less then that of the parent cluster.

These clusters are created together in a hierarchy tree, and filled with the LSC dataset based on the metadata. As an example, consider that a cluster C1 has 1000

images at a certain GPS location L. That cluster C1 can have 2 child clusters C2A and C2B (thus C1 contains two days of images), e.g. 400 images for C2A and 600 images for C2B, with their sum yielding the original 1000 images of C1. Then, cluster C2B, having all images at date B (and thus also at location L), could have 3 more child clusters C3A, C3B, C3C. Therefore, C3A has e.g. 200 images at a certain hour of C2B's day at C1's location L, and so on. So, further down in the hierarchy, less images are returned, but they will be more specific. The higher up, the more images will be returned, but they will be less specific. This hierarchy can be used to cluster, partition and thus navigate large sets of images easier. However, in this research, due to time constraints, only the first two levels are visualized in the system (location and date clusters).

3.2.2 Filtering

Finally, in order to further narrow down the dataset, (visualized) images can be filtered based on high-level concepts (e.g. people, food, landscape, etc), to show the images that users want. These filters will be the high-level annotated concepts provided by the dataset (see 3.2). A filtering menu will be created that will be used to filter the dataset, and will be explained in more detail in section 4.5.3. A novel interface/system will be created that will use the HTC Vive and respective controllers to perform this undirected image browsing approach.

4 Implementation

In order to turn our research approach into an actual system, various steps needed to be taken to create the system. A high-level overview is given in figure 3, and each component as well as the overall implementation will be discussed next.

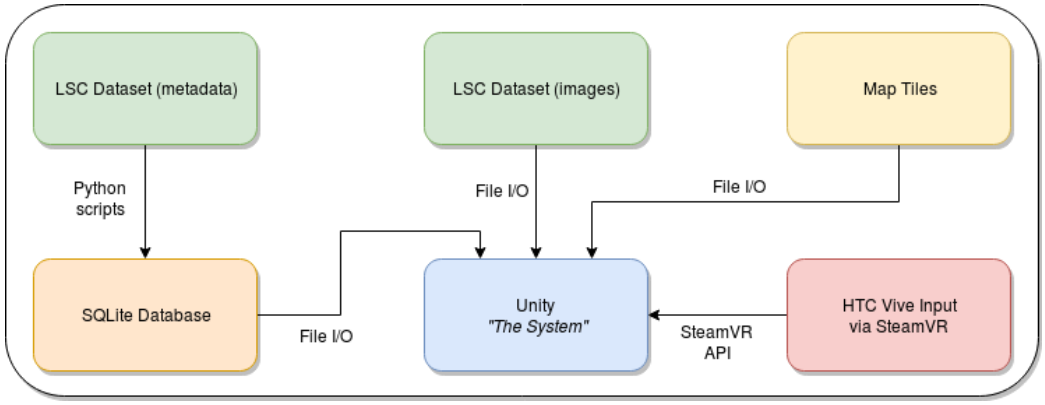

Figure 3: High-level overview of the system.

As mentioned in section 2.3, our implementation is a direct offshoot from Marijn's work. The most notable changes include:

- The change from a web-based Flickr database of images, to local geo-tagged lifelogging data, by using a SQLite database (green and orange blocks in figure 3).
- The optimization of (rendering) performance and user interaction.
- The addition of an extensive, dynamic filtering menu.
- A dynamic map system (yellow block in figure 3).
- The improved visualization of images, by using an image wall.
- The clustering of images, by creating a hierarchy of clusters.

4.1 LSC Dataset

The dataset used in the program is the LSC 2018, or NTCIR-13 dataset as mentioned in section 3.2. It is divided into 3 parts; the first part contains the actual image files, the second part contains an XML file with all the metadata, and the

third part contains the detected concepts. These parts will be discussed in the following subsections.

4.1.1 Images

The dataset contains 110782 images in total, for two lifelogging users. The first lifelogging user (u1) has 90311 images, and the second user (u2) has 20471 images. Of those 110782 images, only 56450 have GPS coordinates, so the remaining 54332 images are not used in the program. That means for user 1 and user 2, only 42255 and 14195 of usable images are left, respectively.

Each image of user 1 has a resolution of 3264x2448 pixels and is encoded as a JPEG image. Each image of user 2 has a resolution of 768x1024 pixels, and is also encoded as a JPEG image. For u1, the images span from August 8th, 2016 until October 5th, 2016, and there are no days without images (but some days have more or less images than others). For u2, the images span from September 9th, 2016 until October 11th, 2016, again without missing days. This part corresponds to the green block, labeled 'LSC Dataset (Images)' in figure 3.

4.1.2 Metadata

The metadata XML file that is included has more information on the images, and contains the following meta information:

1. Music listening history.
2. Biometrics information 24/7 (heart rate, calorie burn, steps, etc).
3. Blood pressure, measured daily in the morning before breakfast and exercising.
4. Blood sugar levels, measured daily in the morning before breakfast and exercising.
5. Semantic locations visited. Used to name the locations that the lifelogger went to.
6. Exact locations visited from GPS coordinates, denoted as latitude and longitude values.
7. Physical activities (e.g. 'walking', etc).
8. Daily mood, according to Thayers 2 dimensional modal of mood [29].

9. Diet log, manual logging of photos of food.
10. Computer input via keyboard and information consumed, per-minute (filtered).

Of all that metadata, only items 5 and 6 are used. The metadata file also contains an organization (per lifelogging user) per day, per minute, to indicate what images were taken at what time, and at what location: either named, or with GPS coordinates, or both. This is used to annotate images with a date and time, location, and to assign them to user 1 or 2. All images that lack an exact GPS location with latitude/longitude coordinates were omitted from the system, as it is impossible to place them on the world map while this information is lacking. The only images that actually lack GPS coordinates, are the ones for which the GPS location is named 'home' and 'work', and they are omitted because of privacy reasons. This part corresponds to the green block in figure 3, labeled 'LSC Dataset (metadata)'.

4.1.3 Metadata Part 2: Concepts

Furthermore, there are a total of 633 unique tags or concepts ('car', etc) detected by computer vision programs, provided by the dataset in the form of a CSV file. The 5 most occurring tags are mentioned below:

1. 'indoor', with 53925 occurrences
2. 'wall', with 24746 occurrences
3. 'person', with 22089 occurrences
4. 'computer', with 14539 occurrences
5. 'laptop', with 10893 occurrences

Only 34146 images (roughly 60%) have tags associated with them, so the remaining 22304 images do not have such computer-detected tags. There are 91 tags that occur only once, and in order to limit such infrequently occurring tags, tags that occur less than 10 times are omitted from the system. This leaves us with 'only' 333 tags (slightly more than half of the original amount) to use for the filtering aspect. All images without tags are not shown in the program by default, but they can be made visible again via the filtering interface (more on that later in section 4.5.3). This part corresponds to the green block in figure 3, labeled 'LSC Dataset (metadata)'.

4.2 Conversion to SQLite database

Using Python3, scripts were written to read and parse the metadata XML file, as well as the concepts CSV file. Using these scripts, SQL statements were generated and saved to a temporary file, to create and populate a SQLite3 database [7]. Using the SQLite3 program, these statements were read from that file and the actual SQLite3 database was created. Note that the actual LSC images are omitted from the database, but read from disk instead, as illustrated by the orange block labeled 'SQLite Database' in figure 3.

This database is used as an intermediate metadata-representation between the LSC data format and organization, and the main system in Unity. Unlike other database systems such as MySQL or PostgreSQL, SQLite does not need a running server, and runs entirely from the local database file only, which simplifies the end-system significantly. The final output of this conversion step is a SQLite3 database file containing 3 tables: one for the images, one for the tags, and one for the mapping between images and tags.

4.2.1 Table schematics

The 'Images' table presented in figure 1, contains the general image metadata, such as the path to the image file, what user it belongs to, and what date and time it was taken. It also includes the GPS location with coordinates, and optionally a named location. The image_id is a primary key, and is later used as a foreign key to relate tags to images.

image_id	path	user_id	loc_lat	loc_long	loc_name	date_time
1139	u1/2016-08-11/20160811_125340_000.jpg	u1	53.2890118	-6.2002897	Starbucks Stillorgan	2016-08-11 12:53:40
etc	etc	etc	etc	etc	etc	etc

Table 1: Table showing the SQLite schema for images, including an example image.

The 'Tags' table presented in figure 2, contains a list of all tags, with an unique id per tag, as well as how often a tag occurs. The tag_id is a primary key, and is later used as a foreign key to relate tags to images.

tag_id	tag_name	tag_occurrences
1	indoor	53925
3	person	22089
9	ceiling	6233
etc	etc	etc

Table 2: Table showing the SQLite schema for tags, including 3 examples.

The last table presented in figure 3, contains the n:m cardinality mapping, assigning tags to images.

image_id	tag_id
1139	1
1139	3
1139	9
etc	etc

Table 3: Table showing the SQLite schema for tags assigned to images, including 3 examples.

4.3 Map Tiles

In order to create the map on which to place the images, a tile map was created. The map uses the Spherical Pseudo-Mercator projection (also known as Web Mercator) with equi-rectangular (square) tiles for simplicity, since they are widely available as well as easier to implement than other systems [6]. This projection system uses the assumption that the Earth is modeled as if it were a perfect sphere. The main reason for this projection system is to significantly reduce the complexity of the computation of tile coordinates, at the cost of having less accurate aspect ratios further away from the Equator, resulting in a fast and sufficiently accurate map representation.

A tile map works by using x and y coordinates for the tiles, and a zoom parameter to indicate the level of zoom (and detail) of the map. Tile maps use individual tiles (images), and combine multiple tiles into a final image that represents a map, or a portion of it. Given the fact that (usually) tiles have fixed image resolutions, multiple tiles are used to create maps with more detail. In our case, the fixed resolution is 256x256 pixels, as the tiles are provided by an external tile provider. At zoom level $z = 0$, only one tile with coordinates $x = 0$ and $y = 0$ is available,

and that single image contains the map of the whole world. Given the fixed tile resolution, that single tile image is thus not very detailed, see image 4.

Figure 4: The single map tile at zoom level 0.

The actual tiles were downloaded from [5], by using a Python script to download them automatically and save them to disk. The mentioned URL contains placeholders, meaning that they need to be filled in to produce a working URL. There are 4 servers (a-d) that serve the tiles, so the following URLs are valid:

- http://a.basemaps.cartocdn.com/dark_all/0/0/0.png
- http://b.basemaps.cartocdn.com/dark_all/6/31/32.png

4.3.1 Zoom levels

Each subsequent zoom level increases the number of tiles by 4, so zoom level $z = 6$ already has 4096 tile images (x and y range from 0 to 63), taking up 4.8 MB of disk space. At $z = 10$, there are 1048576 tile images (x and y range from 0 to 1023), taking up a bit more than 1 GB of disk space. Since each subsequent zoom level increases the number of tiles by 4 (and thus the storage requirements for all those images), the maximum zoom level that is used for the system is

capped at $z = 10$, so all levels combined take up 1.5 GB of disk space in total. With the current size of the map in the system (in VR), higher zoom levels are not necessary, as the map is not large enough (in VR) to warrant the extra increase in disk usage and computation, as the added detail is barely to not visible. Figure 9, 7 and 10 demonstrate the map at various zoom levels, as seen in VR.

Initially, the system worked at a fixed map size and zoom level of $z = 6$, meaning that the entire map had a fixed level of detail (and thus quality/resolution). Moving too close to the map revealed the individual pixels of the tiles, therefore a dynamic system was created (more on that later in section 4.5.2). Ideally, the system would use a quadtree approach to load and unload images of higher/lower zoom levels, to provide a dynamic map that changes its level of detail (and thus quality) based on the users distance to those tiles. However, due to the nature of the already complex system, as well as time constraints, this approach was not used. Instead, the dynamic map system keeps the map fixed at 4096 tiles ('simulating' the original $z = 6$), and instead swaps out individual tiles based on the users distance to those tiles. To facilitate this, Python scripts were written that would combine multiple images from higher zoom levels ($z = 7$ and higher) into the required 4096 tiles for the simulated $z = 6$. Likewise, images from lower zoom levels ($z = 5$ and lower) were split into multiple images to form the required 4096 tiles. This approach changes the final resolutions of the tiles at different zoom levels, with (the merged) higher zoom levels containing tiles of higher resolutions, and (splitted) lower zoom levels containing lower resolution tiles. However, this is not an issue, as these tiles are loaded as dynamic images into Unity textures anyway (see section 4.5.2).

4.4 HTC Vive and SteamVR

The HTC Vive was the hardware used to interact with the VR system, and the SteamVR API allowed us to interact with that hardware, which can be seen in figure 5. It includes 6-axis (positional and orientational) tracking of the two controllers and the VR HMD (Head-Mounted Display, also known as the headset), using 2 basestations. The basestations work together wirelessly to ensure correct tracking of the devices. The controllers feature a touchpad, a menu button, a Steam button, a trigger, and a grip button, as well as haptic feedback. The headset has two 1080x1200 resolution screens per eye, for a total of 2160x1200 pixels, and a 90 degrees Field of View (FoV), at a refresh rate of 90 Hz. The latest SteamVR driver was used to run the Vive (at the time of writing, version 1.1.4).

