Visual Memorability for Egocentric Cameras

Marc Carné Herrera
June 2016

Supervised by Dr. Xavier Giró-i-Nieto
and Dr. Cathal Gurrin
Acknowledgements

I would like to thank my supervisor Dr. Cathal Gurrin for allow me to use his egocentric images for this project and Dr. Xavier Giró, my supervisor from UPC (Polytechnic University of Barcelona). Thanks are also due to Alejandro Cartas, PhD candidate from University of Barcelona (UB) for his support on web application (that will be presented on this work) building, and Eva Mohedano, PhD candidate from Dublin City University for her contribution in the regression model. As there is an experiment as part of this work, and user interaction was needed I would like to thank all that volunteers that done the visual game, contributing to this research. Graham Healy, as an electroencephalographic signals acquisition and process expert, has an important contribution in the final chapter of this project. Finally I would like to express my deep appreciation to Albert Gil and Josep Pujal, for helps me with the server set up and computing support. As this project is related with other projects I want to thank the special collaboration to Dra. Petia Radeva and Dra. Maite Garolera.
Abstract

This project explores visual memorability of egocentric in different ways having three main contributions. The first and the main contribution of the project is a new tool visual memorability in egocentric images. This tool that consists in a web application that allows the annotation of the visual memorability associated to still images with an online game.

The second contribution of this work is a convolutional neural network model for visual memorability prediction that adapts an off-the-shelf model to egocentric images. Moreover, a visualization study has been pursued to localize the regions of the images that are more memorable than others. With this maps a comparison with saliency maps and is explored. This part of the research opens a new branch in visual memorability that consists in use memorability maps for saliency prediction. Also the memorability of the images is related with a sentiment analysis applying a model that predicts that feature.

The final contribution is related to join visual memorability of images with human behaviour and physical state, finding a relation between memory and some physiological signals as: heart rate, galvanic skin response and electroencephalographic signals.
Table of Contents

ACKNOWLEDGEMENTS ............................................................................................................. II

ABSTRACT ............................................................................................................................. III

CHAPTER 1 - INTRODUCTION ............................................................................................... 1
  1.1 OBJECTIVES ................................................................................................................. 2
    1.1.1 Annotation tool for visual memorability in egocentric images ................................. 2
    1.1.2 Adaptation of a model for visual memorability prediction to egocentric images ....... 2
    1.1.3 Correlation between physiological signals and visual memorability ....................... 2

CHAPTER 2 - ANNOTATION TOOL FOR VISUAL MEMORABILITY ................................. 4
  2.1 INTRODUCTION ............................................................................................................. 4
  2.2 RELATED WORK .......................................................................................................... 4
  2.3 EGOCENTRIC VISUAL MEMORY GAME .................................................................... 5
    2.3.1 JavaScript algorithm .............................................................................................. 6
    2.3.2 Image selection for the game ................................................................................ 6
    2.3.3 Docker platform .................................................................................................... 7
  2.4 CONCLUSIONS ............................................................................................................. 8

CHAPTER 3 – EGOMEMNET: VISUAL MEMORABILITY ADAPTATION TO EGOCENTRIC IMAGES ........................................................................................................... 9
  3.1 INTRODUCTION ............................................................................................................. 9
  3.2 RELATED WORK .......................................................................................................... 9
    3.2.1 Machine learning and deep learning ....................................................................... 9
  3.3 EQUIPMENT AND SOFTWARE .................................................................................... 12
    3.3.2 Software ............................................................................................................... 12
  3.4 PERFORMANCE OF MEMNET MODEL OVER EGOCENTRIC IMAGES ...................... 13
  3.5 DOMAIN ADAPTATION OF MEMNET TO EGOCENTRIC VISION ......................... 15
    3.5.1 CNN model evaluation .......................................................................................... 15
  3.6 MEMORABILITY MAPS VISUALIZATION .................................................................. 17
    3.6.1 Method ................................................................................................................ 18
    3.6.2 Memorability maps ............................................................................................... 20
3.7 VISUAL MEMORABILITY VS. VISUAL SALIENCY................................................................. 24
  3.7.1 Similarity method ........................................................................................................ 25
  3.7.2 Similarity results using Jaccard index ........................................................................ 27
  3.7.3 Similarity results using state of the art saliency metrics ........................................... 32
3.8 MEMORABILITY AND SENTIMENT ANALYSIS................................................................... 33
  3.8.1 Method.......................................................................................................................... 33
  3.8.2 Results ........................................................................................................................ 34
3.9 CONCLUSIONS .................................................................................................................. 34

CHAPTER 4 - VISUAL MEMORABILITY AND PHYSIOLOGICAL SIGNALS .................. 36
4.1 INTRODUCTION ................................................................................................................ 36
4.2 INSIGHT DATASET ........................................................................................................... 37
4.3 METHOD ............................................................................................................................ 40
4.4 RELATION BETWEEN MEMORABILITY AND PHYSIOLOGICAL SIGNALS ........... 41
  4.4.1 Analysing obtained data ............................................................................................. 42
  4.4.2 Visualizing memorability prediction results ................................................................ 43
  4.4.3 Searching a correlation. ............................................................................................ 44
4.5 DETECT SNAP POINTS WITH PHYSIOLOGICAL SIGNALS .......................................... 53
  4.5.1 Method........................................................................................................................ 53
  4.5.2 Results ........................................................................................................................ 55
4.6 ADDING PHYSIOLOGICAL CUES TO MEMORABILITY PREDICTION ...................... 55
  4.6.1 Method........................................................................................................................ 56
  4.6.2 Results ........................................................................................................................ 57
4.7 GALVANIC SKIN RESPONSE EXPLORATION ............................................................... 58
  4.7.1 Method........................................................................................................................ 58
  4.7.2 Results ........................................................................................................................ 58
4.8 EEG SIGNALS FOR MEMORABILITY PREDICTION ...................................................... 62
  4.8.1 Method........................................................................................................................ 62
  4.8.2 Results ........................................................................................................................ 64
4.9 CONCLUSIONS .................................................................................................................. 67

CHAPTER 5 – CONCLUSIONS......................................................................................... 68
REFERENCES ....................................................................................................................... 72
GLOSSARY ............................................................................................................................. 74
ANNEX I: JAVASCRIPT ALGORITHM DETAILS................................................................. 75
ANNEX II: COMPUTE SPEARMAN'S RANK CORRELATION ............................................. 78
ANNEX III: FIRST PERSON VISION WORKSHOP CVPR 2016 POSTER ...................... 78
ANNEX IV: DOCKER IMPLEMENTATION DETAILS ........................................... 80
ANNEX V: FPV CVPR 2016 SPOTLIGHT VIDEO IN YOUTUBE ............................ 75

Table of Figures

Figure 1. Visual memory game snapshot .......................................................... 6
Figure 2. Example of fillers and targets ............................................................ 12
Figure 3. Convolutional neural network layer structure ...................................... 16
Figure 4. Convolutional neural network design .................................................. 17
Figure 5. Visual features for a layer of a convolutional neural network .................. 17
Figure 6. Predicted memorability scores against manual annotated scores .......... 19
Figure 7. Image with less prediction (maximum difference) at left and image with most exact prediction (minimum difference) at right .................................................. 20
Figure 8. Saliency map example ...................................................................... 26
Figure 9. Images to explain results in this section .............................................. 27
Figure 10. On top, overlap between original image and the memorability map computed with a transparency of fifty percent. On the bottom, raw memorability map for these images .......... 27
Figure 11. Filtered memorability maps obtained with EgoMemNet CNN model corresponding to the test images presented before .......................................................... 28
Figure 12. Filtered memorability maps obtained with MemNet CNN model corresponding to the test images presented before .......................................................... 28
Figure 13. Visual intersection between saliency and memorability binarized maps .......... 29
Figure 14. Original image, correspondent heat map predicted with a fully convolutional neural network and heat map predicted for the random CNN model for the proposed images ............ 30
Figure 15. Grey scaled and binarized memorability maps for the proposed images .......... 31
Figure 16. Intersection between saliency and memorability map predicted with fully convolutional neural network model for proposed images ................................................................. 31

Figure 17. Comparison between saliency and memorability maps. From left to right: original image, predicted saliency map, predicted memorability map and the value of the similarity and out-region ................................................................. 33-34

Figure 18. Saliency and memorability maps binarized with different thresholds ......................... 34-35

Figure 19. Jaccard indices image, top-left corner corresponds to saliency and memorability thresholds equal to 0 and the step between thresholds is equal to 0.05 ......................................................... 36

Figure 20. Original image at left and coloured map at right. Coloured map has green region that are salient and memorable, red region highlight zones there are salient but not memorable and blue regions shows these zones that are memorable but not salient ........................................ 36-37

Figure 21. Example of a text file with metadata information that includes physiological signals ... 43

Figure 22. Plot of galvanic skin response at left and heart rate values at right for this new dataset 44

Figure 23. Filtered heart rate values .......................................................................................... 45

Figure 24. Filtered galvanic skin response values .................................................................... 45

Figure 25. Filtered heart rate and galvanic skin response for a certain interval of samples ......... 45

Figure 26. Memorability score variation for the dataset analysed ............................................. 47

Figure 27. High and low memorability scores. In red the five hundred highest values and in blue the five hundred lowest values ................................................................. 48

Figure 28. Memorability score histogram for the dataset .......................................................... 49

Figure 29. Most memorable images at top and less memorable images at bottom .................. 49

Figure 30. Plot of heart rate value of the most (in red) and less (in blue) memorable images ....... 50

Figure 31. Plot of sorted heart rate valued for the most (red) and lees (blue) memorable images ... 51

Figure 32. Memorability scores per bin, mean heart rate score per bin and mean galvanic skin response per bin for bin size of 0.05 points ............................................................................. 53

Figure 33. Memorability scores per bin, mean heart rate score per bin and mean galvanic skin response per bin for bin size of 0.03 points ............................................................................. 54

Figure 34. Memorability scores per bin, mean heart rate score per bin and mean galvanic skin response per bin for bin size of 0.001 points ............................................................................. 55

Figure 35. Memorability scores per bin, median heart rate score per bin and median galvanic skin
response per bin for bin size of 0.05 points

Figure 36. Memorability scores per bin, mean heart rate score per bin and mean galvanic skin response per bin for bin size og 0.05 points applying a temporal averaging of scores after bin clustering.

Figure 37. Images woth low heart rate value at top and images with high heart rate value at bottom

Figure 38. Top three images with highest galvanic skin response value

Figure 39. Top three images with memorability score value

Figure 40. Allocation of brain sensors over the scalp

Figure 41. Left - In red Pz channel used; Right – Set up of the experiment

Figure 42. Temporal brain responses (columns) for each target image (rows)

Figure 43. Examples of Pz signal evolution

Table 1. Statistical results

Table 2. Evaluation results between convolutional neural network models

Table 3. Mean Jaccard score comparing binarized saliency and memorability maps with different thresholds

Table 4. Mean value of similarity metrics between saliency and memorability maps

Table 5. Regression models for snap points detection evaluation results

Table 6. Regression models for memorability prediction evaluation results

Table 7. Relation between galvanic skin response and memorability values

Table 8. Memorability ranks based in memorability scores predicted in relation with galvanic skin response

Table 9. Mean position of the top ten GSR images in the memorability ranked list
Chapter 1 - Introduction

“The brain is designed to forget, we need to forget to survive, because we can’t live remembering in each moment all that we have lived”, said Maite Garolera, from the neuropsychology unit of Terrassa Hospital (Spain). For that reason, humans have developed several tools to remember, like wearable cameras.

An egocentric camera is a device that takes images in first person vision (FPV). There are many brands and models of egocentric cameras, with different image quality, lenses… but the purpose of this devices is the same, capture people’s life.

A lifelogger is a person that captures his daily life in order to create a virtual and digital memory of it. As our brain is designed to forget, humans have developed alternatives for persistent memory, like drawings, writings or photographs. These tools have artificially extended our capabilities for knowledge discovery and transference. Wearables cameras are new tools that take one step further, by allowing a much more finer visual memory.

Our days have an average of 16 hours awake, meaning that correspond to 960 minutes in one day. Lifelogging cameras usually takes a photo every 30 seconds. This acquisition rate implies that they generate around 1.400-2.000 images per day in average. This is a large amount of data, which generate around two gigabytes per day of image storage.

In addition to this big data problem, most of these images are unintentionally taken, so many of them might be blurred (because a camera wearer is moving quickly), with little information, fast illumination changes…

To sum up, wearable cameras define a problem of big and noisy visual data that require automatic tools for the indexing and retrieval. This work presents new tools in this field, by introducing the concept of visual memorability and the contextual data captured from physiological signals.

This bachelor thesis is a natural evolution of the previous theis of Ricard Mestre\(^1\) and Aniol Lidon\(^2\) in exploring techniques for automatic analysis of egocentric images in collaboration with the Consorci Sanitari de Terrassa and Universitat de Barcelona.

\(^1\) [Ricard Mestre work](https://imatge.upc.edu/web/resources/visual-summary-egocentric-)

\(^2\)
The long term goal of this partnership is the creation of visual summaries to exercise memory for patients with mild dementia (early stages of Alzheimer).

