UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
BACHELOR OF COMPUTER SCIENCE

GUILHERME SCHIEVELBEIN

Similarity Collect - Clustering of Saccades

Work presented in partial fulfillment
of the requirements for the degree of
Bachelor in Computer Science

Advisor: Dr. Luciana Nedel
Coadvisor: MSc. Thomas Kübler

Porto Alegre
July 2015

CIP – CATALOGING-IN-PUBLICATION

Schievelbein, Guilherme

Similarity Collect - Clustering of Saccades / Guilherme Schievelbein. – Porto Alegre: CIC da UFRGS, 2015.

51 f.: il.

Trabalho de conclusão (graduação) – Universidade Federal do Rio Grande do Sul. Bachelor of Computer Science, Porto Alegre, BR–RS, 2015. Advisor: Luciana Nedel; Coadvisor: Thomas Kübler.

I. Nedel, Luciana. II. Kübler, Thomas. III. Título.

"In its majestic equality, the law forbids rich and poor alike to sleep under bridges, beg in the streets and steal loaves of bread."

— Anatole France

ACKNOWLEDGEMENTS

My dear friends for the good times together.

Dr. Mara Abel for advising me during most of my bachelor studies.

My colleages at UFRGS and University of Tübingen.

Dr. Luciana Nedel and Msc. Thomas Kübler for advising me in this work.

Petrobras and Tübingen University's Department of Computer Science for the scholarship during my studies.

Department of Art History of University of Vienna for providing the testing data.

CONTENTS

LIST OF ABBREVIATIONS AND ACRONYMS

EOG	Electro-oculography
VOG	Video-oculography
UI	User interface

LIST OF SYMBOLS

$\overline{AB}$	Line segment between points A and B.
$\angle AOC$	Angle between points A and C at point O.

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Eye tracking has become widely popular in number of research fields such as human-computer interfaces, marketing, art history, and cognition over the last years. This increase in popularity is due in part to recent advancements in eye tracking technology which have increased the availability of cheaper, faster, and more precise eye tracking devices.

While studying eye movements has the potential to reveal and help understand human cognitive processes, the raw data obtained with these devices does not provide much meaningful information on first sight. Several methods were developed to simplify these data and analyse it quantitatively and qualitatively. In this work a clustering method for the collection and summarization of similar saccadic eye movements was designed. The method is applied to several datasets of a static image viewing task. The results and their relation to artistic techniques such as image composition and composition lines as well as further domains of application are discussed.

Keywords: Eye tracking, cluster analysis, visual perception.

Coleta de Similaridades - Agrupamento de Movimentos Sacádicos

RESUMO

Rastreamento ocular tornou-se amplamente popular em vários domínios de pesquisa como interfaces homem-computador, marketing, história da arte, e cognição ao longo dos últimos anos. Esse aumento em popularidade se deve em parte devido a recentes avanços tecnológicos que aumentaram a disponibilidade de dispositivos de rastreamento ocular mais baratos, rápidos e precisos.

Embora o estudo dos movimentos oculares tem o potencial de revelar e ajudar a entender os processos cognitivos humanos, o dados não processados obtitos com esses dispositivos não provêm muita informação significativa à primeira vista. Vários métodos foram desenvolvidos para simplificar esses dados e analisá-los quantitativamente e qualitativamente. Nesse trabalho, um método de clusterização para a coleta e sumarização de movimentos oculares sacádicos foi projetado. Esse método é aplicado em diversos conjuntos de dados de uma tarefa de visualização de imagens estáticas. Os resultados e suas relações com técnicas artísticas como composição de imagens e linhas de composição, assim como outros domínios de aplicações, são discutidos.

Palavras-chave: rastreamento ocular, análise de clusters, percepção visual.

1 INTRODUCTION

The scientific interest in the study of eye movements is over a century old, but due to technological limitations, these studies were limited to simple recordings of the path on which the eyes moves when performing a determined task. With the advancement of eye tracking technology we are now able to reliably track not only the eye gaze, but information such as the pupil dilatation. Furthermore, advancements in computing capacity makes it possible to easily manipulate, analyse and compare big amounts of eye tracking data.