Figure 5: The HTC Vive set, including 2 basestations, 2 controllers, and the headset. Image taken from ArsTechnica at https://arstechnica.com/gaming/2016/10/best-vr-headset-2016-psvr-rift-vive/.

In order to interact with the program, the various buttons and their names are visualized in figure 6.

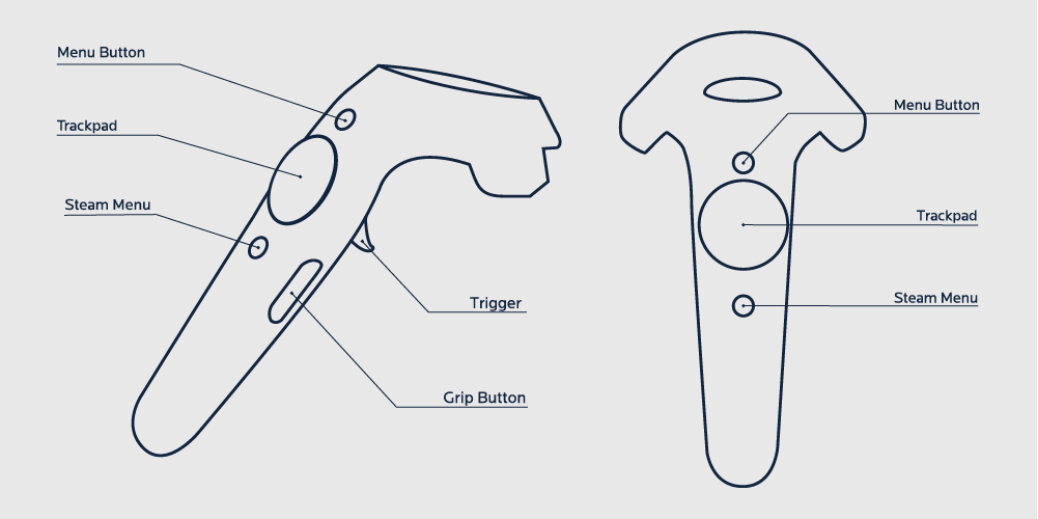

Figure 6: The buttons on the HTC Vive controller. Taken from https://survios.com/rawdata/content/themes/rawdata/assets/img/vive-userguide-white@2x.png

4.5 Unity

The actual program in which our system was created, is Unity (Personal edition, version 2017.3.1f1) [8], and C# scripts were used to program the required functionality of the system. It uses all previously explained components, as can be seen in figure 3. The main project includes a single scene, with a few plugins to ease development, listed below:

- SQLite plugin for interacting with our SQLite3 database [7].
- SteamVR plugin for interacting with the HTC Vive and controllers [10].
- Listview plugin for creating and managing list views, used for the filtering menu [9].
- TaskParallel plugin for managing C# threads, used for loading images in background threads [11].

At startup, the program loads in the SQLite3 database created from the LSC metadata (see section 4.2), and then does the following things:

1. Create a list of Tag objects from the 'Tags' table. This step also filters out tags that occur less than 10 times.
2. Create a list of Image objects from the 'Images' table.

3. Assign each Image object the appropriate Tags.
4. Calculate the GPS upper and lower bounds of all images..
5. Create a hierarchy of clusters based on the Image data, as explained in section 3.2.1.
6. Create pins for each location cluster, to position them on the map.
7. Create the actual map, and place the previously created pins on them.
8. Initialize the VR environment, position the user above the center of the dataset on the map, and run the program.

At step 7, the map is also cut off, based on the results of step 4. An extra 20% of the bounds are added, and the map is scaled to be larger or smaller, based on how large the final bounds are. This is done to ensure that maps do not have lots of empty space, and are thus content-dependent. So, if the dataset is very spread out, the 'physical' size of the tiles of the map are smaller in VR. If the dataset is very dense, the 'physical' size of the tiles of the map are larger in VR.

4.5.1 Image access

When the program is running, the user is positioned above the map, at roughly the center of the dataset. The user is then presented with an overview of the data, as can be seen in figure 7.

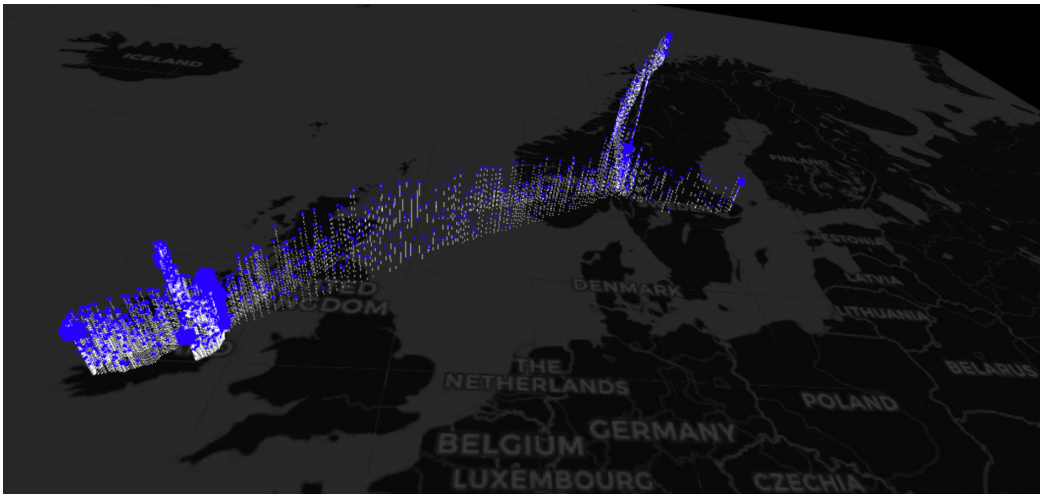

Figure 7: Example screenshot, showing an overview of the images (blue pins) positioned on the map, as seen in VR.

Each blue pin represents one location cluster, and each location cluster contains 1 or more days of data (and thus 1 or more day clusters). Since a lot of images are very close-by, their GPS locations differ only by the last few digits. In order to reduce the number of visible clusters/pins, these GPS coordinates are rounded down, to group images at relatively close-by locations. In our system, they are rounded down to 2 digits, yielding an accuracy of up to 1.1 km (see appendix section 10.1). This seemed an acceptable trade-off between accuracy and number of clusters given the detail and scale of the map, which is roughly city-level.

When moving closer to those blue pins, they gradually change into image billboards, showcasing an image at that location, and the billboard will rotate towards the user so they are always visible, if in close proximity. These images are loaded using background threads, and managed by the TaskParallel plugin [11].

If the user aims his controller at one of the blue pins, and presses the trigger button, the images at that location are retrieved, and presented in an image wall around the user's controller, as can be seen in figure 8. The user can then navigate through the images by using the touchpad. Each row of the image wall represents one day of images at that location (the date is displayed below each row), and the actual image details are displayed below the main, central (selected) image.

Figure 8: The image wall with various images per day, presented when the user grabs the images from a blue pin or image billboard.

4.5.2 Map and geospatial navigation

When the program is started, the map is created, and all tile images are loaded at the initial zoom level $z = 6$ (so, 4096 tiles/images). After the program is fully

running, the distance of the user to each tile is calculated, but only a few tiles per frame are calculated, to ensure responsive frame rates. If the quality of tiles needs to be changed, this is done in a separate coroutine, as the Unity API is not thread-safe, and actively blocks non-main-thread calls, so this cannot be done efficiently in a background thread unfortunately. Figure 9 and 10 show the map at its lowest ($z = 3$) and highest level of detail ($z = 10$) respectively.

Figure 9: The map of the system, showcasing the lowest zoom level at $z = 3$ and thus the lowest level of detail. Notice how only country names are readable.

Figure 10: The map of the system, showcasing the highest zoom level at $z = 10$ and thus the highest level of detail. Notice how individual city names, and even smaller ones, are easily readable.

In order to navigate around the map, 2 methods are available. The first method is 'horizontal' navigation, and allows the user to 'teleport' anywhere on the map. When clicking and holding down the touchpad, a white cylinder appears, and the user can aim the controller to move the cylinder (see figure 11). When letting go of the touchpad, the user slowly flies towards the selected location. The white cylinder also scales with its distance to the user, so the user can get a sense of the distance to his destination.

The second method is 'vertical' navigation, and allows the user to fly upwards, to get a better overview of his position on the world map. This is done by pressing and holding the grip buttons, and it stops when the user lets go of the grip buttons. The speed at which the user flies upwards starts low, and then grows linearly with time, to ensure easy acclimation and no motion sickness.

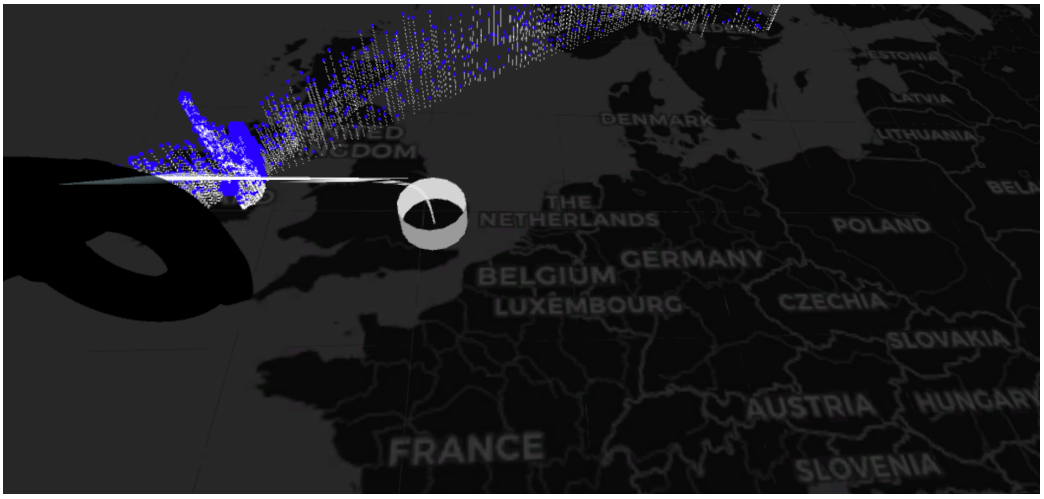

Figure 11: When clicking and holding down the touchpad, the user can teleport around the map.

4.5.3 Filtering

Last but not least, the images in the dataset can be filtered, by enabling or disabling certain tags (concepts). For this, a filtering menu was created that uses the listview plugin [9]. The filtering menu consists of 3 parts, for easy filtering operations. Figure 12 shows the filtering menu on the left controller.

The middle menu has the list of all tags shown, along with how often they occur, as well as whether they are active or not. The left-most menu has a list of A to Z, indicating the first letter of the tag to filter on. When clicked, it will select the first tag with that letter (based on the sorting method used), so the user can easily search for and find certain tags.

The right-most menu has special options, and includes an option to enable or disable all tags at once. Also, all images without tags can be made visible or hidden in this menu. Finally, the user can change how the middle menu is displayed, by changing how the tags are sorted.

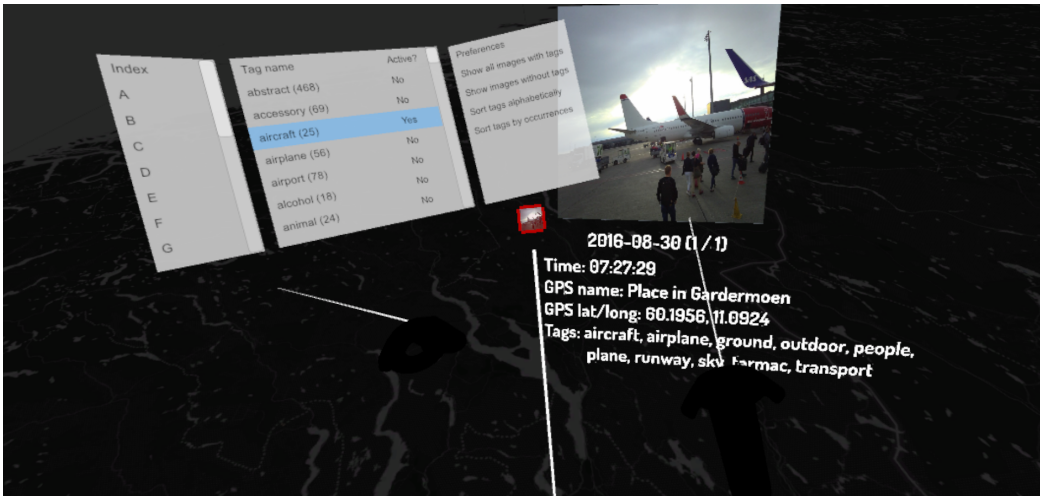

Figure 12: Example result of using the filtering menu, which is shown when the menu button is pressed. The left controller 'holds' the filtering menu, and the right controller has grabbed an image. It also shows an image billboard in-between the controllers, with its thumbnail being the same image as the right controller has grabbed.