1.1 Objectives

As this project has three main contributions, three main objectives motivate the work. Due to the three contributions are different between them, each contribution is presented in an independent chapter self-contained with its related work, methodology, results and conclusions. Despite this independence all contributions are focused to visual memorability study, so a final conclusion relating chapters is presented.

Next subsections explain in a general way these three motivations.

1.1.1 Annotation tool for visual memorability in egocentric images

The study of visual memorability of egocentric images requires a ground truth of the probability that humans will remember this image. This probability can be estimated by showing the images to a group of human annotators and measuring how well they remember them. Despite this easy description of the task it is not trivial to measure this property of images. A specific visual memorability a tool for manually annotate images from a visual task is needed. For that purpose, a web application will be designed based in previous work [4].

1.1.2 Adaptation of a model for visual memorability prediction to egocentric images

Previous works [1] in computer vision have shown that egocentric images have a different composition compared with human-taken images. As a consequence, algorithms designed and trained with human-taken images perform poorly with egocentric images. For that reason, the second objective of this work is to study the performance of an off-the-shelf model for visual memorability over egocentric images, as well as adapting it to the egocentric domain. This adaptation will be based on the annotations collected with the tool described in the first objective.

---

1.1.3 Correlation between physiological signals and visual memorability

The emergence of wearable cameras has been accompanied with several other wearable sensors capable of tracking physiological signals of the user, such as the heart rate of the galvanic skin response. We hypothesize that these signals may be affected in some cases by the field of vision of the user and, so, also related with the visual memorability of the captured images. If exists a correlation between them we will be able to estimate visual memorability directly from physiological signals and detect snap points as the most memorability images.
Chapter 2 – Annotation tool for visual memorability

2.1 Introduction

The study of visual memorability requires knowing the probability with what a human can remember an image. A specific annotation tool for manually annotate image visual memorability with human interaction is needed.

This part of the project was inspired from the work [4] where the memorability of human-taken images was explored. In their work an application to annotate images was develop but the authors have not made available neither the tool or its source code. For this reason, it was necessary to reimplement it from scratch.

The main assumption of this chapter is that an image is memorable if a user can detect its repetition when the image is seen for a second time. As shown in [4], not all the images are equally memorable and those images which are more memorable have a similar structure or composition.

2.2 Related work

The memorability of an image is a topic that has appeared recently in the research community. These works [4], [5], [6], [7] and [8] study what makes an image memorable and explore visual features and composition of the image.

Decide when an image is memorable or not can be interesting for advertising, photography, etc.

In our case, memorability is useful because from a day, we want to select those images that are relevant and we will assume that an image is relevant if we can remember it. So, previous work from [4] define that an image is memorable if when we see an image for a second time, we can detect that it is a repetition. This assumption seems to be coherent. The authors of the research did an experiment with many users. The task for the users was to a visual memory game. The game consists in an application that shows images during 5 minutes. They define two types of images: the targets are the images that they want to annotate. The are also fillers, the images that they put between targets. Targets are shown only twice, while fillers can be repeated many times. Some of fillers are considered for
vigilance, to ensure that a user is paying attention in the game and the results of the game can be used in the research.

With the data collected during the game, they compute a memorability score for each image that consist in:

\[
\text{memorability} = \frac{\#\text{detections}}{\#users}
\]

For each image they obtained a score value. For example, if the value of the memorability score was 0.6, that means that 60% of users had detected the repetition of that image.

All this works can be revised on their website. There it is possible to find all the papers and an online web application that computes the memorability score from an image. Also, there is some information about the API. The code for the convolutional neural network is publicly available and has been used for this project.

### 2.3 Egocentric visual memory game

This egocentric visual memory game was inspired in MIT research and consists in a web application to manually annotate image memorability. The implementation is a HTML source code that a part from text contains a command to show an image. Images change 1.2 seconds controlled by JavaScript algorithm. So, the main complexity of the game is in this script.
2.3.1 JavaScript algorithm

This visual memory game is a web application generated with HTML and JavaScript, that generates the random sequence of image (set of targets and fillers) but ensure targets will be repeated twice and separated by a certain number of filler images. Also generates lists for control target detection, and user attention based on vigilance fillers.

All the details about implementation details can be found in the annex I.

2.3.2 Image selection for the game

Egocentric images are different from human-taken images and present some problems. The main problem of these images is that neighbour images can be very similar, so if there is a target or filler that is similar to another target user can detect a repetition erroneously. So, a first step is to select these two groups of images with some criteria. All these egocentric images have been taken using wearable cameras.

For targets we need a total amount of 50, so a uniform sampling of the dataset was done in order to take images well temporal separated. By this way we ensure that targets are enough different between them. Few targets can be similar sometimes but contains some details that allow us to find a difference between them.

For fillers, to ensure that are very different from targets another dataset of images was used. This dataset is available publicly: UT Egocentric that was created for a research in [1] to detect snap points. That dataset consist in four videos of about four hours each one.
Video was sampled extracting frames with a fixed interval of time, as were captured by an egocentric camera. The quality of these frames is more or less similar than dataset used for this work. By this way we can ensure that no filler is similar to a target, but these images must have the same point of view than targets to make that users cannot distinguish fillers and targets. The number of fillers is higher than targets to achieve a non-repetition of fillers and focus user attention on targets.

Figure 1 shows some samples of targets and fillers, to have an idea of how these images look like.

![Fillers](image1.png) ![Targets](image2.png)

*Figure 2. Example of fillers and targets*

Privacy considerations are specially important because in egocentric images, taken with any intentionality often faces, car plates… appears and we can store these images without any permission. This is not illegal, but the problem appear if we share this images, so it is critical to submit any research results where this kind of images can be shown. So special cares must be taken when we decide to work with a certain database of images.

### 2.3.3 Docker platform

The annotation tool was set online to allow the access to multiple annotators from their web browsers. In this case, computation services in the Image Processing Group (GPI) at the Universitat Politècnica de Catalunya (UPC) deal with that. A server was set up with the game configuration to allow an online use of the game.

A common implementation of this kind of servers consists in running the algorithm in server machine and starts a server process. A problem may occur if there are some incompatibilities with software versions. To avoid that problem, a new technology to set up
a server was introduced in GPI. Based on the idea of a virtual machine a new concept has appeared: *docker*.

A “*docker*” is a container where an operative system is placed and all files and software required for the use are added. With that system a *docker* image was created, being a virtual configuration of the required machine.

Docker containers wrap up a piece of software in a complete file system that contains everything it needs to run: code, runtime, system tools, system libraries – anything you can install on a server. This guarantees that it will always run the same, regardless of the environment it is running in. Containers running on a single machine all share the same operating system kernel so they start instantly and make more efficient use of RAM. Images are constructed from layered file systems so they can share common files, making disk usage and image downloads much more efficient. Docker containers are based on open standards allowing containers to run on all major Linux distributions and Microsoft operating systems with support for every infrastructure. Containers include the application and all of its dependencies, but share the kernel with other containers. They run as an isolated process in user space on the host operating system. They are also not tied to any specific infrastructure – Docker containers run on any computer, on any infrastructure and in any cloud.

A benefit of that technology is that a docker image (built from a Dockerfile) can run on any operating system without having incompatibilities.

For more details about Docker implementation see the annexes.

### 2.4 Conclusions

In this chapter a visual memory game has been presented, a tool for manual memorability annotation of egocentric images. Technical features have been described and server setup has been presented as a Docker implementation.

In next chapters of the project this tool will be used to annotate images.

---

4 Docker web site - https://www.docker.com
Chapter 3 – EgoMemNet: Visual Memorability Adaptation to Egocentric Images

3.1 Introduction

Visual memorability of images is a novel field of research that has been explored for human-taken images. Previous works from egocentric field [1] highlight that using algorithms that have been trained with human-taken images over egocentric images may be possible, but that an adaptation to the egocentric domain will improve performance.

This adaptation, though, requires an annotation of a dataset of egocentric images based on their visual memorability.

This chapter uses the data collected with the annotation tool, as described in Chapter 2, and uses them to adapt a model of human-taken visual memorability to the egocentric domain.

3.2 Related work

As mentioned before previous works [4], [5], [6], [7] and [8] explores visual memorability of images. In these works with the tool for manually annotate visual memorability of images the authors trained a model for automatic memorability prediction. In [4] the model for memorability prediction was trained with hand-crafted features and in the other works a convolutional neural network for memorability prediction was implemented. The benefit of the convolutional neural networks (CNN) is that they learn automatically the features of the images and is shown in these works outperforms hand-crafted models.

The current state of the art model that has the best results is MemNet CNN model. To understand why a CNN models performs other algorithms is important to introduce this novel technique.

3.2.1 Machine learning and deep learning

Machine learning can be defined as sets of techniques automatically learn a model of the relationship between a set of descriptive features (or attributes, the input) and a target
feature (or attributes, the output, what we want to predict) from a set of historical examples. We use a machine learning to induce a prediction model from a training dataset.

This algorithms learn from image features, and there are a lot of different features we can select from images and different ways to manage this data in order to construct a model. But for all these techniques, features are handcrafted and we define a set of rules that the algorithm must apply from data, defining arbitrary threshold and operations.

A recent technique called deep learning outperforms all existing techniques. The implementation of deep learning known as convolutional neural network (CNN) have been extensively applied for image recognition, detection and segmentation problems. CNNs are capable of automatically learning complex features that cannot be designed for the programmer of the algorithm, features learned by the own structure, which tries to simulate brain architecture, achieve superior results performance to hand-crafted features.

**Convolutional neural network**

Convolutional neural networks are biologically inspired variants, as we know the brain contains a complex arrangement of cells. These cells are sensitive to small sub-regions of visual field, called receptive field. These cells act as local filters and are connected between them, structured in many layers, as shown in the next figure.

![Convolutional neural network layer structure](image)

*Figure 3. Convolutional neural network layer structure*

As we know that each cell is a filter, all cells that belong to a same layer shared the same weights, weights that will be applied to the input. In computer vision the input is an image and each cell corresponds to a convolutional filter that act to a certain part of the image. Many layers are connected between them, where the output of a layer is the input of the next layer, as we can see in the next figure:
How the network understand an image or process the image is difficult, but it is known that features obtained in different layers of a network works better than human-built features. As a need of understanding what a CNN see, some works have tried to represent the output of each layer, showing that if we go deeper in the network, features are more complex, and more details of image are understand, starting understanding textures, shapes and finally objects.

The previous example show what convolutional network see in case of faces, but in the next figure we can see what it sees for different categories:

In particular MemNet CNN model presents a convolutional architecture and ends with a fully connected layers. In particular, this network is composed by five convolutional layers (with many convolutional kernels in each layer and their correspondent layers for linear rectification and pooling layers to decrease the number of outputs or parameters) and two fully connected layers (where a neuron is connected to a single input; for example if the input is equal to 256x6x6, so equal to 4096, a fully connected layer has 4096 neurons. At the end of these structure there is a layer that does the final step, transform the multidimensional input into a single value output (memorability score).
3.3 Equipment and software

The following sections will introduce us to the technical context: devices used and available resources for that project.

3.3.1 Software

This part of the project used some machine learning and deep learning techniques and algorithms that allows us to explore and process the data.

Matlab

Matlab, matrix laboratory, is a multi-paradigm numerical computing environment and fourth-generation programming language. A proprietary programming language developed by MathWorks, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, Java, Fortran and Python.

In this part of the project Matlab is used to develop algorithm using a powerful library and framework for deep learning application.

Caffe framework

Caffe is a deep learning framework made with expression, speed, and modularity in mind. It is developed by the Berkeley Vision and Learning Centre (BVLC) and by community contributors.

Many deep learning techniques use this framework and can be applied without too much effort using standard functions created.

Graphical processing unit

Each computer has his own CPU (central processing unit), an electronic circuitry within a computer that carries out the instructions of a computer program. This part is essential and enough for normal computing uses, but deep learning techniques require a lot of processing and computation, so only with a CPU this algorithms takes long time of execution and might have a memory problem during computation.
3.4 Performance of MemNet CNN model over egocentric images

For this part of the project, fifty images were explored and studied, using them as targets for the visual game. After obtaining results and computing the memorability score, a number of twenty-five users were considered. By this way, if the definition of memorability is used, the resolution of memorability score is 0.04 points (as the score was computed from twenty-five users).

Once score was computed for each image, a quick comparative of the scores can be done as next figure shows, where a 2D graphic plot predicted memorability score (with convolutional neural network model) against manual annotated score (with visual memory game).

![Figure 6. Predicted memorability scores against manual annotated scores](image)

From the figure a difference between predicted and manual memorability score is clear since points are not following a linear pattern as we expect if a correlation of values exist, but some statistical results can be explored:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Average annotated score</td>
<td>0.7294</td>
</tr>
<tr>
<td>Average predicted score</td>
<td>0.7687</td>
</tr>
<tr>
<td>Max score difference</td>
<td>0.2797</td>
</tr>
<tr>
<td>Min score difference</td>
<td>0.0024</td>
</tr>
<tr>
<td>Mean square error</td>
<td>0.5713</td>
</tr>
</tbody>
</table>

*Table 1. Statistical results*
As shown in table 1, the average score for all these images is similar between annotated and predicted. An interesting value is maximum and minimum scores differences, taking the absolute value of that difference to compare between images.

The lowest difference for this dataset is near zero, meaning that the score is well predicted for that image, and the highest difference means that for some images the predictor not works properly.