Eye tracking has been widely used in reading (HENDERSON; FERREIRA (1993)), marketing (PIETERS; WEDEL (2004)), human computer interaction (JACOB; KARN (2003)), and driving(KASNECI et al. (2014)) studies, among others. A survey of many eye tracking applications is available in DUCHOWSKI (2002).

1.1 Historical context

The first systematic studies of eye movements were made by Buswell (BUSWELL (1935)) by recording beams of light reflected on the subject's eyes when looking at pictures. This work was later expanded (YARBUS; HAIGH; RIGSS (1967)) by recording the eye movements of the subjects during different tasks, as shown in figure 1.1. This study demonstrated that the subject's eyes tended to fixate on specific areas of the pictures. Further research by NOTON; STARK (1971) demonstrated that these fixations, or areas of interest, are independent of the tasks. The pattern taken by the eye when visualizing the pictures was named scanpath.

The identification of the fixations and the scanpath forms the basis of eye tracking data analysis. According to JUST; CARPENTER (1976), the identification of fixations provides the locations of the meaningful content in the pictures. The first eye tracking studies were limited to mere observations of the subjects eye movements, and the first eye tracking devices were intrusive and imprecise. That has changed in the recent years with the development of new eye tracking technologies that enables us to analyse a bigger amount of data and provides more information about the subject's eyes, fixations, and eye movements during these recordings.

1.2 Motivation and objective

According to DUCHOWSKI (2007), the goal of eye tracking studies is to gain a better understanding of the human visual and attention behaviour, and underlying cognitive processes. While many studies examined the importance of fixational patterns, the thesis on