5 Evaluation

In order to examine the usability of this system as an entertainment-oriented (undirected, relaxed) image browsing system, various aspects were evaluated. The major focus is thus on exploratory search, and not on performance oriented (search) tasks. The complete experiment setup consists of the following steps:

1. Pre-experiment actions (questionnaire and signing of consent form).
2. Qualitative testing.
3. Optional break.
4. Quantitative testing.
5. Post-experiment actions (survey and final questions).

For the evaluation, only the data of user 1 of the dataset will be used (see section 4.1.1, as they are more spread out over the world map, and also contain more interesting images. When interacting with the program in VR, test subjects were standing for the entire duration of the experiment, which lasted about 30 minutes in total, per test subject. Each part of the experiment will be explained in more detail in the next sections.

5.1 Pre-experiment actions

Before the experiment starts, the system and its purpose was explained to the test subject. Also, the purpose of the experiment was explained, emphasizing that the system needs to be tested, and not the test subject. It was mentioned that taking a break, or stopping completely was always allowed. Since (VR) motion sickness is always a possibility, test subjects needed to sign a consent form (included in the appendix, section 10.2). Then, the following questions were asked:

1. Age, in years.
2. Sex, either Male, Female, or Unspecified/Rather not say.
3. Whether the test subject is left or right handed.
4. Whether the test subject has any eye deficiencies, such as glasses.
5. Test subject's experience with VR, options are:
   (a) I have never used VR.

(b) I am familiar with it, e.g., tried it out a few times, but do not use it normally.

(c) I occasionally use VR, e.g., a couple of hours per month.

(d) I use VR often, e.g., more than 10 hours per month.

6. Test subject's experience with (online) map systems such as Google maps, options are:

   (a) I have never used them.

   (b) I am familiar with it, e.g., tried it out a few times, but do not use it normally.

   (c) I occasionally use them, e.g., a couple of times per month.

   (d) I use them often, e.g., more than 10 times per month.

7. Test subject's experience with systems for image access, browsing and retrieval (e.g., their own digital photos), options are:

   (a) I have never used them.

   (b) I am familiar with it, e.g., tried it out a few times, but do not use it normally.

   (c) I occasionally use them, e.g., a couple of times per month.

   (d) I use them often, e.g., more than 10 times per month.

This data was gathered to find out if certain groups of users (e.g. experienced with VR but not with image browsing systems, or the other way around, etc) find our program more or less enjoyable than others. It is also known that people with glasses enjoy VR less than people without glasses, so we expect to see that same bias in our results as well [12]. The results might also indicate that although they enjoy VR less, they might find our program intuitive and enjoy it relatively the same nonetheless. It was mentioned on the form that this data will solely be used for the purposes of the experiment and deleted afterwards. As mentioned, the purpose of the experiment was to test the system, and not the test subjects. Therefore, they can take a break or stop at any time if they desire. If they wish to continue the experiment, time spent taking a break will not be considered part of the time it takes them to perform a task. If they did not fully finish the experiment, they will have their test results excluded.

5.2 Qualitative testing

Instead of only raw, numerical, quantitative data, we can use qualitative analysis to investigate the subjective experience of users. This complements the quantitative analysis, since low performance can also have an impact on experience, and vice versa. To aid in this aspect, the computer screen, showing what the test subject sees on his/her headset, will be recorded. These recordings will only be analyzed manually, to find possible explanations for anomalous data, which cannot be explained normally, for example: Why did it take user X so much longer than others to find image Y or similar images? By looking at that video, we could find that the user spent some time re-orienting himself, or was just looking at random images instead of searching for the one that was requested.

The first, qualitative part is included with the purpose of getting users familiar with the system and learning how to use it, at their pace. It is also a vital moment to find out how intuitive the system and controls are. Also, because our system is more designed for leisure and exploratory search, and less for performance-focused targeted searches, qualitative statements on how people experience it, how they like and enjoy it, are essential in verifying its usefulness.

5.2.1 Explanation of the system and controls

First, the controls and features of the system are explained and demonstrated, so that the user knows them. They can ask the test administrator at any time for them in the next steps, so that they can get to know the system and work with it to the best of their ability. Then, the VR headset is put on, and the controllers handed to the test subject.

5.2.2 Free-roaming

Then, the user is free to browse and explore the dataset for 5 minutes. This part is subdivided further into 2 phases. The first phase (2-3 minutes at most) is to get the user to familiarize himself with the controls, and the second phase is to explore the data (2-3 minutes or more, depending on how long the first part takes). The overall duration should be roughly 5 minutes (minor changes of up to 30 seconds more are no problem). For the second phase, it is suggested to make use of the features of the system to browse the images, with the controls learnt from the first phase.

Two questions will be asked with the intention of having the user examine more than just a handful of images, and to make the user think about the data and how it is organized on the map, as well as how possible filtering options might make sense here. The answers to these questions do not need to be noted as these results are not part of the actual test, and are used only to give the test subject a hint on what to look for and how to do it. These questions are:

- The owner of the lifelogging images traveled abroad twice, which 2 places did he visit?
- The images are positioned based on their GPS coordinates. There should be ”gaps” or ”empty spots” on the map where there are no images, because they are labeled as ”WORK” and ”HOME” and have no GPS coordinates attached due to privacy reasons. Where do you think these 2 gaps are, so: where do you think the owner works and lives?

After 5 minutes, the system is restarted by the researchers (to start with a clean slate again), and the user is allowed an optional break of 1-2 minutes if he/she wishes it.

5.2.3 Quantitative testing

To gather quantitative feedback about the usage of the system, test subjects had to perform the following four tasks of the LSC 2018 challenge [1]:

1. Find the moments when I was looking at an airplane (and not sitting in one).
2. Find the moments when I was walking by the sea and taking photos.
3. Find the moments when I was eating lunch.
4. Find the moments when I was making juice using fruit and/or vegetables.

As can be seen from the tasks, only task 1 and 2 have a location aspect, and task 3 and 4 do not. The first two tasks were manually selected from the LSC challenge because of their location-related aspect. The last two tasks were also manually selected, because they seemed most representative of typical lifelog search tasks (compared to other non-location-related tasks). The order of these tasks will be random for each test subject, so that the overall task performance is not influenced by performing the other tasks (and thus getting more used to the system).

The original LSC challenge featured 24 tasks, but many of them were of similar nature, and these 4 were the most fitting for our research. Also, user 1, who has

created more than half of the LSC dataset (see section 4.1.1), created these 24 tasks originally, including the four mentioned above.

We expect to see more images found with the first 2 tasks than with the second 2 tasks, as the location aspect of these first 2 tasks can aid the user into finding more images within the allotted time because of the map-based approach of our system. Users will get up to 1 minute per task to find the correct images (the time limit is automated and enforced).

5.2.4 Tracking of interactions

Per task, the number of different actions that the user takes will be kept track of by the system, with those being:

- Clicking on a pin or a billboard to get its stack of images counts as 1 action. Letting go of a stack does not count as an action.
- Teleporting somewhere else, each teleportation act counts as 1 action. Also, the teleportation distance is recorded.
- Enabling or disabling a tag/concept to filter the images. Using the special case of enabling/disabling all tags counts as only 1 action. Re-ordering the list of tags does not count as an action. Also, what filters are enabled/disabled is recorded, so we can track what filters were used by the test subjects.
- Navigating through the image stack, left/right/up/down, each click counts as 1 action for this case. Holding down the button to navigate faster through the stack, will count as if it were many separate clicks (and is thus not treated as special). Also, the number of images visited will be kept track of, both unique and total.

5.2.5 Tracking of time

Along with those actions, per category, the time spent interacting with the system will be kept track of:

- Time spent idling on the map, and likely just looking at pins/billboards (detecting this looking at part is hard and out of scope for our research purposes).
- Time spent geo-spatially navigating the map using teleportation. Physical movement of the user will not be kept track of as this is much harder to

determine; e.g. is the user actually moving, and thus stepping away, or simply tilting his body slightly? Again this is out of scope for our research purposes.

- Time spent navigating images using the image wall.
- Time spent using the filtering interface.

Given our unique approach/system design, its possible that the user can hold a stack of images with 1 controller, and then teleport somewhere far away with the other controller, after which that controller can be used (while teleporting/moving) for the filtering interface. Since it is not our goal to determine if such methods are used by the user, we simply keep track of the time spent per category, so the cumulative tracked time spent for a task may well be over 1 minute.

5.2.6 Motivation of tracking

Summarizing, for a single test subject, the result for a single task looks like this (using example values):

- Actual result of the test, number of correct images found: 8.
- Total number of unique images visited: 41, total: 57.
- Number of pin/billboard clicks: 7.
- Number of teleportations: 4, total distance 231.3.
- Number of filter operations: 2, tags used: all, food, airplane.
- Number of image wall navigations: 56.
- Time spent idling on the map: 15.6 seconds.
- Time spent navigating the map: 3.2 seconds.
- Time spent navigating the image wall: 45.4 seconds.
- Time spent using the filtering interface: 7.1 seconds.

These things will be kept track of with the purpose of measuring the usability of our map-based approach in VR. For all users, these actions and times will be plotted per task in a graph, indicating the average and spread of the values per task. More specific graphs can be made where specific groups are compared to the rest, e.g. as mentioned before (experienced VR/image-browsing users vs non-experienced). This should give us a better understanding of the applicability of a map-based approach in VR with regards to image browsing. Although the system

is not designed to be used as a system for performance search, it can be interesting to see the results, and might reveal missing/wanted features of the image browsing system.

5.3 Post-experiment actions

After all tests are done, a System Usability Survey using a Likert scale needs to be filled in [13]. This will give us an indication of how intuitive the system is, based on the final score from all test subjects. Afterwards, there are some final questions to be filled in by the test subject.

5.3.1 Survey

The System Usability Scale (SUS) survey contains the following questions:

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

The answer scale of questions are as follows:

1. Strongly disagree.
2. Disagree.
3. No opinion.
4. Agree.
5. Strongly Agree.

SUS yields a single number representing a composite measure of the overall usability of the system being studied, per test subject. Scores for individual items are not meaningful on their own. To calculate the SUS score, the score contributions from each item are summed. Each item's score contribution will range from 0 to 4. For items 1, 3, 5, 7 and 9, the score contribution is the scale position minus 1. For items 2, 4, 6, 8 and 10, the contribution is 5 minus the scale position. By multiplying the sum of the scores by 2.5, we obtain the overall value of SU, which can be between 0 and 100.

5.3.2 Final questions

The final SUS scores give us very important numbers indicating the usability of the system, but the interpretation of it must be more nuanced as clear conclusions cannot always be drawn. If our system has a low usability value, it might mean that our system is not good enough, and not that map-based image browsing in VR is a bad idea. To be able to give a better answer on the usability of our system and map-based image browsing in VR in general, a number of final questions were asked to complement and explain this number in more detail:

1. (Open) What did you like about the system and why?
2. (Open) What did you dislike about the system and why?
3. (Closed) Did you experience motion sickness, discomfort, headache, fatigue, nausea, or disorientation? Answers can be chosen from this list:
   (a) None of the above.
   (b) Only a little bit, but it did not have an impact on the overall experience.
   (c) Yes, and it was enough to have an impact on the overall experience.
4. (Closed/Open) When browsing through the system/images, what approach did you use most? Answers can be chosen from this list (multiple answers are possible):
   (a) Browsing based on the map.
   (b) Browsing based on the size/location/density of the pins/billboards.
   (c) Browsing based on filtering tags.
   (d) Something else, namely: To be filled in by test subject.
5. (Open) What would you use this system for?

6. (Open) Any other comments?

These final questions will be used as support for the SUS score and the quantitative measurements. From these (and in combination with the on-screen video), it should be clear why certain results were obtained, e.g.: users did not find the correct images for task X because they did not use the filtering interface correctly or at all, or found it too cumbersome to use it. Some of these will be used directly to explain certain results, and some of them might be noted as anecdotes. This concludes the experiment setup.

6 Results

The system was evaluated using ten test subjects from the University of Utrecht, and consisted almost entirely of local, male students between 20 and 27 years old. Of those ten test subjects, only one was female, and only one superseded the age of 27 (being 56). Furthermore, the program was also tested with both owners ('users') of the LSC dataset, namely user 1, and user 2. Both users underwent the same experiment setup as the other test subjects, as described in section 5. The only difference is that they also explored each others data in addition to their own data, during the free-roaming phase as described in section 5.2.2. In order to avoid confusion, the following terminology is used throughout the remaining sections:

- Local test subjects: Meaning all ten local test subjects from the University of Utrecht. Does not include users 1 and 2.
- Users 1 and 2: The test subjects who are the original owners (and creators) of the LSC dataset, as described in section 4.1.1.
- All test subjects: Meaning all twelve test subjects, so it includes the ten local test subjects as well as users 1 and 2.