Figure 7 shows the two images associated to the maximum and minimum difference:

![Image with less exact prediction](image1)

![Image with most exact prediction](image2)

*Figure 7. Image with less exact prediction (maximum difference) at left and image with most exact prediction (minimum difference) at right*

The first image corresponds to the image that has the highest difference of scores. It means that the real score (score obtained with user manual annotation) are very different from the score assigned (or predicted) by the convolutional neural network. This image, like the others, was taken in an egocentric point of view. The second image has the same score with both models. We hypothesize that the minimum score is due to the composition of the image, which is similar to many intentional images that contain roads, green and cars. On the other hand, the image associated to the highest difference depicts the back of the seat in a plane, which is rarely chosen to be captured by a photographer.

In the next section, the data collected for egocentric images described in Chapter 2, is used to adapt MemNet to the egocentric domain.
3.5 Domain adaptation of MemNet to egocentric vision

This section presents the adaptation of MemNet to the egocentric domain. Although, from the visual memory game few images were annotated, so new model might overfit. There are many parameters of CNN to change and we can update all layers weights or only some layers.

As only 50 images were available for fine-tune, three different strategies were proposed: the first strategy was fine-tune the CNN giving only fifty annotated images. The second strategy was to do visual data augmentation. This process of an image consists in ten transformations: from an image, five crops were done (centre and the four corners) and from those crops, x-axis flip was applied. For this second way 400 images were available to fine-tune (due to set split) the network. The last strategy applied was to do a temporal data augmentation. The idea is the same as the second strategy, increase the number of samples available to fine-tune, but instead to use an image and apply a set of transformations advantage of egocentric problem was taken to increase the number of samples. This problem is the visual similarity of temporal neighbour images, so for each image similar temporal neighbours were manually selected and for them, the same memorability score was considered due to visual similarity.

The main challenge in the fine-tunning process was dealing with a regression case, instead of a classification problem, as addressed many of the existing publications and documentation. Fine-tuning a ConvNet in Caffe requires an input in hdf5 or lmdb format. The first format requires more memory as all images are charged first, instead of lmdb format, where it can be processed with a certain batch size (number of images processed at the same time).

3.5.1 CNN model evaluation

After many configurations of fine-tune algorithm many models are obtained. These models have been compared using different metrics. SAD metric (sum of absolute differences) and MSE (mean square error) were considered as metrics based on the predicted score against the score manually annotated. Also a metric based on the rank images was used to focus the evaluation of each model in the rank position instead of focus the evaluation just in the memorability predicted value. This measure is useful since this model can be used to summarize a set of images taking images with the highest memorability score (top N rank images). The metric based on rank is rank correlation; in
special as considered in MIT work Spearman’s rank correlation was computed. This metric between two ranks computes the position of an image in a rank and computes a correlation between these positions as next equation shows:

\[ r_s = 1 - \frac{6 \sum d^2}{n(n^2 - 1)} \]

Where ‘\( n \)’ is the number of samples in the rank (pairs of samples) and ‘\( d \)’ is the difference of positions of each sample between the two ranks.

To evaluate all models a test set was needed and this set is built from images that the algorithm has not seen before. To do that the fifty images set were split into forty images to train and 10 images to test. Data augmentation was applied for forty images and test set was not transformed.

**Evaluation results**

Next table shows results of fine-tunning process:

<table>
<thead>
<tr>
<th>Model</th>
<th>SAD</th>
<th>MSE</th>
<th>Spearman’s coef.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemNet</td>
<td>0.7478</td>
<td>0.00846</td>
<td>0.7727</td>
</tr>
<tr>
<td>Finetune_50_iter_25</td>
<td>0.7031</td>
<td>0.00766</td>
<td>0.7182</td>
</tr>
<tr>
<td>Finetune_50_iter_50</td>
<td>0.7542</td>
<td>0.00900</td>
<td>0.7061</td>
</tr>
<tr>
<td>Finetune_50_iter_75</td>
<td>0.7579</td>
<td>0.00910</td>
<td>0.7061</td>
</tr>
<tr>
<td>Finetune_50_iter_100</td>
<td>0.7582</td>
<td>0.00911</td>
<td>0.7001</td>
</tr>
<tr>
<td>Finetune_50_fc7_iter_25</td>
<td>1.0605</td>
<td>0.01574</td>
<td>0.7606</td>
</tr>
<tr>
<td>Finetune_50_fc7_iter_50</td>
<td>1.1067</td>
<td>0.01699</td>
<td>0.7606</td>
</tr>
<tr>
<td>Finetune_50_fc7_iter_75</td>
<td>1.1094</td>
<td>0.01706</td>
<td>0.7606</td>
</tr>
<tr>
<td>Finetune_50_fc7_iter_100</td>
<td>1.1096</td>
<td>0.01707</td>
<td>0.7606</td>
</tr>
<tr>
<td>Finetune_VA_fc7_iter_25</td>
<td>1.1854</td>
<td>0.01977</td>
<td>0.7182</td>
</tr>
<tr>
<td>Finetune_VA_fc7_iter_50</td>
<td>1.5864</td>
<td>0.03100</td>
<td>0.7182</td>
</tr>
<tr>
<td>Finetune_VA_fc7_iter_75</td>
<td>1.6248</td>
<td>0.03233</td>
<td>0.7182</td>
</tr>
<tr>
<td>Finetune_VA_fc7_iter_100</td>
<td>1.6276</td>
<td>0.06242</td>
<td>0.7182</td>
</tr>
<tr>
<td>Finetune_TA_fc7_iter_25</td>
<td>1.0930</td>
<td>0.01635</td>
<td>0.8152</td>
</tr>
<tr>
<td>Finetune_TA_fc7_iter_50</td>
<td>1.1346</td>
<td>0.01751</td>
<td>0.8152</td>
</tr>
<tr>
<td>Finetune_TA_fc7_iter_75</td>
<td>1.1362</td>
<td>0.01756</td>
<td>0.8152</td>
</tr>
<tr>
<td>Finetune_TA_fc7_iter_100</td>
<td>1.1363</td>
<td>0.01756</td>
<td>0.8152</td>
</tr>
<tr>
<td>Finetune_TA_fc6_iter_25</td>
<td>0.8723</td>
<td>0.01085</td>
<td>0.8394</td>
</tr>
</tbody>
</table>

*Table 2. Evaluation results between convolutional neural network models.*
In the previous table models name corresponds to:

- Finetune_50_iter_X: model fine-tuned with forty annotated images. This number is due to set split (train and test sets) ‘X’ determines the number of iterations done. All layers had been updated.
- Finetune_50_fc7_iter_X: model fine-tuned with fitty annotated images where only the layer fc7, the last layer before regressor, was updated (his weights). ‘X’ determines the number of iterations done.
- Finetune_VA_fc7_iter_X: model fine-tuned with a visual data augmentation. A total of 500 annotated images were used for that. As in the previous model only layer fc7 was updated. ‘X’ determines the number of iterations done.
- Finetune_TA_fc7_iter_X: the same procedure as last model but changing the set for temporal data augmentation image set.
- Finetune_TA_fc6_iter_25: fine-tune with the temporal data augmentation but updating at this time fc6 layer, the sixth convolutional layer.

From results table we can see that the best result in Spearman’s rank correlation value is for the model created with a temporal data augmentation set and updating weights for layer fc6 (sixth layer).

Due to the low number of samples, as more iteration was done the error between predicted scores and manually annotated scores increase. We see that update (fine-tune) some layers are better than fine-tune all layers because as many layers there are the number of parameters increase. Also we can see that updating weights from only the sixth layer in the network is better than doing the same for the last layer (seventh layer) before regressor. Note that last layer (regressor) must not be updated because we can obtain values out of the normalised range (that goes from 0 to 1).

### 3.6 Memorability maps visualization

In this section a method to compute memorability maps will be presented. A memorability map is an image where each pixel has a normalized value between 0 and 1 related with the contribution of this region to the visual memorability of the image. By this way we can know what parts or regions of an images make it memorable.
Many methods for memorability maps computation will be presented and explored highlighted the main differences in the results.

Also, there is a map called saliency map that describe with an image where the human fix his eye gaze where he observe a scene in the firsts moments.

From memorability and saliency maps, it is interesting to relate both maps to know what parts of the image make it memorable.

### 3.6.1 Method

For this section we need to compute saliency and memorability maps for each image. Saliency maps have been explored before and there is one convolutional neural network model that obtains these maps from image. SalNet, a CNN model for saliency map prediction, will be used due to the good results obtained in a challenge for this purpose. The result of the algorithm is a grey map where saliency points are labelled with a value near to 1 while points out of the eye gaze have a label near 0.

Until now there is no algorithm or CNN model to compute memorability maps, for that reason three different methods for map computation will be explored.

**Memorability maps analysing image patches**

A first approach just using current convolutional neural network model for visual memorability prediction, EgoMemNet, is to partition the original image in a grid and pass forward each patch though the network. With this approach we obtain a single memorability value for a patch of image.

In particular each image of the dataset has a size of 2592 × 1936 pixels. For these approach all images were resized to 2497 × 1816 pixels. With that size each image was divided in regions of 227 × 227 pixels, size equal to the input size of the CNN model. The result of this method is an image with the same size than the input. Despite that, every patch has a different value and the memorability score between two connected pixels can be very different. For this reason a filtering of resultant memorability map is proposed.

As a filtering technique a Gaussian filter will be applied to the memorability map.

**Memorability maps with a fully convolutional network**

Convolutional neural networks have the capability to keep localization in images. This capability is due to the convolutional layers because there are built with convolutional filters that have local impact.
First approach to compute memorability maps not use this capability because localization was done making a grid of the image.

In this case memorability map will be directly obtained without grid the image.

With the structure of MemNet is not possible to compute that maps because the two last fully connected layers make loss partial information. For that reason a previous step is to change the model and the network definition (structure) to achieve a fully convolutional network. This process can be done in two different ways: the first way is to change the structure of the network (defined in a prototxt file) and fine-tune the original network. Model obtained fits with a fully convolutional structure. The other way is to make a “net-surgery”. This technique consist in create a new model that fits with the fully convolutional architecture based in a current convolutional neural network model and structure (in this case MemNet was used).

Having a fully convolutional neural network the approach is to visualize the output of the last convolutional layer before the regression model that computes the single value. For a correct memorability map visualization is needed to add a global average poling layer (GAP) between the last convolutional layer and the fully connected layer referred to the regression model (inner product layer). This new code can be seen in project’s Github repository.

**Memorability maps with probabilistic model**

Last approach consist in a novel probabilistic framework performed by MIT for automatically constructing memorability maps, discovering regions in the image that are more likely to be memorable or forgettable by human observers. This method requires the creation of a probabilistic model, being discarded in this part of the project because is a tricky implementation and the exploration of the capabilities of CNN is more interesting.

**Comparing saliency and memorability maps**

There are many ways to compare heat maps. For this work we want to know what regions that we see at a first time in a scene are directly involved in the visual memorability. To do that intersection maps will be computed, doing the multiplication between saliency and memorability maps. We want to obtain a map that have a value equal to 1 where saliency and memorability maps has an activation (values near to one in the grey scaled map) and a 0 value where maps has not in common. For that reason a pre-process of the
Visual Memorability for Egocentric Cameras - Marc Carné Herrera

maps is required. The processing is to binarize both maps before multiplication defining an arbitrary threshold.

### 3.6.2 Memorability maps

The first part of the section is to compute saliency maps. For that, using Python and Caffe framework, an algorithm was designed to compute an image with the same size than CNN input where each pixel has the probability that a human put his eye gaze there. *SalNet* CNN model was used for that purpose. Next images show the results:

![Saliency map example](Figure 8. Saliency map example)

As we seen saliency maps are focussed in regions where object appear, in image centre, etc.

The second part of the section is to compute memorability maps. Many methods was presented in the previous section and results are presented below.

**Memorability maps analysing image patches**

As described above, each image was resize and grid to pass forward each patch through the convolutional neural network to obtain a score for each patch. *EgoMemNet* CNN model was used in this part of the project.

To explain the results of this method we consider three images that will be used in all section to explain the results and compare between them:
After apply the method to these images correspondent memorability maps were obtained observing is a low resolution memorability map because a single value are assigned to a large number of pixels, as result of apply the convolutional neural network model to them:

![Images](image_url)

**Figure 9. Images to explain results in this section**

Before binarize the map is needed to filter it. To do that a Gaussian filter using Matlab was applied to each map and the values binarizated and the results obtained are showed below:

![Images](image_url)

**Figure 10. On top, overlap between original image and the memorability map computed with a transparency of fifty percent. On the bottom, raw memorability map for these images**
This filtering requires selecting some parameters for the Gaussian function. For this work a kernel size of 256 x 256 pixels was selected.

Memorability maps was also computed using MemNet (MIT) CNN model. Final memorability maps obtained are not quite different with respect to maps computed with EgoMemNet:

Following the approach, saliency and memorability maps were binarized and multiplied, obtaining a binary mask. Selected threshold for binarize was 0.2 in the range between 0 and 1. Once this mask was obtained for each image, the mask arithmetically multiplied original image. Intersection pixels have a value equal to 1 and the other pixels have a value equal to 0. When multiply only these pixels where have value equal to 1 in the mask keep the pixel value of the original image. The rest of pixels acquires 0 value.
The results for these three example images are:

![Figure 13. Visual intersection between saliency and memorability binarized maps](image)

In last figure, intersected regions between saliency and memorability maps show objects, faces, etc. but always partial occluded. More image examples can be shown in the Github repository.

**Memorability maps with a fully convolutional network**

As explained before, MemNet and EgoMemNet CNN models present a structure that is not fully convolutional, so cannot be used for memorability map estimation. For that purpose a fine-tune of the model was required. The procedure is the same as in the model adaptation to egocentric images, but in this case the file with the structure was changed replacing the two last fully connected layers by to convolutional layers.

Full resolution images were passed forward though the network. To plot heat map the output of the sixth convolutional layer was taken. Despite in a paper related with this method [11] the authors took the last convolutional layer, for our model that layer didn’t give valuable information, assigning the same value to each pixel.

Algorithm used, publicly released return a coloured heat map.

Due to the low number of images for fine-tune (as happen in the first part of the project) there is a high probability this new fully convolutional model overfit. For that reason a good way to see if this process has been made successfully is to create a random model and compare it with the ones trained with egocentric images.

Adding two new convolutional layers instead of the fully connected layers generate random maps. These two new layers are in this case layers: conv6 and conv7. After adding these two layers, a random initialization was set and the update of the weights is what fine-tune does. To create the random model the model with random weights has to be stored.
Having both models a visual comparison can be done and the results obtained can be seen in the next figure:

![Figure 14. Original image, correspondent heat map predicted with a fully convolutional neural network and heat map predicted for the random CNN model for the proposed images](image)

### 3.7 Visual memorability vs. visual saliency

In this last figure we can see how memorability maps computed with a random convolutional neural network model differs from the fully connected neural network model based in EgoMemNet CNN model. Also we can see how memorability maps created with fully connected model seems to has sense as highlight objects, faces, etc.

To do the same with the last section a transformation of the heat maps to grey scaled and binarized maps is needed. Using the same algorithm to obtain the heat maps is possible to obtain that maps:
Figure 15. Grey scaled and binarized memorability maps for the proposed images

After multiply binarized saliency maps (the same as used in the last section) and memorability map, this mask was applied to the original image:

Figure 16. Intersection between saliency and memorability map predicted with fully convolutional neural network model for proposed images

With the last examples we can see how objects, faces, etc. are the most memorable parts and we assume these regions contributes directly to visual memorability due to human see this parts first.

Another interesting thing to explore is if we can use memorability maps for saliency prediction. By this way the similarity between both maps will be explored. This approach is not the same as before: in the last section map intersection was done, but in this section with both maps the similarity will be computed.

3.7.1 Similarity method

In this subsection the method for saliency and memorability maps comparison will be explained and evaluated in the next subsection. There are multiple techniques or methods
to compare two maps, but in this case a new technique has been designed to know if the map is equal, saliency map has more region than memorability map or reverse.

For this approach is essential that both maps have the same size, be normalized and be scaled between 0 and 1. After any comparison a binarization of the maps is needed. In this case an arbitrary threshold will be defined to compare all the maps. This arbitrary value can be results differ, but applying the same threshold in both maps is a good approach.

In this case grey-scaled maps will be binarized with a threshold of 0.2. The main idea is to use both binarized maps to compare areas.

Proposed metric

Metric for map comparison has to have a range between 0 and 1, where 1 is the maximum similarity between two maps. Assuming two identical maps, the similarity value has to be equal to 1. Likewise if the maps not share any region with value equal to 1 (where is the saliency points or memorable regions respectively) the value of the similarity has to be equal to 0.

For similarity study there are many algorithms to compute similarity and correlation between two maps. In this case maps are not computed with the same information. Saliency map indicates these eye gaze fixations regions while memorability map highlight memorable regions of an image.

Whether the answer to response is to use memorability maps for saliency prediction, the use of direct map comparison is a good approach but in this case apart from this metric is important to understand which areas of the image contributes to the saliency prediction and in this way, relate retrieve if saliency are directly related with memorability.

By this way first metric is Jaccard index, a similarity metric, that measures similarity between finite sample sets, and is defined as the size of the intersection between these two sets divided by the size of sets union.

\[ J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}. \]

The value of this metric is increased when most of the area is shared in the two maps. To compute this value a map pre-processing is needed. This pre-process consists in: normalize the map to have values between 0 and 1 (for saliency and memorability map because the objective is to compare both maps) and binarize both maps with an arbitrary threshold. The value of the threshold has to be the same for both maps because they have
been normalized. Binarization of the maps is needed because Jaccard index will compute the similarity between two spaces in this case regions with a value equal to 1. Due to the arbitrary value of the threshold, the experiment will be done with different values (0.2, 0.4 and 0.6). Apart from the coefficient value is interesting to visualize the area shared between both maps and out-going regions.

A second metric used in many challenges as MIT saliency benchmark, will be used to compare directly grey maps. This metric is useful because any pre-process of the maps is needed, highlighting if it is possible to directly estimate or not saliency regions with just memorability maps. In particular two metrics will be used:

- **Similarity**: computed the histogram intersection and measures the similarity between two grey-scaled maps viewed as distributions.
- **Pearson’s linear coefficient (CC)**: computes the correlation between two different grey-scaled maps.

For this second approach the visual similarity between maps will be displayed for some examples.

### 3.7.2 Similarity results using Jaccard index

As said before, this section explores the similarity between saliency and memorability maps for the dataset used in this work.

Examples of the results are:

<table>
<thead>
<tr>
<th>Image</th>
<th>Saliency map</th>
<th>Memorability map</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="saliency1.png" alt="Saliency map" /></td>
<td><img src="memorability1.png" alt="Memorability map" /></td>
</tr>
<tr>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="saliency2.png" alt="Saliency map" /></td>
<td><img src="memorability2.png" alt="Memorability map" /></td>
</tr>
</tbody>
</table>

---

5 MIT saliency benchmark website - [http://saliency.mit.edu/results_mit300.html](http://saliency.mit.edu/results_mit300.html)
Figure 17. Comparison between saliency and memorability map. From left to right: original image, predicted saliency map and predicted memorability map

After compute saliency and memorability maps both were binarized as described before and as the next example provide:
Doing that method mean Jaccard coefficient for dataset images (fifty images) can be computed, but first as seen in the example binary memorability maps has a bigger white area (corresponding to ones) than saliency maps. That fact suggest fix a threshold for saliency and change the value of memorability map threshold. Testing for different threshold values for binarize the maps aims to explore how memorability maps can be used for saliency prediction.

Results are described in the next table:

<table>
<thead>
<tr>
<th>Mean Jaccard score</th>
<th>Memorability map threshold</th>
</tr>
</thead>
<tbody>
<tr>
<td>Saliency map threshold</td>
<td>0.2</td>
</tr>
<tr>
<td><strong>0.2</strong></td>
<td>0.3853</td>
</tr>
<tr>
<td><strong>0.4</strong></td>
<td>0.1597</td>
</tr>
<tr>
<td><strong>0.6</strong></td>
<td>0.0593</td>
</tr>
</tbody>
</table>

Table 3. Mean Jaccard score comparing binary saliency and memorability maps with different thresholds

Table 3 shows when memorability map threshold is increased having the value of the saliency map threshold fixed, similarity between binary maps increases.

That suggests a good approach for use memorability maps as saliency maps is to found this threshold that increase the similarity between maps. This similarity in this case is computed as the Jaccard index. To do that a threshold has to be learned to apply it to novel images. As exploration of this threshold a grid search was done for the twenty thresholds from 0 to 1 with a step of 0.05. This is a method for explore, not to train, so any partition of the data was done. If a training threshold is needed a partition of the data between train and
test must be done, using train data to train the binary classifier (threshold) and the test set to evaluate each classifier. Results of that technique are shown as a grey-scaled image, where white are the maximum similarity and black the minimum. In the x-axis of the image saliency threshold is increase from left to right and in y-axis of the image memorability threshold is increased from up to down, having in the top-left corner the value of the Jaccard index correspondent to saliency and memorability thresholds equal to 0.

![Image](image.jpg)

*Figure 19. Jaccard indices image, top-left corner corresponds to saliency and memorability thresholds equal to 0 and the step between thresholds is equal to 0.05.*

The maximum value of the Jaccard index is equal to 0.7812 and it corrensponds to a saliency threshold of 0.1 and a memorability threshold of 0.35.

To end with that section, a plot of the difference image can be useful. In this case next images show in green regions shared between saliency and memorability maps, in red the regions of saliency maps that are not included in the memorability maps and in blue these memorability maps regions that outgoes from saliency prediction, assuming a saliency map binarized with a threshold of 0.2 (between 0 and 1) and a memorability map binarized with a threshold of 0.7 (between 0 and 1) inspired by the last table.

![Image](image2.jpg)
Figure 20. Original image at left and coloured map at right. Coloured map has green region that are salient and memorable, red region highlight zones there are salient but not memorable and blue regions shows these zones that are memorable but not salient.

From last figure if we assign saliency map as a ground truth and memorability map the saliency prediction, we can compute precision and recall. Precision will be these parts in the memorability map that are saliency points. Recall will be regions that have been highlighted in the prediction as salient region.
The average over all dataset has been computed and the result are:

- Average precision: 0.6957
- Average recall: 0.9426

These results suggests and opens a new possibility for saliency prediction from binary memorability maps, having a results for a memorability map threshold of 0.35 equal to 0.6957 in precision, meaning 69.57% of memorability points in the binary maps are salient points, and 0.9426 in recall, meaning half saliency map is recovered using binary memorability map (threshold of 0.35).

To binary memorability maps for that purpose is needed to learn a threshold instead of apply an arbitrary value and a partition of the data is needed.

### 3.7.3 Similarity results using state of the art saliency metrics

This subsection explores maps similarity without any pre-process of the maps, just computing the similarity from grey-scaled maps. These metrics have been taken from saliency benchmark to compare ground truth saliency and predicted saliency, so these metrics compare two saliency maps.

For this part of the project that metrics are used for compare predicted saliency maps (with SalNet CNN model) as ground truth and memorability maps assuming can be used as saliency maps.

In particular two different metrics have been computed: similarity and Pearson’s linear coefficient. The first one explores the similarity of maps viewed as distributions. The second metric correlate both maps. Both metrics are represented between 0 and 1 where a value closest to one expresses a higher similarity.

Both metrics have been computed without any map pre-process. Next table shows the results.

To compare obtained results a baseline with a random memorability prediction was develop and these maps were compared with the correspondent saliency. The model for random prediction is the same used in the first part of this chapter (model was obtained assigning random values to the last two convolutional layers).
After apply the same metrics obtained results were:

<table>
<thead>
<tr>
<th>Metric</th>
<th>Similarity (SIM)</th>
<th>Pearson’s linear coefficient (CC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EgoMemNet vs. SalNet</td>
<td>0.6386</td>
<td>0.7187</td>
</tr>
<tr>
<td>Random vs. SalNet (baseline)</td>
<td>0.5752</td>
<td>-0.0293</td>
</tr>
</tbody>
</table>

*Table 4. Mean value of similarity metrics between saliency and memorability maps*

These results highlight that not just one measure is enough to compare grey maps. As we can see, the correlation between maps is inexistent (correlation within random lists is equal to 0, as in this case). The value of similarity (SIM) is quite similar for both convolutional neural network models because the distribution of both maps is explored in that metric. Once a random model has been tested and as a conclusion, the similarity between saliency and memorability maps seems to be high and what is most important, both maps are very correlated.

These results open a new way to compute saliency maps from visual memorability of images. Despite that, a pre-process of memorability map should be done to perform the results and to have more precision.

### 3.8 Memorability and sentiment analysis

There are many convolutional neural network models currently available. Each model is specialized in a task. *EgoMemNet* model, the last work contribution, computes a memorability score from an egocentric image.

Some clinical research papers in memorability field conclude memorability of an image in humans increase when this scene has a negative sentiment associated.

To demonstrate that fact, a convolutional neural network for sentiment prediction will be used to compute the sentiment of each dataset image and Spearman’s rank correlation between sentiment score list and memorability score list will be computed.

#### 3.8.1 Method

A current convolutional neural network model allows us to compute a score for image sentiment. This CNN was trained with annotated twitter images dataset. Despite this
A convolutional neural network was trained with human-taken images; it will be used to compute sentiment score for egocentric images.

As in the other sections and previous part of the project, to evaluate the correlation between this computed score and memorability score we use Spearman’s rank correlation metric to relate positions in lists. For memorability, manually annotated scores will be used instead of predicted scores.

This convolutional neural network gives two values as an output. The first value is the probability that image has a positive sentiment associated and the second value is the probability that image has a negative sentiment associated. These two values are normalized, so the sum of them is equal to 1.

3.8.2 Results

After forward pass each image through the network and take only the first output value (probability that an image has a positive sentiment associated) we compute rank positions for all images in terms of image memorability and sentiment prediction.

Result obtained was a Spearman’s rank correlation of -0.0594. To understand this value we have to compare it with the Spearman’s coefficient of a random assignation of sentiment scores to images. In this case the coefficient is equal to 0. That means that there is not correlation existent between memorability score and sentiment score associated to images.

With this results it is not possible to ensure scenes with negatives sentiments are more memorable for humans.