First, information about all test subjects is mentioned, showing the results of the pre-experiment questions as mentioned in section 5.1. Then, the quantitative results are shown, showing the task, interaction, and time results, as well as comparing several groups of test subjects. After that, the qualitative results are shown, showing the scores of the SUS survey [13] as mentioned in section 5.3.1. Finally, the results of the open questions are mentioned.

6.1 Information on test subjects

The results of the pre-experiment questions are shown below. Figure 13 shows the age distribution of all test subjects. As mentioned, most test subjects were local students from the University of Utrecht.

The eye deficiencies of all test subjects is shown in figure 14. Even though it was possible to select both 'glasses' and 'lenzes' as eye deficiencies, no test subject had done so, meaning that there is no overlap between those two groups. Given the fact that seven out of twelve test subjects have glasses, we expect them to enjoy our program slightly less than others [12]. Interestingly enough, all test subjects noted their right hand as their dominant hand, as shown in figure 15. When examining the screen recordings, all users used the right controller (in their right hand) dominantly as well.

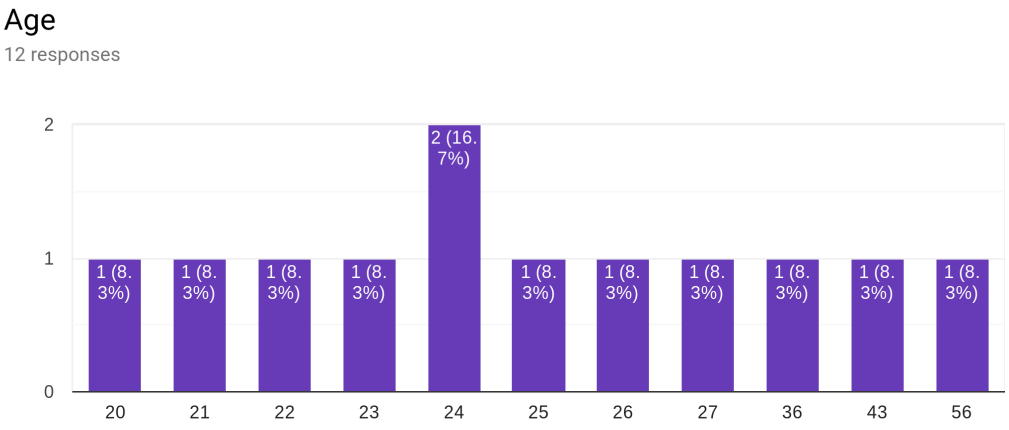

Figure 13: Age of all test subjects.

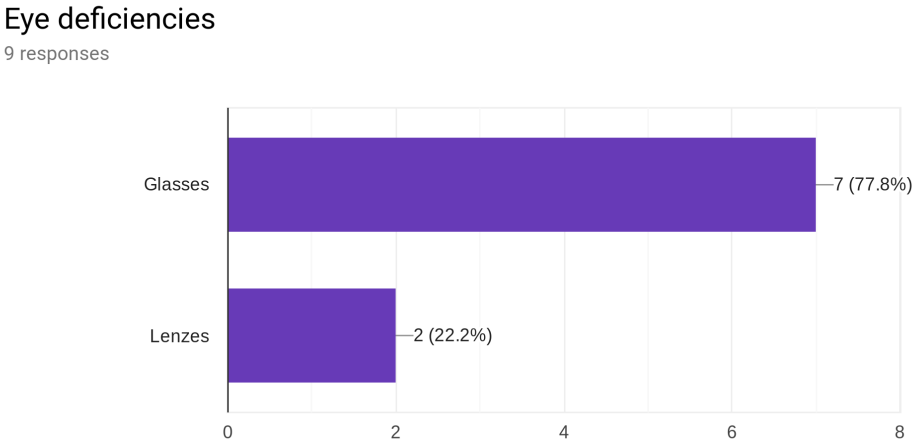

Figure 14: Eye deficiencies of all test subjects. Even though test subjects could select both glasses and lenzes as deficiency, no test subject had done so, meaning that there is no overlap between the two groups.

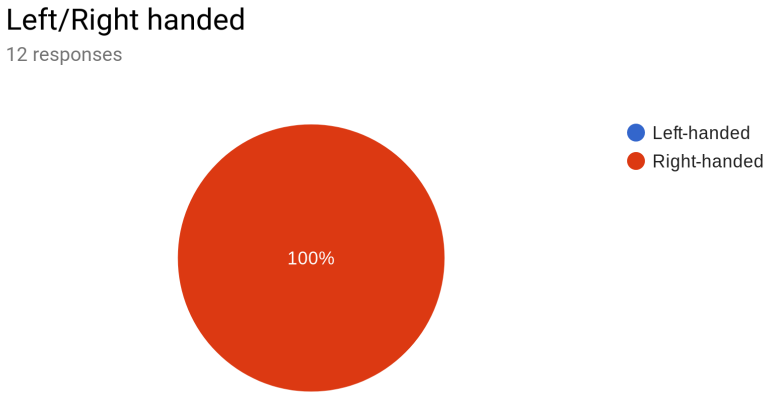

Figure 15: Handedness of all test subjects.

When asked about their experience with VR, two-third of the test subjects answered that they were familiar with it, but do not use it normally (see figure 16). Only two subjects had never used VR before, and none of them use VR often. The experience with map systems of all test subjects is shown in figure 17, and indicates that all users are sufficiently familiar with such systems. Finally, figure 18 depicts the experience of all test subjects with image browsing systems, showing a more divided result between the answers.

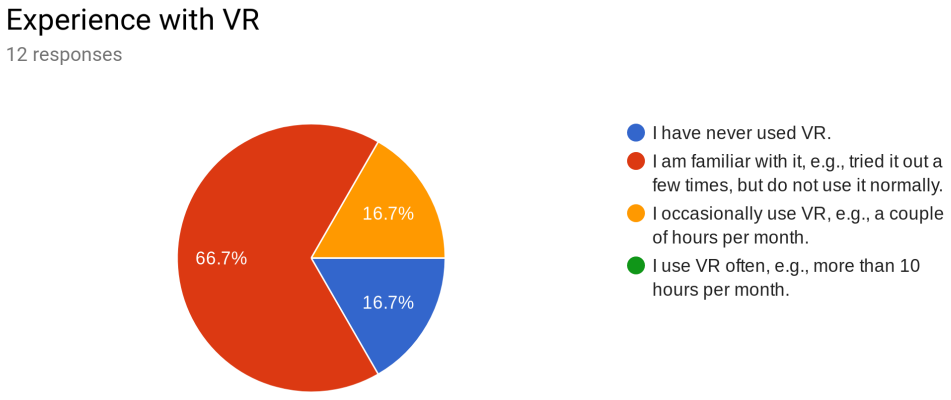

Figure 16: VR experience of all test subjects.

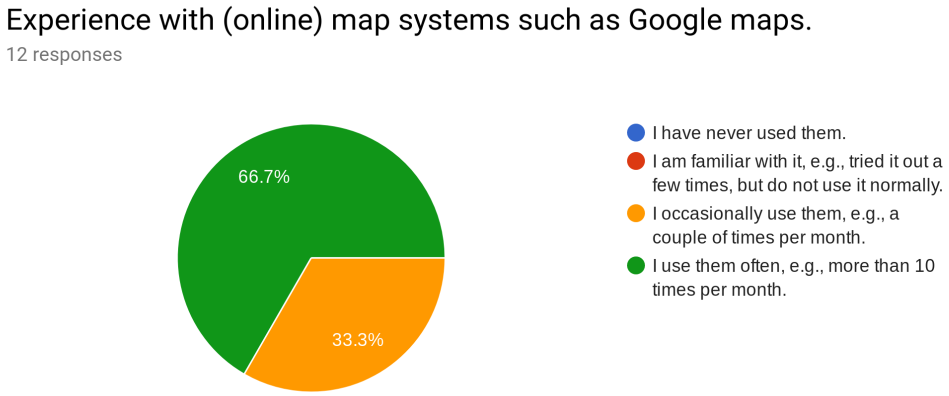

Figure 17: Map system experience of all test subjects.

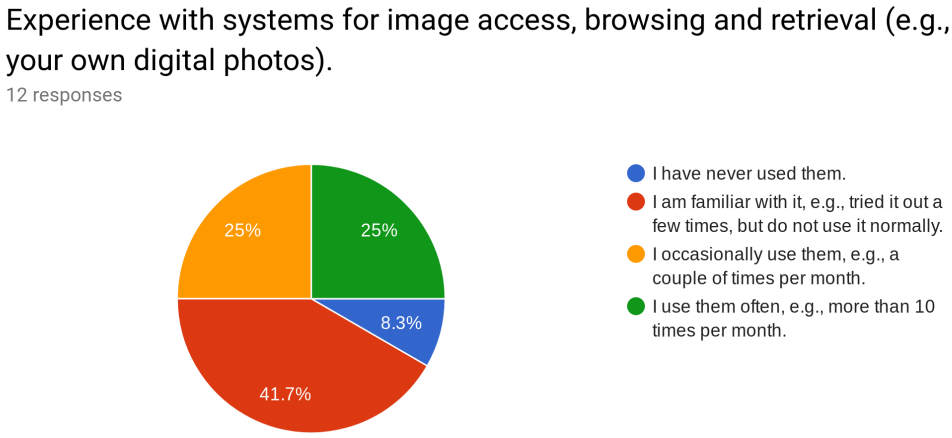

Figure 18: Image browsing experience of all test subjects.

6.2 Task results

For the task results, we will first mention the results of all test subjects, and then discuss subgroups separately. It can be interesting to see the results of users 1 and 2 separately, since both users know each other and may also know about (parts of) each others data. This separation makes even more sense for user 1, since all tasks are based around the data of user 1. f

6.2.1 All test subjects

The average task results of all 4 tasks is shown in figure 19, with error bars indicating their standard deviation.

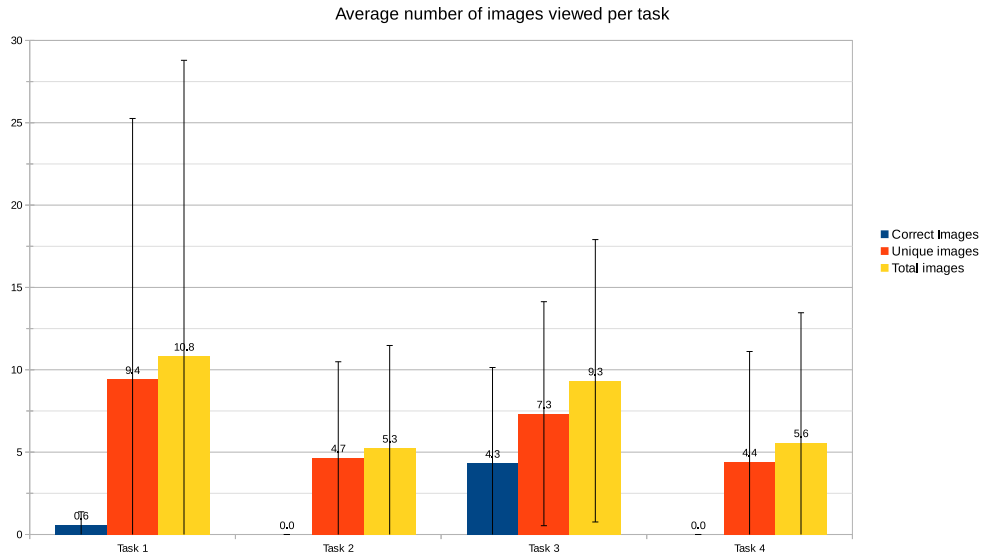

Figure 19: Average number of images viewed per task, for all test subjects. No correct images were found for task 2 and 4. The error bars indicate the standard deviation. Note how high some of these deviations are, compared to their average values, indicating widespread result values per test subject.

Interestingly enough, no test subject found images for tasks 2 and 4 within the allotted time of 1 minute, and the reason for this is two-fold. First, the enforced time limit of 1 minute is very short, as can be seen by the low number of correct images found for tasks 1 and 3, compared to the total number of images viewed. All test subjects so far were either surprised by how fast their time was up and/or remarked that they would like more time. When looking through the video recordings, we observed that some test subjects found more correct images outside of the time limit by only a few more seconds.

Secondly, tasks 2 and 4 were (deliberately) much harder than tasks 1 and 3, as they required more fine-grained filtering or were less present in the dataset than the other images. When looking at the screen recordings, most test subjects found images for task 2 that only met the criteria halfway, as the images showed the

lifelogging user looking at the sea, but not when taking pictures. However, no test subject came close to finding the correct images for task 4.