With this approach a sentiment score associated to each image has been computed. Having manual annotated memorability score related to images (Insight Dataset), a correlation between these two values was retrieved. After take as a metric: Spearman’s Rank Correlation we obtained a value of correlation equal to a random assignation of sentiments. For this set of images, negatives sentiments cannot be related with high memorability of images.

3.9 Conclusions

In this section many techniques for memorability maps prediction have been designed and implemented. A comparison between these different memorability maps has been done showing how memorability map predicted with a fully convolutional neural network outperforms the others techniques.
Using saliency and memorability maps, a visualization of memorable regions that are seen in a first instance; looking for these regions of the image that have a largest contribution in the memorability. Regions with objects, people, faces, etc. are these more memorable regions where human eye gaze is focus.

Memorability maps for saliency prediction was explored. Two different metrics were applied to compare saliency and memorability maps: the first one, Jaccard index, compares regions and requires binarized maps. As part of the results this binarized maps suggested a future work for direct use of memorability maps for saliency prediction applying a learned threshold (this threshold achieves the best Jaccard index comparing both maps, assuming just memorability threshold change). The second metric consists in use current metrics used in saliency challenges. In this case two different metrics have been used, using directly grey-scaled maps. As one metric is not enough to compare saliency and memorability maps a metric to compare maps viewed as distributions (Similarity, SIM) and another metric to compute the correlation (Pearson’s linear coefficient, CC) between maps were used. Results were compare with the results obtained with a random convolutional neural network model. Similarity (SIM) measure was quite similar in both CNN models but Pearson’s linear coefficient was very different showing the performance of memorability maps over random maps to predict saliency maps.
Chapter 4 – Visual memorability and physiological signals

4.1 Introduction

As we have seen previously, current works try to deal with the objectives detailed in this project with visual features, hand-crafted or obtained with deep learning techniques. This works well with human-taken images, but egocentric images are related with a certain persons and as a digital memory, not all images present the same level of memorability or relevance. Applying existing works to this kind of images, we can identify some of the desired images, but may be the detection of this key frames can be done with more precision using physiological signals.

This idea appear knowing that egocentric images are taken with a constant interval of time so that images are objective, because have been taken without any criteria or intentionality. Comparing these images with human-taken images, which present certain intentionality, we can try to involve people to them own egocentric images but without any direct interaction. This idea is to use physiological signals at the same time that visual features and detect that snap points\textsuperscript{6}.

But not all physiological signals can be used for this purpose because we need signals that could be easily measures and be less invasive as possible. Taking advantage of technology improvements, different wearable devices will be explores to see what they can offer and study if this kind of features can be easily used for detect snap points.

Actually, there are not much datasets of egocentric images and sets that contain physiological signals and images are very difficult to retrieve. In DCU (Dublin City University) there is a centre for data analytics called Insight Centre. At this centre, where this project was developed, there are experts on egocentric topics like Dr. Cathal Gurrin, supervisor of the project, who from many years ago wears an egocentric camera with him all the time, having a valuable database. One of his databases consists in a big number of images that have physiological signals. Gently Dr. Gurrin gave his dataset for this project.

\textsuperscript{6} Image that look like intentionaly taken.
4.2 Insight dataset

The dataset available for this research consist in over than 20 days of images and physiological signals. Images present a lot of activities and actions. All this egocentric images had been taken with Autographer egocentric image.

Dataset belongs to EyeAware a tool for egocentric images developed in the same centre and allows researchers to explore data. EyeAware is a platform technology that supports any Google Glass (or equivalent) developer to create context-aware apps that understand what the user sees and make this searchable historically. This is achieved through developing the EyeAware API, based on extensive DCU know-how, IP and experience. EyeAware accepts images uploaded from apps and generates meaningful contextual information from the images.

The dataset contains a lot of images but restrictions of privacy have to be considered. For that reason the dataset was reviewed and cleaned and finally a total of 16228 images could be used.

With these images a text file was given. This text file contains some information per image: image identification number in EyeAware, heart rate, galvanic skin response, calories, date, image name in the database.

![Figure 21. Example of a text file with metadata information that includes physiological signals](image)

The first problem that appears was empty image names of many samples, so it means that not all physiological signals had image related, for example because the user...
only wears the wearable device and not the egocentric camera. For that reason, was needed to retrieve and store only images that had them physical signals associated. All this data exploration was do with Matlab, creating my own scripts. After select images that satisfy: have signals associated, belong to image that follows privacy statements… a total of 8474 images remain for the research.

Before start exploring data, the visualization of how the signals look like is the first step. Using Matlab raw signals were plotted as we seen below:

![Graphs showing galvanic skin response and heart rate values](image)

*Figure 22. Plot of galvanic skin response at left and heart rate values at right for this new dataset*

First impression of data is that is noisy and low informative, but it is because data had to be cleaned. As heart rate was very explored by clinical research compared with galvanic skin response, heart rate data was filtered. A simple filtering process was applied, deleting all sample that has a heart rate value associated equals to 0, because it means the person is not alive. This fact can be related with sensor error.

After do that, data presented other structure,
Figure 23. Filtered heart rate values

Figure 24. Filtered galvanic skin response values

Figure 25. Filtered heart rate and galvanic skin response for a certain interval of samples
So this data was considered useful for this research, a total of 7652 images.

4.3 Method

The retrieve of relevant images can be done by different ways. In this project, we will assume that a snap points is an image that has been taken with intentionality and that should be memorable for the owner. So snap point will be related with memorable images. Assuming that previous works in memorability for predict memorability score for egocentric images were used. The code for convolutional neural network such is available\(^7\) for the research. This code run in Caffe framework, available for Matlab, and this implementation is faster with GPU use. In this project, virtual machines were used, where Ubuntu and Caffe were installed. Once installed Caffe network architecture and Caffe network model were imported. The first file corresponds to a text file that specify how many layers the network have and how many convolutional filters (cells) are in each layer. Also include the size of the input and the output of the network. These two parameters are very important because a Matlab script to initialize and run the CNN must be created, so input requirements have to be considered.

Matlab script uses functions of Caffe to initialize the CNN and with a simple command we can obtain the result for a simple image, but we have to take care of the next paragraph in the deploy.txt, the file that specify the architecture:

```plaintext
name: "MemNet"
input: "data"
input_dim: 10
input_dim: 3
input_dim: 227
input_dim: 227
layers {
  name: "conv1"
  type: CONVOLUTION
}
```

\(^7\) Source code available - http://memorability.csail.mit.edu/download.html
We have to see that this implementation requires an input image of 227 height pixels, 227 width pixels and 3 channels. This is image requirements but a forth number, ‘10’, means that we have to give 10 images to network. After review the work from MIT, they obtained the best results when use 10 transformations per image and did the mean of 10 memorability score to compute the global memorability score for an image. The same criteria was followed and using a common used function 10 transformation of each egocentric image were done.

These transformations are: crop of the centre for specified dimensions, crop of 4 corners and their x-axis flips. The results were saved in a text file where the first field was the name and the other field the score obtained.

After process all images, the follow step is to plot this memorability score against heart rate value and galvanic skin response value in order to find some correlation.

### 4.4 Relation between memorability and physiological signals

After compute memorability score for all dataset images described before, a single value was obtained for each image, as the next figure shown. In the x-axis we see the index of image in the list and in the y-axis the value of memorability score. This plot allows us to see the variability of the memorability score.

![Memorability score variation for the dataset analysed](image.png)
With predicted memorability score results and physiological signals a matrix was created, where each row corresponds to a sample of the dataset and each column to: image name, heart rate value, galvanic skin response value, memorability score and identification of each image in this experiment. The identification number is very important because was assigned in temporal order and when a sort rows was done to the matrix, so the identification of images was well done.

4.4.1 Analysing obtained data

A first step was to sort the matrix by memorability score. Doing that a new matrix was obtained. The exploration of the most memorable and less memorable images predicted by the convolutional neural network was the first step. A total of 500 images of both cases were selected and plotted the value of memorability score, to see if most and less memorable images had different values.

![Histogram of memorability scores](image)

*Figure 27. High and low memorability scores. In red the five hundred highest values and in blue the five hundred lowest values.*

From the graphic we can look that the values differs but we can see also that there are a lot of images that have more or less the same values of memorability score, and these images can be very different between them. For that reason the memorability histogram was computed, to have a general perspective of the values,
and it shows that a lot of images had a memorability score value near 0.8. This means 80 people of 100 can remember this image in a second repetition, as authors of memorability works (from where this study was based) said. This fact made me think that the algorithm from MIT cannot be applied to egocentric images, since that model had been trained with human-taken images, as mentioned before.

### 4.4.2 Visualizing memorability prediction results

Before continue with the research a visualization of images with high and low information was needed.

*Figure 28. Memorability score histogram for the dataset*

*Figure 29. Most memorable images at top and less memorable images at bottom*
Last image shows us ten images with high memorability score at top and ten images with less memorability score at bottom. This images had been taken by the next method: the most memorable image was taken and then the next memorable image in the sorted list with a slope (image not taken) of ten images was taken, and this process was repeated until had ten images. The same method was done for less memorable images. As we can observe in the images, most memorable images are these images that contains objects, persons, centred scenes… despite images with low memorability score are these images that contains landscapes, outdoor scenes, etc. This results are the same that authors of memorability study obtains, so that prediction of network was assumed as valid. Although many images with intermediate memorability score was analysed sampling all dataset and a general pattern was identified.

From that results next sections of project were inspired, being a good idea to outperform CNN for egocentric images.

4.4.3 Searching a correlation

Having these predicted memorability scores and physiological signals a pattern in heart rate and galvanic skin response in relation to memorability score was searched. A plot of memorability score against heart rate for high and low memorable images in the same graphic shows us that for these two subsets of image, predicted values are mixed.

![Figure 30. Plot of heart rate value of the most (in red) and less (in blue) memorable images](image)
Red dots are the most memorable images and blue dots are less memorable images. For example, if we want to decide if an image with a heart rate of 100, from this plot we cannot have an answer. From that plot we can think that as prediction might have an error and sensors for data acquisition can introduce noise, try to find a correlation image-by-image has no sense. From here, the idea of organizes data into groups started.

Following previous steps the same graphic as before was done but considering that all most memorable images have the same memorability score, without establish an order between them, doing the same assumption for low memorable images, two new matrix were constructed for high and low memorable images and the value of heart rate was sorted and plotted.

![Figure 31. Plot of sorted heart rate values for the most (red) and less (blue) memorable images](image)

In this figure we see that a pattern start to appear, and knowing that red values are high memorable images and blue values are low memorable images the graphic shows how the range of values for high and low values are similar but with an offset. That fact encouraged me to definitely group samples. To do that, data had been separated into bins. A bin is a group of images that have same value or are similar in something. In my case bins from memorability score were done, doing quantification of the predicted value.

To divide data into groups a bin size was defined, defining how to group the data. The first division was into 8 groups or bins, dividing the range of memorability score between 0.5
and 1 each 0.05. After clustering was done, each bin had a different number of samples. It has sense because in memorability score histogram we saw a lot of images have the same predicted value.

As mentioned before, data was cleaned deleting images with heart rate associated equal to 0, but it’s possible that images with very high value of heart rate are noise. So a second filtering of data was needed, but this time a different filtering technique was applied. In image processing, a filter is a convolutional filter that applied to an image modifies the content. A mean image filter is a convolutional filter of N*N dimensions with the same value in each position. This is the same that do the mean of a matrix. In this case the filtering of clustered data was done applying the mean to each bin. Doing that we can “delete” noise having that correct values (that as supposed are most of samples) have more weight. The mean value of a bin is not the only filtering technique used in this project. There is another filter called median filter. This is not a linear filter; there is any operation with the values of the bin. This filter consists in sorting the values of a bin and selects the central value. With this method we can discard noisy values and take that value that is common in data.

For each bin the mean value and the median value was computed. This two values obtained was very similar in some cases. The use of these two values is motivated due to find a pattern valid in these two cases.

To have a general point of view of cluster data, the follow graphic was plotted. In this graphic there are 3 plots.

First mean values per bin had been explored. The first graphic corresponds to the mean value of memorability score of each bin. The second graphic is related with the mean value of heart rate for each bin and the same is plotted in the third graphic, in this case with galvanic skin response.
First of all, we can directly observe that memorability score grows, as we expect because each bin is a group of images that has a high value of memorability score each time. With only eight bins we data seems to follow a pattern. Results seem to tell that when memorability score grows, heart rate and galvanic skin response tend to decrease. Despite that observation, our dataset have about 8,000 images, so eight bins cannot be significant to split data. We have to remember that eight bins means a slope of 0.05 when the bins were done and that all bins have not the same number of samples.

To go deeper, high number of bins was done. To do that the size of each bin was decreased, assuming that the difference of memorability score between the samples of a bin are very small. Next plot shows what happen when a bin size was decreased to 0.03. That means have eleven bins.

Figure 32. Memorability scores per bin, mean heart rate score per bin and mean galvanic skin response per bin for bin size of 0.05 points
Figure 33. Memorability scores per bin, mean heart rate score per bin and mean galvanic skin response per bin for bin size of 0.03 points

As results show, the same pattern had retrieved, so has sense to think that memorability score are related with heart rate and galvanic skin response, where this two physiological signals seem to follow the same pattern for memorability score. Despite that observation this results have been obtained filtering data, not considering sample-by-sample, so it seems to be clear that we cannot work directly with these two signals values, so may be a process of original signal must be needed to work with a single sample.