In order to look for images with certain tags, all test subjects disabled all tags first, and then enabled only the tags that they want. It should not come as a surprise, that the most used tag for task 1 is 'airplane', with seven test subjects using it. Second are 'aircraft' and 'airport', with four test subjects using it. Furthermore, some test subjects enabled both 'airplane' as well as 'aircraft' (three uses). Finally, two test subjects used 'window blinds' and 'clouds' exclusively, but did not find any correct images using these tags. For task 2, the most used tags were 'water' (six times), followed by 'shore' and 'phone' (twice), and 'cellphone', 'sandy' and 'phone (once). However, no correct images were found using this approach, even though most test subjects found images that matched the required criteria only halfway (e.g. only walking by the sea but not taking pictures). For task 3, test subjects almost exclusively used the 'food' tag (nine times), and only 1 test subject used 'vegetable' exclusively, while another used only 'eating'. Since the dataset contains lots of food pictures, finding correct images was relatively easy, as test subjects only needed to look at the time of the images. For the last task, test subjects used either 'food' or 'vegetable' again (four times), followed by 'kitchen' and 'cooking' (twice), and 'cup' (once). Again, as with task 2, no images were found using this approach, but this time, no test subject came even close to finding correct images.

Another interesting observation is the difference between unique images viewed, and the total number of images viewed, per task, as can be seen in figure 19. For tasks 1, 2 and 4, the difference is only minimal, but for task 3 the difference is twice as much. This can be explained by the fact that there are simply much more images that match the criteria for task 3 than for all other tasks. Users therefore examined the same images a couple of times, because they seem omnipresent in the dataset, which was also observed from the screen recordings. Thus, this explains why test subjects found more images for task 3 than for task 1; there are simply more images of food in the dataset, than e.g. images of the outside of airplanes.

Finally, the standard deviation of these task results are relatively high compared to the averages of the task results, as seen in figure 19. For the first task, this is mostly caused by the results of user 2, which will be discussed in section 6.2.3. Figure 20 shows the same results as figure 19, except the results of the first task of user 2 are omitted. This shows how a completely different browsing approach would influence such task results. However, the deviations still indicate that the task performance of various test subjects differ greatly, as can be seen in both figures. Therefore, we will now examine subgroups of test subjects in more detail.

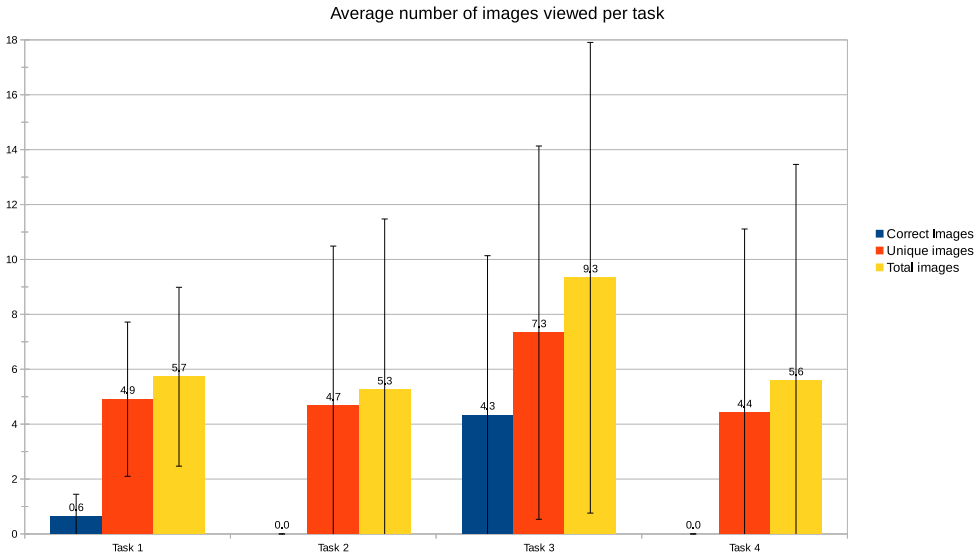

Figure 20: Average number of images viewed per task, for all test subjects. It is the same figure as 19, except the results of the first task from user 2 are omitted. Notice the significantly reduced standard deviation for the results of the first task.

6.2.2 Influence of VR experience

Given our limited sample size, we consider the only two test subjects who mentioned that they use VR occasionally, as experienced with VR, since no one noted that they use VR often. One of these two test subjects is user 1 (see section 4.1.1), the other is a local student. Figure 21 shows the task results for both. As the average results for the first task are skewed because of user 2, comparing these results to figure 20 as well clearly shows that they browsed more images and performed better than the average test subject.

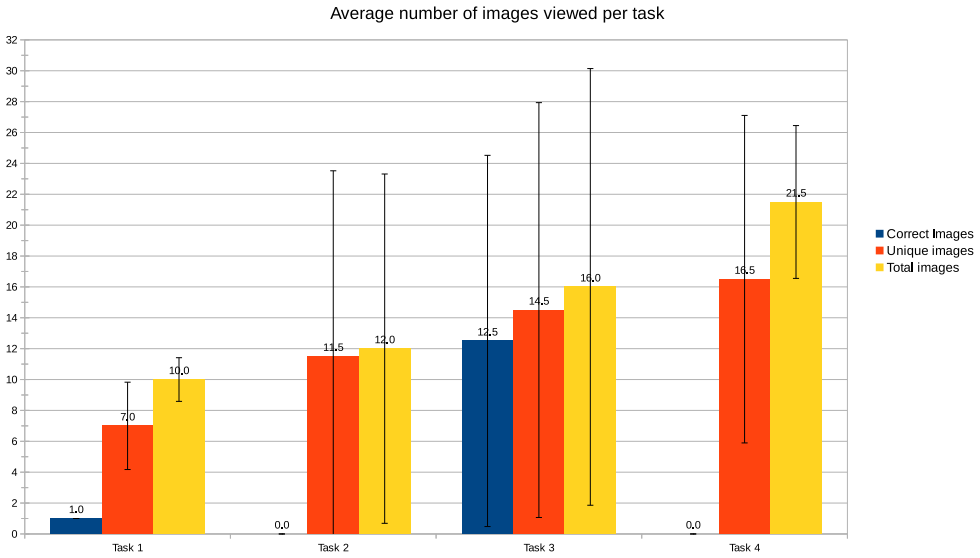

Figure 21: Average number of images viewed per task, for two test subjects experienced with VR. Again, no correct images were found for task 2 and 4. The error bars indicate the standard deviation. Note how low the deviations for the first task are, compared to the other tasks, indicating similar performance.

Again, due to our limited number of test subjects, only two test subjects indicated that they had never used VR before, and both were local students from the University of Utrecht. Figure 22 shows their task results, and clearly showed that they browsed less images, and also performed worse than the average test subject. Again, their results should be compared to both figure 19 and 20, due to the influence of the results of the first task from user 2.

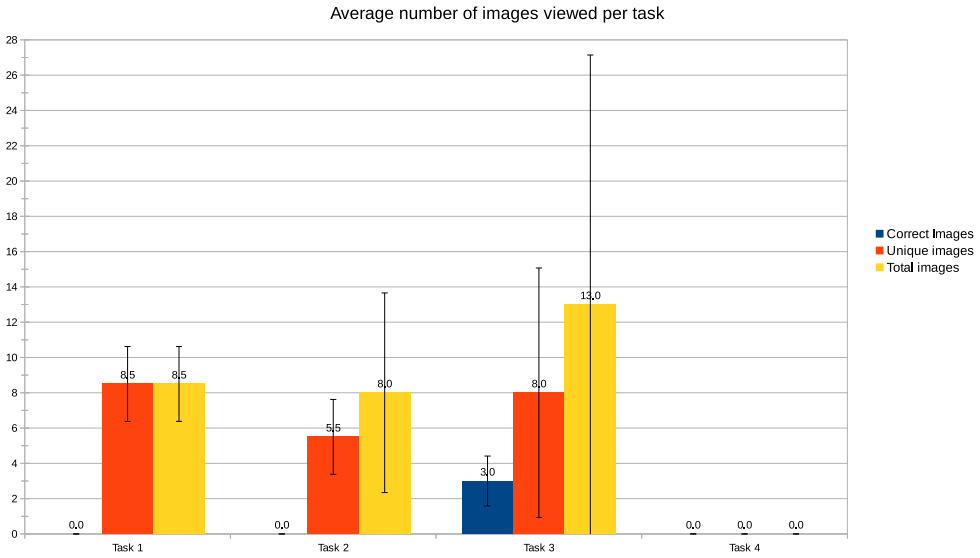

Figure 22: Average number of images viewed per task, for two test subjects who had never used VR before. The novelty of VR greatly impacted their task results, with less correct images found than the average test subject.

6.2.3 Influence of dataset affiliation: user 1 and 2

Figure 23 and 24 show the individual task results for user 1 and user 2 respectively. Interestingly enough, user 1 found practically the same amount of correct images as all test subjects on average. We expected that user 1 would perform better at these tasks, since it is his own, personal data. One reason for this might be that the images were taken over 2 years ago, compared to when they were tested with our system. Given the size of the lifelogging image collection of user 1, it is understandable that not all images and their details are remembered clearly. When examining the recordings from user 1, this was mentioned as well: “I don’t remember about this”. Another reason for these similarly low amounts can be attributed to the enforced time limit as well, which was clearly mentioned in the recordings by user 1, when the time limit was reached: “What? That is not enough time”.

However, user 2 did not find any correct images at all. When looking at the individual statistics and screen recordings, we found that user 2 did not use the filtering menu for the first task, but instead used a map-approach to determine where user 1 made airport stops. For the remaining tasks, user 2 did use the filtering

menu, but did not use it fast enough and had difficulties remembering the controls of the system, resulting in only very few browsed images. This also explains the relatively high number of images viewed for the first task, as it is much lower for the remaining three tasks and then also more consistent with the average numbers for local test subjects.

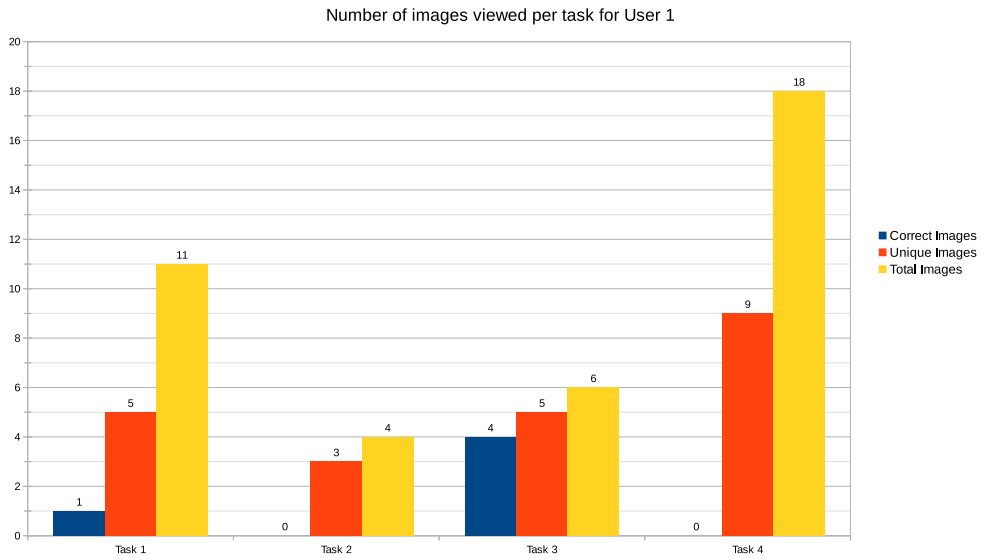

Figure 23: Task results for user 1. Only correct images were found for task 1 and 3, and were found in roughly the same amounts as the average number of correct images found by local test subjects, as can be seen in figure 19 and 20.

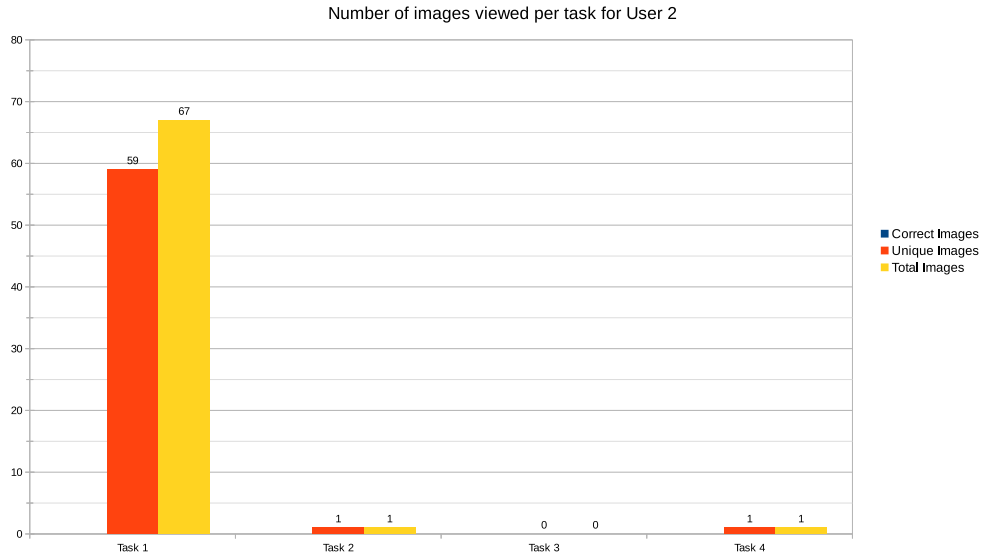

Figure 24: Task results for user 2. Unfortunately, no correct images were found for any task.