To contrast these results and be sure that the results are valid, a fine exploration was done. It implies to reduce the bin size to 0.001, a very low value and see if data keep the same pattern. The results of that last assumption were plotted with dots to offer clear values because each cluster or bin has only a single value, that in this case is the mean value.
By reducing the range of a bin, the results seem to be sparse and unclear, but in general we can conclude that with an increase of memorability score value (in average), heart rate and galvanic skin response tend to decrease and this result cannot be applied to each single sample because with only one value of these signals for images are not enough. At time of data acquisition a processing of the value might has to be done, to achieve a mean value of signals at each time, as done for this data analysis.

We can also see that in the last plot, the first graphic is a line, so that means that the mean value of memorability score for each bin is the same value of the mean value of the range or the domain of each bin, meaning that data are good distributed along the bin.

A clear pattern has appeared when data has been filtered with a mean filter for each bin and with different bin size the results are similar, but to ensure these conclusions an observation of the results when a median filter was applied is a good idea.

Next graphic shows results for median filter application to data cleaning,
One more time, the results obtained are the same, despite taking median value galvanic skin response in all bins the result is equal to 0, meaning that the central value of this signal is 0. Those made me think that maybe this signal is not appropriate for being related with memorability score.

As mentioned before a good idea when data have to be acquired a good idea is to compute a mean value in the temporal domain. In this work data is acquired so a simulation of do that in a real time was done with the following approach: from a matrix with all clean samples without sort rows, a temporal feature is kept, so we can suppose that we are at data acquisition moment, having a difference between samples of about thirty seconds. In real situation for data acquisition this interval of time has to be small, but for my assumption is enough. With Matlab an average of the values for each sample was done. So for each sample his real physiological signals and the value of some neighbours was taken and the mean value computed, assigning that value to the sample. Different number of neighbours was taken, obtaining different mean value in each case. As high is the number of neighbours a low pass filtering is applied, so assuming that sensors can be very noisy, a very restrictive low pass filter was applied. It means that the number of neighbours had to be high. After test
with this value, a value of 50 neighbours per sample was decided to take. This number offers us a good influence of neighbours but keeping the value the original sample. This means to compute a value with 25 minutes data for only one sample (because data are separated by 30 seconds approximately). In real data acquisition separation between samples must be small to compute a value for a small range of time.

With that approach, computing a temporal value of the heart value and galvanic skin response, the same proceed that before was followed, grouping data in bins and compute mean and median value.

![Figure 36. Memorability scores per bin, mean heart rate score per bin and mean galvanic skin response per bin for bin size of 0.05 points applying a temporal averaging of scores after bin clustering](image)

Results are similar taking mean value or median. In last graphic value taken was from a mean filter, low pass filter, and the results are quite similar as the results that not take care the temporal values of signals. The conclusions for this approach are the same of the other procedure. For that approach we can conclude that a temporal filtering at data acquisition moment is a good option and would be implemented in future work.

Median value or mean value of heart rate can be computed without having significant changes but with galvanic skin response we have to be work with caution. For an
easy acquisition and success results in general heart rate might be used instead of galvanic skin response.

Clearly heart rate is related with memorability score but for a direct relation a process of this signal value have to be done in the temporal domain.

To end with this part, a plot of the images with high heart rate and low heart rate has been done to see if the memorability rule can be definitely applied. If images with high heart rate have the same visual composition of low memorable images and images with low heart rate have the same visual composition of high memorable images, a direct use of signals for a single sample can be used.

![Figure 37. Images with low heart rate value at top and images with high heart rate value at bottom](image)

In the previous image we can show at top images with low heart rate value, images that with my data exploration conclusion are more memorable; in the bottom images with high heart rate are shown, and this might imply that these images have a low value of memorability score, being less memorable. As we can observe in general top images contains objects, people, presents well-composition, etc. as memorable images look like, but despite general issue landscapes or outdoor scenes appear, meaning that this assumption introduce some noise that can be avoided doing temporal averaging for physiological signals in data acquisition. With bottom images the same happen: a lot of them follow low
memorability rules as being landscapes, low informativeness scenes, etc. but many images that should be memorable appear in that group.

4.5 Detect snap points with physiological signals

A snap point is an egocentric image that seems to be intentionally taken. This word was created in a paper that adapt an algorithm trained with human-taken images to egocentric perspective. As previous work shows, physiological signals follow a pattern when memorability score change, so a direct use of physiological signals will be implemented. This section aims to implement an algorithm to predict memorability score from heart rate value (galvanic skin response was discarded in the previous work because his value is not directly interpreted).

4.5.1 Method

Physiological signals belong to a user, only to a single user. If a direct relation between them and memorability exist a good goal is to predict memorability of images from just the value of some physiological signals, without any visual feature. This approach suggests a pre-process of egocentric images instead a post-process do it until now for the most of current techniques for image summarization. A part from allow to a pre-process also offer the possibility of implement custom systems, one for each user. Current techniques generalize over all users, so in many cases some error are done due to that.

Last work contributions are a model for image memorability prediction and a new egocentric dataset with annotations.

We want to implement an algorithm or system with the following features:

- Input: heart rate value.
- Output: memorability score prediction.

Machine learning algorithms for image for recognition of instances can be classified in four types, but supervised types, the ones that require annotations (as in this cases) are classified in: classification and regression problem. The first one consist in assign a label to a novel image using a trained model with examples. This label consists in an integer number where each number belongs to a class or category. These kinds of problems are discrete because there are a limited number of categories. The second type of problem, regression,
consist in assigning to a novel image a value from a range with a trained model with examples. This value assignment is continuous and the result is not referred to any class or category.

A regression model is basically a function that tries to fit the data with the least error possible and allows predicting a new value from a sample not seen before. There are many regression functions, but the linear function is the simplest one.

In this case, it is a regression problem because the output is not a labelled category but it is a float number between 0 and 1, where 1 is the score for the most memorable image. As it is a regression problem, a regression model has to be chosen. Two regression models were proposed: the first one is a linear regression and the second one is a regression model implemented with the best parameters and kernel for a certain set of data.

As in the previous work, due to the low number of samples, model can overfit. It can be observed as a low error in the train set but a high error in test set. It is due to the complexity of the model, many parameters to compute and low number of examples.

Dataset used has fifty images with their correspondent physiological signals values and the manually annotated memorability score. For train a regression model a split of this data was done, having forty images for train and ten images for test. Training algorithm for model train split train data into folders to search the best configuration with a k-fold strategy. Models obtained as a result were compared evaluating them with a test set and computing the same metric to compare convolutional neural network models: Spearman’s rank correlation. As this approach does not use visual content the same rank correlation values than MemNet or EgoMemNet cannot be achieved, so a rank correlation for random assignment of memorability score was computed. This random approach assigns to an image a random value between 0 and 1, so when a rank is computed, images have a random position in the list.

That dataset contains heart rate and galvanic skin response value associated to the images, but due to results obtained in the first part of this project only the value of the heart rate value will be used for this experiment. Galvanic skin response is a value that most time is equal to 0, so for that application that aims to be implemented at real time is difficult to use it. In next sections this value will be explored.
Model training was implemented in Python and the correspondent script can be found in the official\textsuperscript{8} GitHub repository of the project.

4.5.2 Results

Regression models for memorability prediction are evaluated in this section with the metrics described above and shown in the next table:

<table>
<thead>
<tr>
<th>Predictive model</th>
<th>Spearman’s rank correlation</th>
<th>Mean square error</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemNet</td>
<td>0.7727</td>
<td>0.00846</td>
</tr>
<tr>
<td>EgoMemNet</td>
<td>0.8394</td>
<td>0.01085</td>
</tr>
<tr>
<td>Linear regressor</td>
<td>0.2970</td>
<td>0.00892</td>
</tr>
<tr>
<td>Tuned regressor</td>
<td>0.0667</td>
<td>0.01707</td>
</tr>
<tr>
<td>Random prediction</td>
<td>0</td>
<td>---------------</td>
</tr>
</tbody>
</table>

\textit{Table 5. Regression models for snap points detection evaluation results}

From previous results a performance over a random selection of snap points is achieved, but no perform is observed with respect to post-process of images with a convolutional neural network.

As regression models rank correlation is high than zero, this system can be used to detect snap points but in some case, images with low information or interest for summarization can appear and images that present a high memorability can be discarded in the capture moment.

4.6 Adding physiological cues to memorability prediction

As a result of the first part of the project a new model for egocentric memorability prediction was presented, performing the actual state of the art achieved by MemNet CNN, a work from MIT research team. This new model allows computing the memorability score of images having best results over the ranked list of values that last best approach offered.

The fact that used images are egocentric it suggest to develop custom systems to image summarization that can help clinical research with cognitive therapies.

\textsuperscript{8} Official GitHub repository - https://github.com/imatge-upc/memory-2016-fpv
Previous section tried to develop a system with just the information of physiological sensors in a real time, offering a pre-process of the egocentric images. As the results are not enough for the application, next step is to develop a system that performs memorability prediction using physiological signals.

### 4.6.1 Method

The main idea is the same as in the last section: train a regression model to automatically predict the value of the memorability but in this case not only using physiological signals. To the values of the physiological signals, the value of the memorability score predicted for EgoMemNet, the convolutional neural network model presented in the first part of the project, will be added.

For the regression model train there were two different possibilities: the first one is to train the model with predicted memorability score and heart rate value as regression model input. The second option is to take the output of the second fully connected layer of the convolutional neural network and with the value of the heart rate value give them as regression model input.

For the first option we have:
- Input: 2 dimension vector (scalar values).
- Output: 1 dimension vector (memorability score value).

For the second option we have:
- Input: 4097 dimension vector (4096 dimensions belonging to visual features and 1 dimension belonging to physiological features).
- Output: 1 dimension vector (memorability score value).

Second option was discarded as visual aspects can have more weight in the regression decision.

The regression model chosen have two-dimension vector as input and one-dimension vector as output. In this case the first dimension of the input vector is the value of the heart rate and the second value is the predicted memorability score (using EgoMemNet CCN model). The output used during the training was the manual annotated memorability score.
As in the previous section, two regression models were trained: first a linear regression model with automatic parameters was trained and then a tuned regression model for the training set was trained doing a cross-folder validation for the best model search.

Model training was implemented in Python and the correspondent script can be found in the official\textsuperscript{9} GitHub repository of the project.

### 4.6.2 Results

After training both models, there were evaluated with the same metrics as in the previous section: Spearman’s rank correlation and MSE (mean square error).

Evaluation results over test set for a regression model that combines heart rate value with the memorability score predicted with a convolutional neural network are presented in the next table:

<table>
<thead>
<tr>
<th>Predictive model</th>
<th>Spearman’s rank correlation</th>
<th>Mean square error</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemNet</td>
<td>0.7727</td>
<td>0.00846</td>
</tr>
<tr>
<td>EgoMemNet</td>
<td>0.8394</td>
<td>0.01085</td>
</tr>
<tr>
<td>Linear regressor</td>
<td>0.8394</td>
<td>0.00547</td>
</tr>
<tr>
<td>Tuned regressor</td>
<td>0.6152</td>
<td>0.01447</td>
</tr>
<tr>
<td>Random prediction</td>
<td>0</td>
<td>-----------</td>
</tr>
</tbody>
</table>

*Table 6. Regression models for memorability prediction evaluation results*

Last results are very valuable because they show how a combination of heart rate value as physiological signal with the predicted memorability score not perform the result of use just the convolutional neural network for image memorability prediction. In the results linear approach has best rank correlation metric than tuned (adapted) model. That fact can be motivated for low samples and the complexity of the model (number of parameters to compute).

With these results linear regression model have the same value of Spearman’s rank correlation coefficient but presents a low error in score assignation. For this work application the most important is rank position of images instead of obtained score. Since the main application of that approach is to select the top-N images from a ranked memory

\textsuperscript{9} Official GitHub repository - https://github.com/imatge-upc/memory-2016-fpv
list, the real value of the score is not important. For that reason, results highlight add physiological cues to memorability prediction is not useful if we use heart rate value.

As a conclusion of this section, heart rate is a physiological signal that measures the number of heartbeats per minute; having a direct relation with motion, stress state, body activity… Galvanic skin response reflects the same but his values are rarely different to 0, so for that application this signal is not useful.

Add heart rate to predicted memorability score with a convolutional neural network for preform the regression problem was a good idea, but after train a regression model the conclusion is that adding heart rate the result for the ranking is the same than just use the result of the convolutional neural network.

4.7 Galvanic skin response exploration

The first part of this section shows how GSR (galvanic skin response) is not appropriate for memorability prediction. The electrical variation in the body is sometimes difficult to catch and this change is not produced with everyday activities, just sometimes is produced for something that makes user react. Despite that, in this section an exhaustive exploration of this value will be done.

4.7.1 Method

This approach wants to highlight if value of galvanic skin response are related with memorability score. It not means that all images that have a high value of memorability score predicted or manually annotated has to had a high value of GSR. What this section wants to demonstrate is if images with high value of GSR have also high value of memorability score. For that method dataset of fifty images will be used chosen ten five images with the highest galvanic skin response. Not only the manually annotated value: manually, predicted value with MemNet and predicted value with EgoMemNet will be inspected.

The original image source, from where Insight dataset was built, has over 7,000 images with their correspondent value, so these images will be also inspected, but in this case, only memorability score predicted with EgoMemNet will be explored.

4.7.2 Results

As a first experiment for each image from Insight dataset (fifty image) galvanic skin response was extracted. To this value three memorability scores of the image were added.
These memorability scores were: manually annotated, predicted with MemNet CNN model and predicted with EgoMemNet CNN model.

From all this images top ten images (with the highest GSR value) were selected and the results obtained were:

<table>
<thead>
<tr>
<th>GSR value</th>
<th>Manual memorability score</th>
<th>Predicted MemNet memorability score</th>
<th>Predicted EgoMemNET memorability score</th>
</tr>
</thead>
<tbody>
<tr>
<td>0,02</td>
<td>0,72</td>
<td>0,76</td>
<td>0,58</td>
</tr>
<tr>
<td>0,03</td>
<td>0,78</td>
<td>0,82</td>
<td>0,67</td>
</tr>
<tr>
<td>0,10</td>
<td>0,72</td>
<td>0,76</td>
<td>0,62</td>
</tr>
<tr>
<td>0,14</td>
<td>0,64</td>
<td>0,75</td>
<td>0,59</td>
</tr>
<tr>
<td>0,21</td>
<td>0,76</td>
<td>0,75</td>
<td>0,59</td>
</tr>
<tr>
<td>0,41</td>
<td>0,68</td>
<td>0,78</td>
<td>0,63</td>
</tr>
<tr>
<td>0,61</td>
<td>0,75</td>
<td>0,68</td>
<td>0,51</td>
</tr>
<tr>
<td>1,26</td>
<td>0,83</td>
<td>0,88</td>
<td>0,73</td>
</tr>
<tr>
<td>7,43</td>
<td>0,73</td>
<td>0,80</td>
<td>0,66</td>
</tr>
<tr>
<td>9,48</td>
<td>0,79</td>
<td>0,4</td>
<td>0,57</td>
</tr>
</tbody>
</table>

*Table 7. Relation between galvanic skin response and memorability values*

From these results we cannot extract any conclusion from the memorability score. For that reason the position of each item in the memorability ranked score was computed using Matlab framework. Using function

\[
\text{position} = \text{tiedrank(score\_list)};
\]

the result is a list where the each number corresponds to the position of each item in the score list, so if we sort score list and compute the position of each item, that vector would be obtained.

Position vector was computer for each memorability score for the list of all targets. This position corresponds to sorted list in ascendant order, so if a position is higher means that item has a higher value of memorability score.
From that vector indices corresponding to the top ten images with highest galvanic skin response were taken and the result is presented in the next table:

<table>
<thead>
<tr>
<th>GSR value</th>
<th>Position in manual scores list</th>
<th>Position in the MemNet list</th>
<th>Position in the EgoMemNet list</th>
</tr>
</thead>
<tbody>
<tr>
<td>0,02</td>
<td>23,50</td>
<td>19</td>
<td>10</td>
</tr>
<tr>
<td>0,03</td>
<td>32,50</td>
<td>42</td>
<td>43</td>
</tr>
<tr>
<td>0,10</td>
<td>23,50</td>
<td>21</td>
<td>29</td>
</tr>
<tr>
<td>0,14</td>
<td>11,50</td>
<td>18</td>
<td>14</td>
</tr>
<tr>
<td>0,21</td>
<td>29,00</td>
<td>17</td>
<td>13</td>
</tr>
<tr>
<td>0,41</td>
<td>19,50</td>
<td>32</td>
<td>31</td>
</tr>
<tr>
<td>0,61</td>
<td>27,00</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>1,26</td>
<td>40,50</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>7,43</td>
<td>25,00</td>
<td>37</td>
<td>38</td>
</tr>
<tr>
<td>9,48</td>
<td>35,50</td>
<td>14</td>
<td>9</td>
</tr>
</tbody>
</table>

*Table 8. Memorability ranks based in memorability score predicted in relation with galvanic skin response*

From previous results the mean position of the top ten items was computed. If these items that have the highest value of galvanic skin response the mean position value expected is 45.5 because the list have 50 items, so most memorable images are in the last list positions.

The value of the mean position for the three computed memorability score is:

<table>
<thead>
<tr>
<th>Memorability</th>
<th>Mean position of top 10 GSR images</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual</td>
<td>26.75</td>
</tr>
<tr>
<td>Predicted with MemNet</td>
<td>25.50</td>
</tr>
<tr>
<td>Predicted with EgoMemNet</td>
<td>23.90</td>
</tr>
</tbody>
</table>

*Table 9. Mean position of the top ten GSR images in the memorability ranked list*
Last results show that most images with highest value of galvanic skin response have not placed in the top ten memorability score ranking. For memorability scores computes manually only one image over ten (those have highest GSR value) is in the top ten most memorable images. For predicted memorability two images over ten are in the top ten most memorable images.

As final part of this experiment top 3 images with highest galvanic skin response were plotted,

*Figure 38. Top three images with highest galvanic skin response value*

Against top 3 images with highest manual memorability score,

**Top 3 highest memorable images**

*Figure 39. Top three images with memorability score value*

As results show images with a high value of galvanic skin response value have not a correspondence with a high memorability score value. Definitely this signal is not appropriated for memorability prediction.
4.8 EEG signals for memorability prediction

In the first section of this chapter a correlation between physiological signals and visual memorability of images was searched. As a conclusion of the methods designed using heart rate value and the convolutional neural network implemented in last work, there is not possible to establish a hard correlations and predict visual memorability with these values because heart rate value vary with user affective and physical state.

There are other studies that explore EEG (electroencephalographic) signals for perform tasks as image segmentation and user interaction. Actually this signal is a variation of brain signals when user detects a stimulus and is directly related with brain activity.

The possible acquisition of brain activity aims to explore that signal for memorability prediction due to be directly related with visual information processing and proposed it as a valuable cue for visual memorability study. In this section brain activity will be captured with under controlled environment and supervised by expertise in this field. After acquisition raw signal will be processed and compared with visual memorability value.

4.8.1 Method

The process for brain activity acquisition requires a complex set up because this is a weak signal. There are many devices for EEG signals acquisition. A wearable acquisition device consists in 4 channels that capture pre-frontal and lateral activity and have an approximate cost of 300€. Despite this device is useful for example for eye blind detection or eyes movement is very noisy and can give wrong results and values for this task.

Another acquisition device that is non-wearable consists in 21 sensors offering a fully brain activity capturing. Not all sensors are useful because the response to visual stimulus happens in certain parts of the brain and the signals of the rest of the parts have non-valuable meaning. An expertise in this field recommends using just seven channels corresponding to the central part of the scalp, from the front to the back, as red line shows in the next figure and also two sensors placed near the ears.
The method for this experiment is based in the visual memory game presented in the first part of the project. In this case a random sequence of images was built from the visual memory game, saving the order of images shown. This same order was used to set up the EEG experiment. Interval between images was the same than in the visual memory web game, but in this case blank image to separate images was deleted to reduce game duration. A total of 201 images were showed to the user including target repetitions and filler random visualization as part of the visual memory game.

User task consist in fix eye gaze in the images and see if an image was repeated. In this case user did not push any button, just see the sequence.

Theoretical assumption was that when a user detect a repetition of an image, EEG sensors could detect this event, so a difference between EEG signals of first and second repetition will be positive and should have a correlation with memorability score (manually annotated).

Once user saw the sequence and the EEG signal recorded, a process of data is needed. After revise some field literature from all signals the best one to capture this events is a channel called Pz (on the back of the scalp).
To do that, a visualization of the results from Pz sensor will be done to estimate the pick response and extract the response value from an interval of time. In this case, an interval of time will be delimited and the average of the values computed as a good metric of P3 signal, used to compute the event response.

After that process, a single value will be available for each image (a total of 201 values) and a correlation with manually annotated memorability scores will be searched.

It is important to highlight that the setup is difficult and may induct the experiment to bad results.

### 4.8.2 Results

For this part of the project, an as a first contact with EEG signals, a user that has seen the images before did the experiment. In a correct implementation of the experiment, users cannot see the images before because they can detect a first repetition as a second repetition and the results can be erroneous, but in this case, the most important part is to compute and know how to process data instead of accurate results, as is a first work with this kind of physiological signals.

The experiment was done twice and the results were stored for targets and fillers for separated. In this case, just target results were used as we have their manually annotated memorability scores.

Next figure shows the results for all target images over the time. In heat map, each row corresponds to an image and each column value to the brain response over the time when the image was shown. As a heat map, red positions correspond to a higher value and blue ones correspond to lower value. Second graphic show the average response value for all targets, and it show how a pick can be observed near 400 milliseconds. That value
suggests computing the response value for each image as the average between 350 and 600 milliseconds of the Pz signal, because this interval of time corresponds to the image event brain response.

![Temporal brain responses (columns) for each target image (rows)](image)

**Figure 42. Temporal brain responses (columns) for each target image (rows)**

To ensure that is an appropriate value, a plot of the temporal response for four random images will be plot below, between -200 and 1000 milliseconds.
In each image pick amplitude is different and sometimes lower than other temporal values but it is due to the brain response to each image. Area inside red square corresponds to the temporal brain response to analyse ensuring corresponds to image stimulus and not to any other event.

For each target image (those images to analyse) the average value of Pz signal between 350 and 600 milliseconds was computed. Each target image was shown twice in the experiment so two different values of the response are available for each target image. Taking advantage of these two values difference between them was computed and the assumption was that the second repetition has to be a higher response than the first visualization. To demonstrate that the number of targets over fifty that have a higher response in the second repetition was computed. From the experiment and for the fifty targets analysed (contained in the Insight dataset) 58% of these targets have a second response higher than the first repetition. This means not all second detections seem to be detected. That is not critical because it means not all images are memorable.

From difference of the response a correlation with manually annotated memorability score was computed. Spearman’s rank correlation found the correlation between positions in two ranked list as seen in other previous parts of this project. After compute the correlation the value obtained for target images was -0.0117, closer to random ranking of the images. It
means with the difference of average Pz signal is not possible to determine what image is more memorable.

Another interesting correlation to study is the ones between memorability scores and the brain response to the first visualization of each target. The result for that correlation was equal to 0.1150, a value higher than a random ranking (correlation 0), but not comparable to convolutional neural network model for memorability prediction performance. The reason for this correlation computation is that user had seen the images before, so the first visualization can be used as a seconds repetition with a large gap of time.

The same as before was done but in this case for the second repetition of the image, to see whether just the second repetition for each image can produce a higher response. The value of that correlation was lower that the correlation with the first visualization, equal to 0.0545, closer to the random assignment.

Results obtained suggest that the method for use this kind of physiological signals is correct but the experiment could be altered by the fact user seen the images before. For ensure results are useful the experiment must be done for many users and average the results.

4.9 Conclusions

Electroencephalographic signals, brain response, can be a valuable cue to predict image memorability. In this case visual memory web game was extended to an EEG experiment and a complex set up was to be done. After signal capture a processing was needed, computing just Pz signal between -200 and 1000 milliseconds and obtaining a response value as the average of the interval time between 350 and 600 milliseconds, the interval corresponding to an event brain response.

With the response value for all targets images three correlations were explored: correlation between response difference (first and second repetition) and memorability scores, correlation between target first visualization response and memorability score, and correlation between target second visualization response and memorability score. All these correlations computed as Spearman’s rank correlation show a low value, near zero in two cases meaning that prediction is comparable to random assignment.

This method is a good approach for study the correlation between EEG signals and the memorability score but more users are needed to establish a correlation and a user that has seen the images before can affect in a negative way to the experiment.
Chapter 5 – Conclusions

Egocentric images open a new branch for the research as they present a different image composition. First of all, egocentric devices provide us a large number of images per day that can be sometimes very similar between them and due to their non-intentional acquisition can be blurred, with no information, etc. Many previous works proposes approaches to try to select the most relevant frames from a set of images. In this case visual memorability has been taken as a valuable cue for select these most relevant frames.

Focusing in visual memorability study the first part of this project suggest a web application used for manually annotate a dataset of images. The result of the annotation is to assign a probability of remember each image (as the definition of memorability in the previous MIT works said). This visual memory game requires human interaction in a easy task. The annotation is done with the results of many users, computing the percentage of remembering. The application of the tool is not just to annotate images, can be used in a cognitive therapy with Alzheimer patients.

The second part and contribution of this project takes advantage of the novel tool for annotation and a current convolutional neural network (CNN) model for memorability prediction and demonstrates it need an adaptation to egocentric images due to de different composition between egocentric and human-taken images. As part of this chapter a new model, EgoMemNet, was published and performs the result of state-of-the art techniques. This contribution also aimed to design a novel technique for memorability maps prediction, allowing highlighting most memorable regions in an image. This maps were compared with saliency maps in a different ways and suggest to use memorability maps for predict saliency points.

In that section many techniques for memorability maps prediction were designed and implemented. Comparison between these different memorability maps has been done showing how memorability map predicted with a fully convolutional neural network outperforms the others techniques.

The possible estimation of memorability maps aims to relate saliency maps with memorability maps and the first question done was what regions that we see at first instance contribute to visual image memorability. The result of this method suggest that regions
containing object, faces, people, etc. present a higher value of memorability and that region belong to the salient region of the image.