Furthermore, when browsing, user 1 used tags ‘airport’ and ‘airplane’ to find the correct image(s), similar to the ones used by other test subjects. For tasks 2, ‘water’ and ‘sandy’ were used, yielding images that met the criteria halfway (only walking by the sea), exactly the same as with the other test subjects since no correct images were found. Interestingly, user 1 used the ‘eating’ tag for task 3 (no one else used it), and found 4 correct images using that approach. For the last task, ‘cooking’ was used, but it did not yield correct images.

In contrast, user 2 barely used the filtering menu, and did not even use it for the first task, being the only one who did not use it for a task. User 2 only used the ‘sale’ tag for task 2, but the interpretation of the tag contradicted its actual concept. For task 3, the ‘food’ tag was used, but due to interaction mistakes (observed from the recordings) and the enforced time limit of 1 minute, no correct images were found. The tags used for the last task was ‘cup’, but again, it did not yield correct images.

6.3 Interaction results

For the interaction results, users 1 and 2 were not separated from the local test subjects, since their affiliation with the dataset likely does not influence how they

interact with the program, since neither subgroups have used or seen the program before.

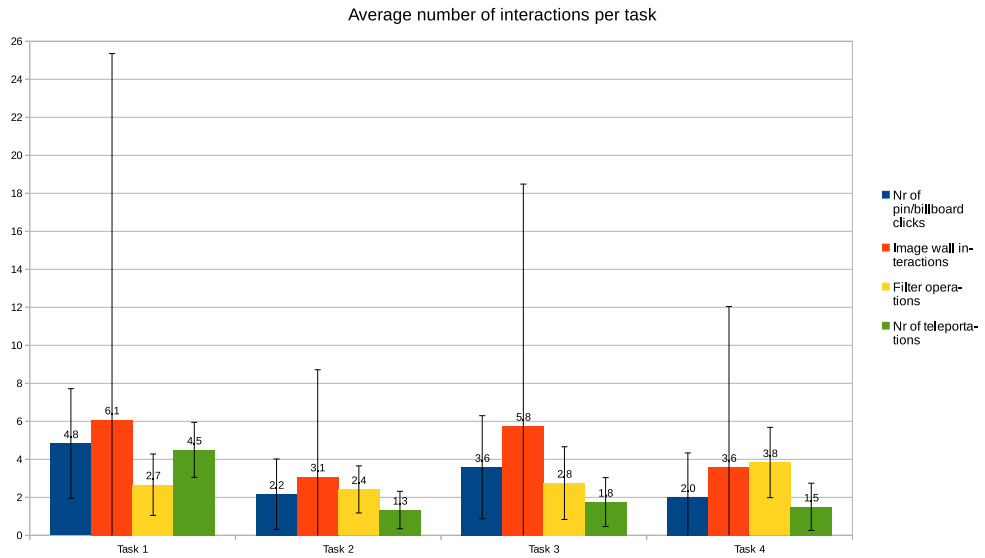

Figure 25: Interaction results for all test subjects, per task.

Figure 25 shows the interaction results for all test subjects, per task. This time, the standard deviation of most results is relatively low compared to their averages, with the exception of the image wall interactions results, and, to a lesser extent, the number of pin/billboard clicks. This figure indicates that for these tasks, all test subjects used a similar approach of two steps. First, the filtering menu would be used to enable only specific filters, and then the test subjects would teleport to locations with active images. Second, the user would grab and navigate those images, in search for the correct ones for their current task. This was also confirmed when examining the screen recordings. Only user 2 deviated from this approach for the first task (but not for subsequent tasks) as explained in section 6.2.3, which also explains why the number of image wall interactions had such a high standard deviation for the first task.

Of course, no test subject performed equal, as some looked at more images than others during their tasks, furthermore explaining the deviation between the number of image wall interactions. However, a more interesting result is the number of teleportations performed per task. Overall, test subjects teleported around the map more for the first task, indicating that the map aspect might have played a role for this task.

However, such a conclusion cannot be easily made in this case. When examining

the screen recordings and the dataset, all (active) images showed a reasonably similar spatial distribution on the map after initial filtering from the test subjects. In other words; there are multiple locations that have images matching the filters, at distances sufficiently far away that the user would need to teleport closer to them in order to view and navigate them. However, for the first task, each location had only a very small amount of matching images, in contrast to the results of e.g. the third task, which had significantly more images at each location. Thus, this means that users were more or less forced to visit other locations, as the images at other locations were quickly exhausted.

6.4 SUS Scores

For the SUS scores, users 1 and 2 were again separated from all test subjects, since their affiliation with the dataset likely does influence their opinion of the program, and thus their SUS scores. In this subsection, we examine the results of all test subjects, and then a few relevant subgroups again.

6.4.1 All test subjects

The SUS scores for all test subjects are shown in figure 26. The average SUS score was almost 70, with a standard deviation of 15.6. Overall, the program was well-received by almost all test subjects, with relatively high SUS scores, indicating that the they found our system quite usable and intuitive.

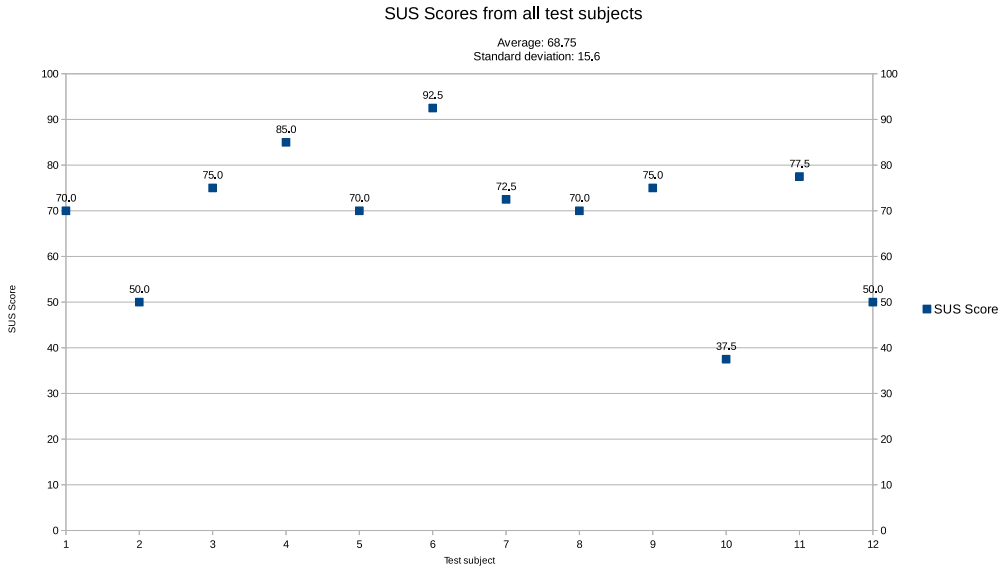

Figure 26: SUS scores for all twelve test subjects. The first ten scores are from the ten local test subjects, the last two scores are from user 1 and user 2 respectively.

6.4.2 Influence of glasses

Figure 27 shows the average SUS score for the seven test subjects who noted that they use glasses, versus the remaining five that did not use glasses. As mentioned before, we expected that people without glasses would enjoy VR less than those without glasses, and we can observe the same bias. This means that test subjects with glasses enjoy our program nonetheless (as shown by their average SUS score of 63.6), but do so less than those without glasses, very likely because of their glasses.

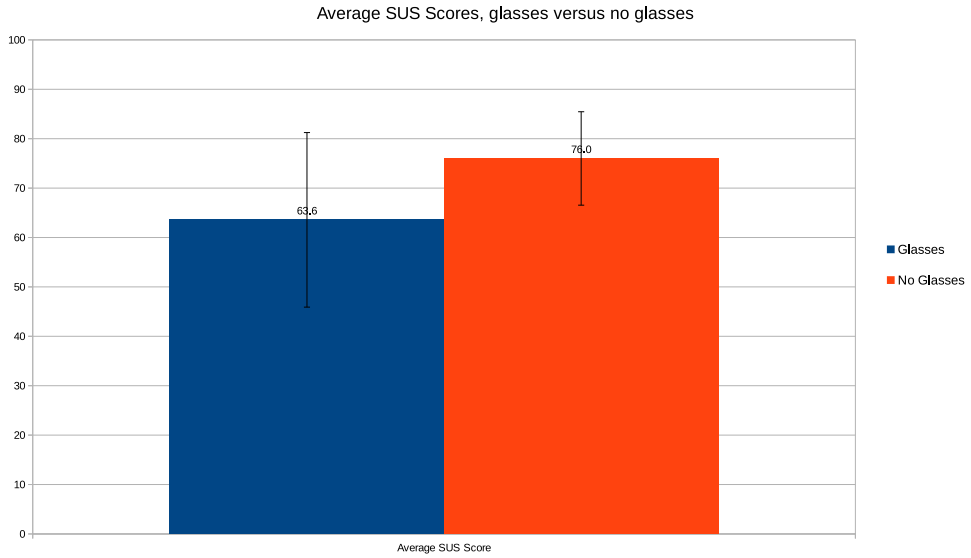

Figure 27: Average SUS scores for the subgroups of test subjects that use glasses, versus those that do not. The error bars indicate the standard deviation.

6.4.3 Influence of VR experience

Of the two test subjects who occasionally used VR, both gave the system high SUS scores. The given scores were 70 and 77.5 respectively, shown in figure 26 as test subject 1 and 11 respectively. Interestingly enough, the two test subjects who had never used VR before, gave even (slightly) higher SUS scores of 85 and 72.5 respectively (test subject 4 and 7 in figure 26).

It seems reasonable that this apparent difference can be attributed to the novel experience of VR, rather than to our program. However, it also indicates, that our system is sufficiently optimized (both performance-wise and interaction-wise), as novel users rated our program as high as experienced users, despite their initial learning curve of VR.

6.4.4 Influence of task performance

In order to evaluate the influence of task performance, an objective measure is needed first, for consistency and clarity. Therefore, we define “bad task performance” as the case when a test subject has not found a single correct image for a task. Then, we define “good task performance” as the case when a test subject

has found more correct images for a task than the average test subject. Given the average task results, this means that a test subject performed well for the first task, if he found 1 or more correct images. For the third task, a test subject performed well if he found 5 or more correct images. The remaining two tasks will not be used, since no test subject found correct images for it. We make this distinction, so we can objectively compare both subgroups consistently.

When examining the individual results of all test subjects, we found that test subjects 1, 2, 5, 10 and user 1 performed well on the first task, having found at least 1 correct image. For the third task, almost the same test subjects performed well, except user 1 only found 4 correct images. When examining the screen recording for the third task of user 1, more correct images were found just after the time limit by half a second, which would have been found legitimately if user 1 did not make an interaction mistake (teleporting instead of image grabbing). Therefore, we consider test subjects 1, 2, 5, 10, and user 1, as the group of test subjects performing well. On the other hand, test subjects 6, 8, and user 2 found no images for any task, and will thus be considered the group of test subjects that performed badly.

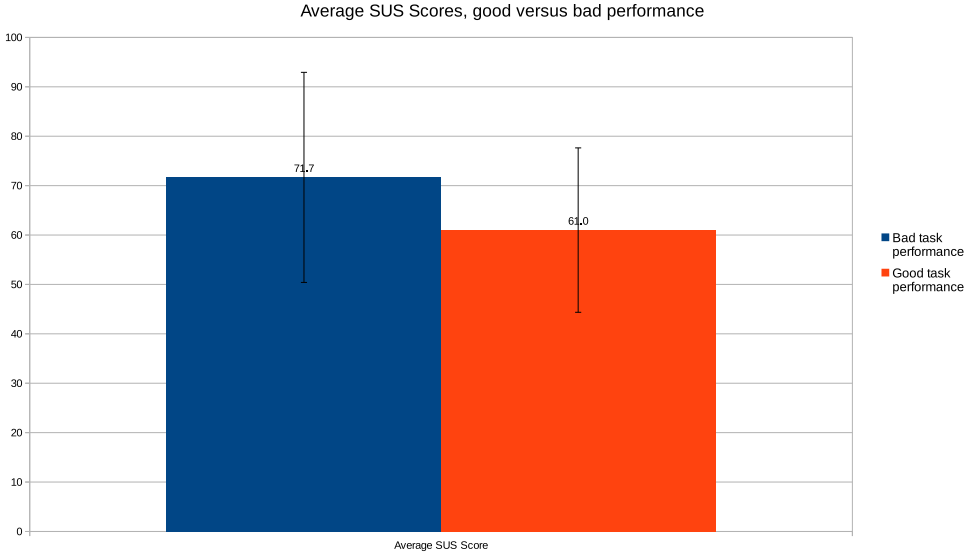

Figure 28: Average SUS scores for the subgroups of test subjects that performed well versus those that performed badly. The error bars indicate the standard deviation.

Figure 28 shows the SUS scores of both subgroups. Interestingly enough, and completely against our expectations, test subjects that performed worse, did not

give lower SUS scores than the other group that performed better. In fact, it seems that test subjects who performed worse, actually gave higher scores. This may be explained by our relatively small sample size and the high standard deviation of these values. Perhaps our expectations will be met with larger sample sizes.

6.4.5 Influence of dataset affiliation: user 1 and 2

Users 1 and 2 gave our system a SUS score of 77.5 and 50 respectively. Since both users are active lifeloggers, who might benefit from using a system such as ours to look back on their old images, the first score of 77.5 is reasonably high. It is even higher than the average SUS score of 70 as given by the average test subject. However, the second score is not, but this can be partly explained by the fact that user 2 did not perform well on the tasks, since no correct images were found for any task.

6.5 Qualitative analysis

For the qualitative analysis, users 1 and 2 were again separated, since their affiliation with the dataset likely does influence their opinion of the program, and thus their responses. In this subsection, we examine the results of the final questions as discussed in section 5.3.2. We start with a more general analysis of all test subjects, followed by the analysis of local test subjects, and finally users 1 and 2.

As can be seen from figure 29, not a single test subject experienced significant motion sickness. Exactly half experienced none at all, whereas the other half only experienced it a little bit. This is a clear indication that our program is optimized sufficiently, and our approach sufficiently intuitive so motion sickness does not happen severely.

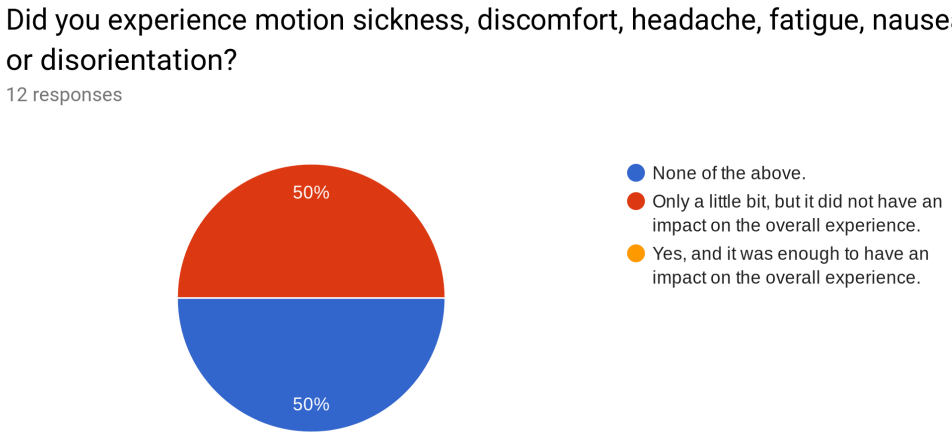

Figure 29: Answer to the motion sickness question for all twelve test subjects, as described in section 5.3.2.

Furthermore, when browsing through the images, ten out of all twelve test subjects browsed images using mostly a filtering approach when performing the tasks, as seen in figure 30. This is in line with the general results on interaction statistics, as most test subjects spent their time interacting with the filtering menu, as seen in 6.3.

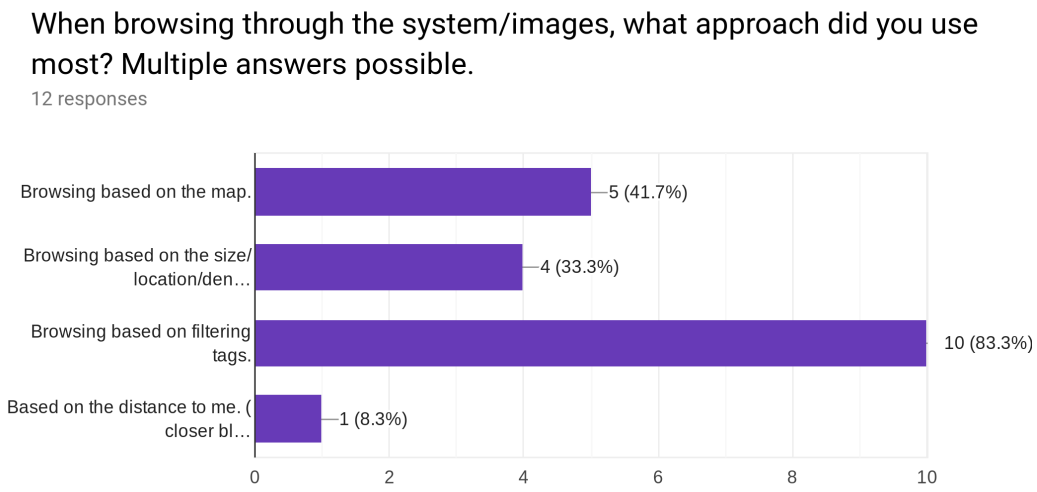

Figure 30: Answers to the browsing approach question for all twelve test subjects, as described in section 5.3.2. The last option should read 'Based on the distance to me (closer blocks are easier to click)'.

6.5.1 Local test subjects

Overall, the system was well received by the local test subjects, as can be seen from the SUS scores in figure 26, as well as from the answers to the first open questions from section 5.3.2. Most test subjects complimented the map approach as well as their grouping by location and day, and found the system interesting and fun to browse images with. They specifically liked the location-based approach, as well as the precise placement of images. One test subject complimented the system for its ease of use of scrolling through images, and thereby viewing the images from a journey more clearly, which is great feedback for a system designed for viewing lifelogging images such as ours. Another test subject, who had never used VR before, liked the feeling of depth and moving through space, but did not explicitly attribute this feedback to our approach, likely attributing it to (the novelty of) VR. Furthermore, the filtering of images was very powerful and even though they were automatic, they were found accurate (often) enough to be usable.

However, the actual interaction with the filtering aspect received more criticism, in response to the second question of 5.3.2. Given the rather large dataset of 56450 images, test subjects found it took too long to navigate through all the photos. Mostly, this was because they found the filtering menu not intuitive enough. Two

of them expected that they could interact with the filtering menu by using the other controller, which is not how it currently works. After getting used to the system, they learned to interact with it properly, but found it unintuitive nonetheless. Others found that there were too many irrelevant filters, as well as filters that were difficult to interpret, suggesting more specific filters. A direct example for this is the inclusion of the tag 'vegetable', but the absence of the tag 'fruit'. Also, even though all tags were auto-generated, test subjects would like the filters to be more accurate, in order to find images faster and easier.

Furthermore, the teleportation speed was often too high for almost half of the test subjects, but they found the speed of moving upwards too low. Finally, one test subject explicitly mentioned that the (added) value of the system seems absent, but also commented that looking back at images is not something the test subject likes or would do.

In response to question 5 from the final questions of section 5.3.2, seven out of ten test subjects would like to use this system for their own photos, and/or use it to show their photos to others. One test subject suggested that the map could be bigger, so as to show even more fine-grained image locations when showing images. Another test subject suggested the inclusion of (external) meta-information, so that the program would show some interesting information about various places that the user went to, e.g. historical or biological. The remaining three test subjects would not use the system, all saying that they are not really interested in using such a system and/or do not really look back at their old pictures.

Finally, in response to question 6 of section 5.3.2, a few test subjects suggested that the usability and user-friendliness can be improved. Another suggested to incorporate a timeline feature, either as a filtering option, or as an animation. Others commented their appreciation for the system, by e.g. saying "Great concept!" and "Awesome! Great experience".

6.5.2 User 1 and 2

In response to the first question of section 5.3.2, user 1 complimented the visualization and exploration interface, saying that it was "easy to use, and looks great". When analyzing the recordings, user 1 was surprised by the image wall at first, but after a couple of seconds, commented positively "ohhh, I could get used to this". User 2 even mentioned that "The map view makes perfect sense for lifelogging data". Both answers are great feedback for our system, and an indication that our map-based approach seems warranted.

However, like the other test subjects, both users found the visual concepts not

accurate enough. User 1 commented that the images on the poles (the white lines connecting pins and images to a position on the map, not the Earth's poles) were too small to analyse its contents, and that their density was too high. Furthermore, the filtering menu took user 1 "away from the image interaction to a new menu", degrading his experience.

User 2 disliked the number of buttons to control the navigation, and noted the absence of time-based access functionality. Also, user 2 was the only test subject to request a feature to rotate the map, which would likely be used to re-orient the user, and to make the names of locations on the map readable from all angles.

In response to question 5 from the final questions of section 5.3.2, user 1 would use our system for "entertainment, and to share his content with friends". User 2 also wanted to use it to show his life experience to friends. Furthermore, it would be used to remember trips and events, as well as people that user 2 met. User 2 even commented that "The VR is entertaining to pull you out of reality and relax". Again, this is great feedback for our system, and an indication that our leisure approach, in combination with VR, seems warranted.

When answering the last question of section 5.3.2, user 1 noted that the image billboards should be larger, so that the content of the images would be easier to analyze. Finally, both user 1 and 2 commented that voice commands should be added, suggesting an alternative interaction method in addition to the HTC Vive controllers, which "would be greatly helpful to explore lifelogging data".

7 Conclusion

In this paper, we demonstrated a proof-of-concept implementation of a map-based image browsing system, that allows for browsing geo-tagged lifelogging images in VR. It allows for browsing very large image datasets in real-time, as well as dynamically filtering it via the use of concepts (keyword-based image content, e.g. 'car', 'food', etc.) detected from automated computer vision programs. The LSC 2018 dataset [1] was used as a representative test set, containing data from two active lifeloggers, including the detected concepts from the computer vision programs. In order to provide easier and more fine-grained image access, a clustering hierarchy was used to group images of similar locations, and present them per day via the use of an image wall, which could be navigated by the user. Furthermore, a high-resolution dynamic map was used, to show the world at various level of detail ('zooming'), and to provide content-dependent maps (map scaling), resulting in accurately displayed locations of images. Even though our system was designed for leisure browsing, a pilot study was performed that tested both quantitative and qualitative aspects, using ten general users as well as the two active lifeloggers who created the aforementioned LSC dataset.

7.1 Quantitative aspect

Our quantitative results indicate that the system does not excel in performance search, mostly because it was not designed for this. The quantitative testing consisted of four tasks, of which two had a location aspect, whereas the remaining two did not. We expected that users would perform better for the tasks that have a location aspect due to the map-based approach. However, users found more images for a task that did not have a location aspect, because the dataset simply contained more matching images for that task than for all others. Also, the novelty of the system, in combination with the low amount of time allowed per task, limited test subjects from finding many images and utilizing the map properly.

When examining subgroups of test subjects, we found that the test subject's experience with VR greatly influenced their task results. When comparing test subjects experienced with VR, to test subjects who are inexperienced with VR, results showed a large increase in task performance for the first subgroup, even though the sample size was limited. Given the novelty and the relative unfamiliarity with VR of most test subjects, this posed a noticeable learning curve.

Interestingly enough, affiliation with the dataset did not show an increase in task performance. The two lifeloggers whose data was used for the LSC dataset, did

not perform better than the average test subject. This can be attributed to the fact that the enforced time limit was limiting their performance. Also, the age of the dataset (a two-year difference between its creation, and our testing) prevented one lifelogger from remembering their context, which was explicitly mentioned: "I don't remember about this".

7.2 Qualitative aspect

The results of a System Usability Survey (SUS) [13] showed that test subjects found our system quite usable and intuitive, as indicated by a high average SUS score of 68.75 ± 15.6 (scaled 0-100). Furthermore, the influence of wearing glasses noticeably lowered the average SUS score by more than ten points, compared to test subjects that did not use glasses, thereby confirming our expected bias.

However, test subjects that performed better did not give a higher SUS score than those who performed worse; in fact, they even gave a ten point higher average SUS score. Given the high standard deviation of these average SUS scores (showing great overlap from their averages), this could be attributed to our relatively low sample size. Also, affiliation with the dataset did not influence the SUS scores noticeably.

When examining the answers to our more qualitative-oriented questions, no test subjects experienced motion sickness enough to impact the overall experience, and none mentioned low system performance, proving the high performance of our implementation. Most test subjects explicitly complimented the map approach, the VR aspect, and the grouping of images by location and day, as well as the overall visualization. Its ease of use, and applicability to lifelogging images was also mentioned, especially by one of the lifeloggers in particular, saying: "The map view makes perfect sense for lifelogging data". However, as mentioned by most test subjects, the filtering aspect could be improved, suggesting more accurate and specific filters, and better interaction with the filtering menu. Furthermore, the teleportation speed was often found to be too high, whereas the zooming out speed was too low.

Additionally, nine out of twelve test subjects would like to use our system for their own photos, and/or to show their photos to others. The remaining three indicated that they were not motivated to use it, and do not look back on their photos in general, even though one test subject mentioned that the system was "kind of fun" nonetheless. The "fun" aspect was mentioned by a few other test subjects, complimenting the system by e.g. saying: "Awesome! Great experience" or "Great concept!".