A visual relation between saliency and memorability map suggest using memorability map for saliency prediction. For that purpose the similarity between saliency and memorability maps was explored in two different ways: the first one use a threshold to binarize both maps and the Jaccard index was computed. The same threshold was applied to saliency and memorability maps in a first instance but analysing the results the application of different thresholds suggest to be able to estimate saliency maps with a defined threshold. For a proper use of memorability map for saliency prediction this threshold should be learned from examples and manual annotation of images. The second method to compare both maps was to apply current metrics used in saliency prediction challenges where ground truth is compare with the predicted saliency map. In particular two different metrics were used: similarity (SIM) that compute the similarity between two grey-scaled maps viewed as distributions and Pearson’s linear coefficient (CC) that compare two maps using correlation. The second method allow us to work directly with a grey scaled map instead of use an arbitrary threshold as in the first method. Similarity of grey scales maps was higher than a random map, suggesting the prediction of saliency maps from memorability maps is possible but requires some processing to achieve accurate results.

Convolutional neural networks improve other existing methods and allow us to analyse images in different ways with accurate results. In particular, a convolutional neural network model for sentiment analysis and prediction was used to estimate the probability that one image had a positive sentiment assigned. After compute this new value for Insight dataset (a set of fifty images) this score was compared with memorability score. With Spearman’s rank correlation a lower value near zero was obtained. For that reason, there is no relation between memorability scores manually annotated and the affective score predicted by the CNN model.

Last chapter of this project was involved in the human physical state. With the help of physiological sensors user that generate image set was monitored and three different physiological were acquired: heart rate, galvanic skin response and electroencephalographic signals.

As first part of signals exploration the correlation between memorability of images (predicted with a trained convolutional neural network trained for that purpose) and physiological signals have been retrieve. From a new dataset that contains images and their
correspondent physiological signals (heart rate, galvanic skin response and calories) all this data were processed as described in previous sections and the results shows that a pattern between these physiological signals and memorability exists. A general rule is that when memorability of an image increases the value of heart rate tends to decrease, and the same happen with galvanic skin response value. Also results show that median value of galvanic skin response is equal to 0, so that value only change if there is a special event occurs. For that reason this signal should not be used for relate it directly to memorability score. Despite the general pattern shown the value of heart rate value cannot be directly related to memorability score, so a temporal processing in data acquisition must be done to not only consider one value of a physiological signal, a temporal change is more valuable.

In the same chapter, electroencephalographic signals were explored. After setting up and run the designed experiment a single value for each target image was computed and compared with the memorability score.

Electroencephalographic signals, brain response, can be a valuable cue to predict image memorability. In this case visual memory web game was extended to an EEG experiment and a complex set up was to be done. After signal capture a processing was needed, computing just Pz signal between -200 and 1000 milliseconds and obtaining a response value as the average of the interval time between 350 and 600 milliseconds, the interval corresponding to an event brain response.

With the response value for all targets images three correlations were explored: correlation between response difference (first and second repetition) and memorability scores, correlation between target first visualization response and memorability score, and correlation between target second visualization response and memorability score. All these correlations computed as Spearman’s rank correlation show a low value, near zero in two cases meaning that prediction is comparable to random assignment.

This method is a good approach for study the correlation between EEG signals and the memorability score but more users are needed to establish a correlation and a user that has seen the images before can affect in a negative way to the experiment.

Once physiological signals was explored and with the pattern founded in the heart rate value when memorability score changed, last sections encouraged to train two types of linear regression: the first one to predict memorability scores just with heart rate values with the aim to determine the moment of image capture. The second regression model trained tries to predict memorability score from heart rate value and the output of EgoMemNet
CNN, combining human and visual features. This last regression model presents the same performance as only using the value of the convolutional neural network model. For that reason the use of heart rate value is not useful for memorability prediction due to the variation of this value with respect to physical or emotional user status.

Last section of this project explores an important physiological signal for memorability study: electroencephalographic signals. This is the brain response to a certain stimulus. In this case the response to each target image in its first and second visualization was recorded and compared using the difference. With these three values (first, second and difference response) a correlation between them and memorability scores (manually annotated) was search. Results show any of these responses are correlated with memorability scores directly. The assumption of this non-correlation is that user had seen images before or many users are needed to compute the average of the responses.
References


[10] Ceren Ergorul and Howard Eichenbaum, “The Hippocampus and Memory for “What,” “Where,” and “When””. In Center for Memory and Brain, Program in Neuroscience, Boston University, Boston, Massachusetts 02215, USA

Glossary

**CNN** – convolutional neural network.

**Fc6** – fully connected layer of a convolutional neural network, sixth layer.

**MSE** – mean square error.

**SAD** – sum of absolute difference.

**Fine-tune** – from a pre-trained model of a convolutional neural network update weights from learning new examples.

**EEG** – electroencephalographic

**SIM** – similarity metric

**CC** – Pearson’s linear coefficient
Annex I

**JavaScript algorithm details:**

At the begging of the algorithm there are two text files that are read and stored in two different variables. This text files contain the name of images, one file for targets and other for fillers. Apart from charge this names to a variable, these news variables have more fields. This variable consists in a table that at the end of the game will be stored in a local storage.

The structure of this table is the following: a first field indicates the name of the image including the format, the others fields are numerical and allow us to have a control of these images, in particular, target images. These numerical fields are:

- Second field: is binary value, 0 means that this image have not been shown during the game, that user have not seen; and 1 means that this image have appeared at least one time, but not ensure the image have been repeated.
- Third field: is a positive integer number that indicates the position in where this image has appeared. This number is the position in the whole sequence including fillers images.
- Forth field: is a positive integer number that indicates the position in where this image has been repeated. It means that the image appeared before and that is the first repetition. This value helps us to not to show this image another time, because we have to ensure targets appeared only two times (this is not mandatory for fillers images).
- Fifth field: is a binary value that indicates if the user has detected the repetition of this image. A JavaScript function takes care of that fact.

This table has also a final line that is different from the rest. This line consists in five more values referred to vigilance fillers:

- First value: number of vigilance fillers that have been shown during the game, to control user attention.
- Second value: number of vigilance fillers that have been detected for the user.
- Third value: number of vigilance fillers that have not been detected for the user.
– Forth value: relation between detected vigilance fillers against non-detected vigilance fillers.

– Fifth value: binary value that define if we can use these results for this research (1) or we have to discard them (0).

The criteria for evaluate user attention is that the relation between vigilance fillers detected and non-detected must be equal or high than 0.5. This relation is computed as:

\[
AC \ (Attention \ control) = \frac{correct \ vigilance \ fillers \ detections}{vigilance \ fillers \ non-detected}
\]

From this value, utility value is defined as:

\[
Utility = 0, \ if \ AC < 0.5
\]

\[
Utility = 1, \ if \ AC \geq 0.5
\]

Once this variables have been created a $interval$ function, from JavaScript library called $Jquery$ start a cycling repetition. This repetition is every 1.2 second and can be stopped by two ways: there is a “stop” button and after press it, this repetition stops immediately; the other option is the normal use of the game, after 9-10 minutes the application stop automatically. This is equivalent to 440 cycles of the algorithm. This does not means that 440 images appear during the game because images are separated by blank image. This image consists in a white images with a little black square in the middle with the purpose of focus user gaze in this part of image and makes that user pay more attention in where the images are shown.

For the game, images have been resized to 300x300 pixels images with Matlab script. With this dimensions we ensure user can have a global view of image and also that objects in the image not appear too big. The blank image has the same dimensions and the black square is 20x20 pixels, allowing to user to focus the gaze at this point.

The first step of the algorithm is to compute a random number between 0 and 3, the number of fillers randomly selected before target visualization. After this visualization a target image are shown and the control table are update, changing the values of the second and third fields, indicating the image has appeared and in what position. After that, the
algorithm computes a new random number between 0 and 3 for fillers visualization. This proceed is repeated 220 times and between each image visualization a blank image are shown. But the algorithm is more complex because the first fillers of the N fillers array created in every iteration of the algorithm is the vigilance filler. This image have to be repeated in a show interval of images because is for user attention control, so that image will be shown in the next iteration of the fillers, as the first filler. Also target selection is not trivial, after randomly select a target, the algorithms look if the image correspondent to that list position has been shown in the game. If the image has not been shown, it has to be shown. If the algorithm looks that this image has been shown and since the first repetition number of images seen are between 8 and 40 it show this image. If the number of frames shown is not in this interval the algorithm computes a new random value and do the same inspection of the image. There are 10 trials for select a target for not to collapse the algorithm. For the new random value could happen that it have not appeared before. But a critical situation may occur: a target image has been shown 40 images ago and the algorithm has not selected it yet. So, a condition in the algorithm avoid that possibility and when the algorithm has to select a new target, it first explores all positions in the list (all targets) for retrieve if any target presents this problem.

After these 440 iterations of the algorithm (including iterations for blank images) the algorithm stops automatically and stores locally the results in a text file. This result is the target table.

The most important part of this game is the user, who annotates image memorability score. The task for the users is to press button “start” to run the algorithm in the web application and if they think that an image has appeared for a second time they have to press key “d”. This web application has been designed to be used in Google Chrome browser and that the duration of the game is between 9 and 10 minutes. In comparison with MIT game that takes 5 minutes to complete, this game allows us to annotate a high number of images in each hit (each time that the game is done). It is important that users have not seen these images before because it can cause some confusion.
Annex II

Compute Spearman’s rank correlation:
To compute Spearman’s rank correlation Matlab tools were used:

- To compute the rank position of each sample:

  \[ [R, T] = \text{tiedrank}(\text{rank}_\text{list}); \]

- To compute the Spearman’s correlation:

  \[
  \text{diff}=0; \\
  \text{for } jj=1:\text{length}(R) \\
  \quad \text{diff}=\text{diff}+(R(jj)-Ro(jj))^2; \\
  \text{end} \\
  \text{spear}=1-((6*\text{diff})/(\text{pairs}*(\text{pairs}^2-1))); 
  \]


Visual Memorability for Egocentric Cameras - Marc Carné Herrera

Annex III

First Person Vision workshop CVPR 2016 poster:
Annex IV

Docker implementation details:

The installation of docker environment starts in his website, downloading two different software:

- Kitematik: is a software that allows us to download docker images that other users have upload to dockerhub. We can use these docker images for free and give us some examples. It is useful for example to see some Dockerfile examples.
- Docker Quick Start Terminal: application that opens a system terminal and start docker virtual machine in order to use docker commands.

After installation, the first step is to create a Dockerfile that contains the structure of the docker and his functions. In the next figure we can see an example of Dockerfile for this work.

![Dockerfile example](image)

Commands used to create the image of the docker are:

- FROM: indicates the operative system that virtual image has to be.
- RUN: in the bash of the virtual machine, this command starts the process that follows. In a Dockerfile can be many of these issues. It is to set up the virtual machine as install packages, change directories, etc.
- WORKDIR: docker command for change the work directory, similar to ‘cd’ command from Linux.
- **EXPOSE**: when a server function is required for *docker image*, we have to specify the port where the process that we want to execute as server will use in the virtual machine.

- **CMD**: command to execute the main action of the docker virtual machine. Only one of this command can be used in the Dockerfile. Comma split must be used to indicate to the operative system what to do.

- **ADD**: command to add some files from physical machine to virtual machine.

As in others programming languages a comment of a line is done adding ‘#’ at the beginning of the line.

Once the configuration file has been created a virtual image has to be built. To do that, after execute *Docker Quick Start Terminal* and change the directory to the place where the Dockerfile is located, next command in the terminal must be used:

```
$ docker --t built image_name
```

After built, this image is available for his use. In dockerhub we can upload this image to allow other users to use it. An easy idea was to install *docker tools* in UPC server and download the image from the hub, but this way to deal with the problem is inefficient. The best way we considered is to use only Dockerfile for the implementation, as it is a 500bytes file.

With only a simple text file we could create a full virtual machine. In this file, as we saw previously, different steps for the set up was specified. The efficiency of this implementation is based on github\textsuperscript{10} use. This webpage is where programmers can store, share and structure their code, so with a simple command we can clone a whole repository of code in a machine, as I did in this work. The code for the game is publicly available\textsuperscript{11}.

In the server computing services built the Dockerfile and run in a physical port, linking virtual machine port with it. To run a virtual docker image next command has to be used:

---

\textsuperscript{10} Github webpage - https://github.com

\textsuperscript{11} Visual game code webpage – https://github.com/imatge-upc/lifelogging/tree/master/memorygame
$ docker run – it image_name

If we need to use the terminal of the virtual machine to supervise any process or execute something in the container, the command is the follow:

$ docker run – it image_name bash

As a result of these steps the visual memory game for this work was available online\(^\text{12}\) and open to everyone that wants to collaborate with this research.

After test the visual memory game with some users not all results were valid because the last line of the text file obtained as results of the game was used for control user attention. Despite website had instructions for users, some of them might not understand the task well. Also since this visual game was run in a server some problems can occur during his use, for that reason a supervision of the results is required.

\(^{12}\) Online visual memory game - http://imatge.upc.edu:8000
FPV CVPR 2016 spotlight video in Youtube:
Find an interesting video about the spotlight in the egocentric (first person) vision at Computer Vision and Pattern Recognition 2016.

EgoMemNet: Visual Memorability Adaptation to Egocentric Images
https://youtu.be/qwM5NNW37YE