7.3 Final conclusion

Our findings indicate that our map-based approach seems warranted, and that our system is useful for browsing lifelogging data. The high performance of our system demonstrates its ability to browse very large image datasets consisting of tens of thousands of images, such as lifelogging data. Even though our system was not designed for performance search, test subjects found correct images within strict time constraints nonetheless. The VR aspect contributes to the entertainment value by successfully providing an immersive experience of browsing images, which cannot be paralleled by traditional 2D screens. Finally, the high entertainment value of our system could motivate users to look back at their photos more often, and thus proves our system's applicability for leisure browsing.

8 Discussion

Our results showed that our system has proven its claims with regards to the map-based approach, the applicability to lifelogging data, and the high entertainment value as well as the usefulness for leisure browsing. However, there is still room for improvement. Most test subjects commented that the filtering aspect could be improved in many ways, but the major issue was the accuracy of the filters. Even though they were automatically generated using computer vision software, the accuracy could be further improved. Also, more distinct and specific filters could be used, e.g. 'vegetable' is present but 'fruit' is missing. The filtering menu itself could also benefit from interaction improvements, as some test subjects did not find it intuitive enough.

Furthermore, the system is currently limited with regards to the map. Since storage requirements for map tiles grow quadratically, the system is currently limited to a fixed level of map detail, as the map tiles are stored locally. If the map tile system would be upgraded to a dynamic quad-tree system (as discussed in section 4.5.2), then tiles could be loaded at even higher levels of detail by fetching them from the internet dynamically. Then, the 'physical' size of the map (in VR) could also be more easily scaled, to allow for even more fine-grained location-based image access.

Finally, clear conclusions about our system cannot always be drawn directly. For instance, one test subject used a completely different approach than all others for one task, thereby skewing the results significantly. Without the screen recordings, this would not have been easily detectable. Another issue was the higher number of teleportations for the first task compared to others, deceptively suggesting that the map aspect played an important role for that task. However, each location had only a small number of images after applying filters, thereby forcing test subjects to visit other locations to examine more images. In addition, the relatively low sample sample size cannot always be used to make clear conclusions.

8.1 Future work

Based on all given feedback, we suggest the following items for future work:

- The improvement of the filtering aspect, by using more accurate and specific filters, and by improving the filtering menu itself, to allow for more efficient image access.
- The improvement of the map aspect, to allow for a (bigger) map of even

higher levels of detail, and thus more fine-grained image access based on location.

- The addition of voice commands and/or other interaction optimizations, to improve the user interaction of the system.
- The addition of a temporal aspect to the system, improving the meta-context applicability to lifelogging data. Currently, this is ongoing research, but perhaps a similar approach as done by Alice Thudt [30] can be integrated into the system, which combines spatial and temporal information into 'visits'.
- The addition of location information to the system, by showing general information about certain locations, thereby adding to the entertainment value.
- The addition of a personal aspect to the system (e.g. face recognition), improving the meta-context applicability to lifelogging data.
- A more thorough evaluation, by testing more people and comparing our system to other systems, verifying the applicability of our system and our results.

We expect that the further optimization of the implementation, by improving the filtering, interaction and map aspects etc., as well as the addition of a temporal aspect, would greatly benefit the system. Such additions could lead to a system that would still be as useful for leisure browsing, but could also be used more effectively for performance search. Currently, the addition of a temporal aspect is ongoing research at the University of Utrecht.

9 References

[1] LSC 2018 website. http://lsc.dcu.ie/index.html and http://ntcir-lifelog.computing.dcu.ie/NTCIR13/styled-3/index.html.

[2] NTCIR 12 website. http://research.nii.ac.jp/ntcir/ntcir-12,.

[3] NTCIR 13 website. http://research.nii.ac.jp/ntcir/ntcir-13/index.html and http://ntcir-lifelog.computing.dcu.ie/NTCIR13/styled-3/index.html,.

[4] Number of GPS digits versus precision. https://gis.stackexchange.com/questions/8650/measuring-accuracy-of-latitude-and-longitude/8674#8674.

[5] Origin of the map tiles. http://{server}.basemaps.cartocdn.com/dark_all/{zoom}/{x}/{y}.png.

[6] Mercator projections. https://wiki.openstreetmap.org/wiki/Mercator.

[7] Sqlite database. https://www.sqlite.org/index.html.

[8] Unity. https://unity3d.com/,.

[9] Listview plugin for unity. https://assetstore.unity.com/packages/tools/gui/listview-for-unity-ui-21430,.

[10] Steamvr plugin for unity. https://assetstore.unity.com/packages/tools/integration/steamvr-plugin-32647,.

[11] Task parallel plugin for unity. https://assetstore.unity.com/packages/tools/integration/task-parallel-82257,.

[12] J. Bolwerk. controlled navigation in virtual reality for exploratory image browsing. Master's thesis, 2017.

[13] J. Brooke et al. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.

[14] T. T. A. Combs and B. B. Bederson. Does zooming improve image browsing? In *Proceedings of the Fourth ACM Conference on Digital Libraries*, DL '99, pages 130–137, New York, NY, USA, 1999. ACM. ISBN 1-58113-145-3. doi: 10.1145/313238.313286. URL http://doi.acm.org/10.1145/313238.313286.

[15] A. Duane and C. Gurrin. Pilot study to investigate feasibility of visual lifelog exploration in virtual reality. In *Proceedings of the 2Nd Workshop on Lifel-*

ogging Tools and Applications, LTA '17, pages 29–32, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-5503-2. doi: 10.1145/3133202.3133208. URL http://doi.acm.org/10.1145/3133202.3133208.

[16] A. Duane and C. Gurrin. Lifelog exploration prototype in virtual reality. In K. Schoeffmann, T. H. Chalidabhongse, C. W. Ngo, S. Aramvith, N. E. O'Connor, Y.-S. Ho, M. Gabbouj, and A. Elgammal, editors, *MultiMedia Modeling*, pages 377–380, Cham, 2018. Springer International Publishing. ISBN 978-3-319-73600-6.

[17] C. Gurrin, A. F. Smeaton, and A. R. Doherty. Lifelogging: Personal big data. *Foundations and Trends in Information Retrieval*, 8(1):1–125, 2014. ISSN 1554-0669. doi: 10.1561/1500000033. URL http://dx.doi.org/10.1561/1500000033.

[18] L. J. Hettinger and G. E. Riccio. Visually induced motion sickness in virtual environments. *Presence: Teleoperators and Virtual Environments*, 1(3):306–310, 1992. doi: 10.1162/pres.1992.1.3.306. URL https://doi.org/10.1162/pres.1992.1.3.306.

[19] W. Hürst, K. Ouwehand, M. Mengerink, A. Duane, and C. Gurrin. Geospatial access to lifelogging photos in virtual reality. In *Proceedings of the 2018 ACM Workshop on The Lifelog Search Challenge*, pages 33–37. ACM, 2018.

[20] S. Khanwalkar, S. Balakrishna, and R. Jain. Exploration of large image corpuses in virtual reality. In *Proceedings of the 24th ACM International Conference on Multimedia*, MM '16, pages 596–600, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3603-1. doi: 10.1145/2964284.2967291. URL http://doi.acm.org/10.1145/2964284.2967291.

[21] K. J. Kim and S. S. Sundar. Does screen size matter for smartphones? utilitarian and hedonic effects of screen size on smartphone adoption. *Cyberpsychology, Behavior, and Social Networking*, 17(7):466–473, 2014.

[22] T. S. H. Munehiro Nakazato. 3d mars: Immersive virtual reality for content-based image retrieval. In *IEEE International Conference on Multimedia and Expo, 2001. ICME 2001.*, CHI '01. IEEE, 2001. ISBN 0-7695-1198-8. doi: 10.1109/ICME.2001.1237651. URL http://clamsitel.pbworks.com/f/Immersive\%20Virtual\%20Reality\%20for\%20Content-Based\%20Image\%20Retrieval.pdf.

[23] W. Plant and G. Schaefer. Navigation and browsing of image databases. In *2009 International Conference of Soft Computing and Pattern Recognition*, pages 750–755, Dec 2009. doi: 10.1109/SoCPaR.2009.152.

[24] J. M. Rigby, D. P. Brumby, A. L. Cox, and S. J. J. Gould. Watching movies on netflix: Investigating the effect of screen size on viewer immersion. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*, MobileHCI '16, pages 714–721, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4413-5. doi: 10.1145/2957265.2961843. URL http://doi.acm.org/10.1145/2957265.2961843.

[25] K. Rodden. Filter image browsing. In *Evaluating Similarity-Based Visualisations as Interfaces for Image Browsing*. University of Cambridge, 2002.

[26] K. Rodden, W. Basalaj, D. Sinclair, and K. Wood. Does organisation by similarity assist image browsing? In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '01, pages 190–197, New York, NY, USA, 2001. ACM. ISBN 1-58113-327-8. doi: 10.1145/365024.365097. URL http://doi.acm.org/10.1145/365024.365097.

[27] G. Schaefer. A next generation browsing environment for large image repositories. *Multimedia Tools and Applications*, 47(1):105–120, Mar 2010. ISSN 1573-7721. doi: 10.1007/s11042-009-0409-2. URL https://doi.org/10.1007/s11042-009-0409-2.

[28] G. Schaefer, M. Budnik, and B. Krawczyk. Immersive browsing in an image sphere. In *Proceedings of the 11th International Conference on Ubiquitous Information Management and Communication*, IMCOM '17, pages 26:1–26:4, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4888-1. doi: 10.1145/3022227.3022252. URL http://doi.acm.org/10.1145/3022227.3022252.

[29] R. E. Thayer, J. R. Newman, and T. M. McClain. Self-regulation of mood: Strategies for changing a bad mood, raising energy, and reducing tension. *Journal of personality and social psychology*, 67(5):910, 1994.

[30] A. Thudt, D. Baur, and S. Carpendale. Visits: A spatiotemporal visualization of location histories. In *Proceedings of the Eurographics Conference on Visualization*, 2013.

[31] R. S. Torres, C. G. Silva, C. B. Medeiros, and H. V. Rocha. Visual structures for image browsing. In *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, CIKM '03, pages 49–55, New York, NY, USA, 2003. ACM. ISBN 1-58113-723-0. doi: 10.1145/956863.956874. URL http://doi.acm.org/10.1145/956863.956874.

[32] C. C. Yang. Content-based image retrieval: A comparison between query by example and image browsing map approaches. *Journal of Information*

Science, 30(3):254–267, 2004. doi: 10.1177/0165551504044670. URL https://doi.org/10.1177/0165551504044670.

[33] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '03, pages 401–408, New York, NY, USA, 2003. ACM. ISBN 1-58113-630-7. doi: 10.1145/642611.642681. URL http://doi.acm.org/10.1145/642611.642681.

10 Appendix

10.1 GPS Coordinates and precision

Table 4 shows the difference between the number of digits in a decimal degree, and the precision [4].

Range of values	Maximum precision
10 digits	about 1000 km
1 digit	about 111 km
1 decimal	11.1 km
2 decimals	1.1 km
3 decimals	110 m
4 decimals	11 m
5 decimals	1.1 m
6 decimals	0.11 m
7 decimals	11 mm
etc...	etc...

Table 4: Table showing the relation between the number of digits, and the precision. Here, km stands for kilometers, m for meters, and mm for millimeters [4].

10.2 Consent Form

In order to waiver liability of the researchers, test subjects needed to sign a consent form, included below.

Risks, Discomforts and Benefits
Be aware that when using virtual reality systems, some people may experience some degrees of the following: Nausea, Vomiting, Sweating, Pallor, Headache, Vertigo and/or Dizziness

Furthermore using VR applications and games have the possibility of creating epileptic episodes, therefore people who are known to have suffered from epilepsy are not allowed to volunteer.

Upon request, testing will be immediately terminated or if there are indications that the discomfort becomes unbearable or abnormal responses occur. Participation in this study should be an interesting and enjoyable experience and the results obtained are expected to assist computer science research.

Confidentiality
Any information that is shared during the study will be treated strictly confidential and once the study is completed, it will not be possible to identify individuals. Throughout the study only the aforementioned researchers will have access to the information.

Request for Further Information
You are encouraged to discuss any concerns regarding the study with the testing researcher at any time, and to ask any questions that you might have.

Refusal or Withdrawal
You may refuse to participate in the study and if you do consent to participate then you will be free to withdraw from the study at any time without consequence, fear or prejudice. If you wish to withdraw from the event please contact the researcher and all data pertaining to you will be destroyed.

I have read the information above	YES / NO
I have had the opportunity to ask questions about the procedure	YES / NO
All my questions were answered to my satisfaction	YES / NO
I have received sufficient information about the study	YES / NO
I understand and accept the risks associated with the use of virtual reality	YES / NO
I certify to have no history of epilepsy	YES / NO
Name	
Date	
Signature	

11 Acknowledgements

We would like to thank Marijn Mengerink, for providing the initial implementation of our system so that we could modify and extend it, as well as for his ongoing technical support. Furthermore, we would like to thank Aaron Duane, for performing our experiment with the two creators of the LSC dataset. Finally, we thank all test subjects for participating.