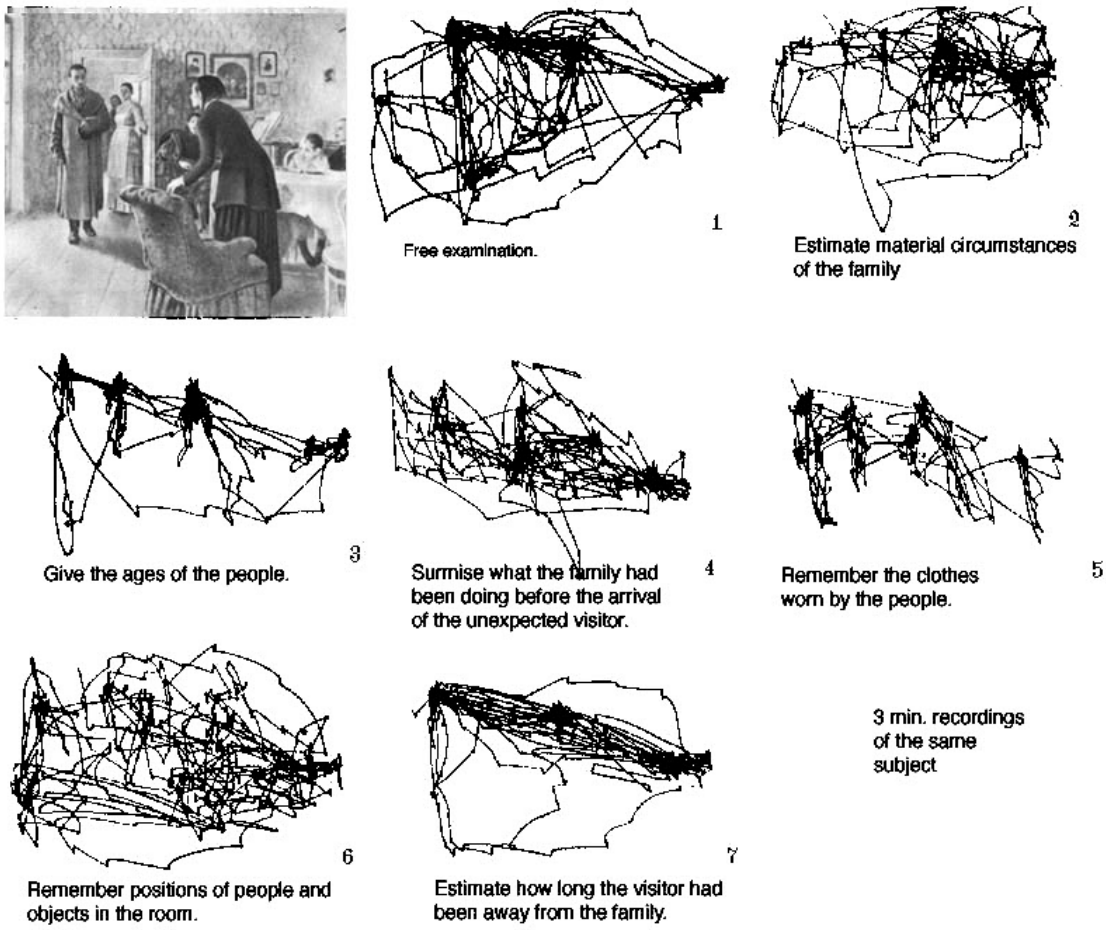


Figure 1.1: Eye movement recordings by Yarbus showing the various gaze paths the observer utilized to solve different tasks. Source: YARBUS; HAIGH; RIGSS (1967)

hand focuses on saccadic patterns. By analysing not just spatial fixation information, but gaze transitions, we may be able to improve our understanding on how different kinds of users observe pictures and how gaze can be guided through a painting by compositional and artistic tools.

A recording of eye movements of a minute and a half may contain up to a thousand saccadic movements. This makes visual clutter a problem in the currently available visualization techniques. Clustering methods were developed (SANTELLA; DECARLO (2004)) that tackle this problem with eye fixations, but no method is available for the clustering of saccades. We designed a clustering algorithm capable of grouping similar eye movements of one or more subjects.

1.3 Monograph organization

Chapter 2 covers the basis of eye movement research and provides an introduction to the most relevant types of eye movements for this work. Also a short introduction to eye tracking technology is provided.

In chapter 3 the Eyetrace software bundle developed at the University of Tübingen and the functionality relevant for this thesis is introduced, namely, properly labelling the different kinds of eye movements given the raw data obtained via eye tracking.

Chapter 4 explains what is hierarchical clustering and how it was implemented in Eyetrace2014 for the grouping of similar eye movements. The results obtained with the algorithm are discussed in chapter 5.

The clustering algorithm and its relation with related work is discussed in chapter 6.

Finally, in chapter 7 further applications and research areas are proposed.

2 BASIC CONCEPTS

In this chapter we will define the types of eye movements that are important for eye tracking research and we will explain the various kinds of eye tracking devices and how this technology has evolved to its current state.

2.1 Eye movements

According to DUCHOWSKI (2007), three types of eye movements are important to be modelled for eye tracking study:

- Fixations
- Saccades
- Smooth pursuits

Fixations occurs when the eye gaze is relatively stationary and information of the gaze target is processed. Saccades are movements of the eyes between the various fixations. A smooth pursuit occurs when tracking a moving object. When working with static pictures, only fixations and saccades need to be modelled.

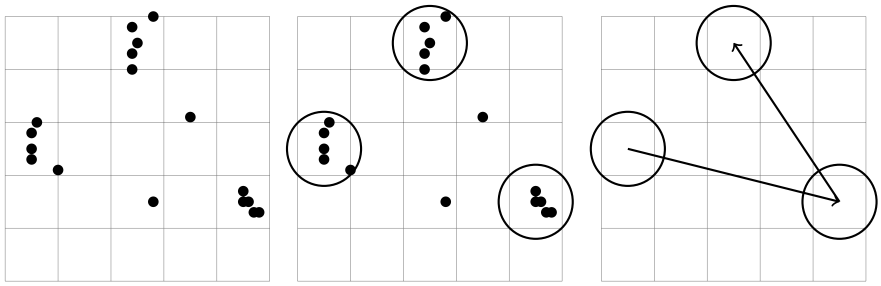
2.1.1 Fixations

The eye never stays perfectly still on a fixation point, but performs tiny movements, such as microsaccades (HUBEL (1988)). This occurs because of the structure of the eye, which is formed by motion-sensitive cells, which causes an image to disappear when it stays stabilized on the retina (DUCHOWSKI (2007)). These small movements are usually irrelevant for the eye movement analysis.

An eye tracker usually provides information about the looked-at point at each time (figure 2.1a). There are different methods of identifying fixations, based on the distance between subsequent measurement points (figure 2.1b) or on the gaze velocity. They are discussed in-depth in chapter 3.

According to SALVUCCI; GOLDBERG (2000), a fixation usually lasts around $300ms$. RAYNER (1998) demonstrated that the fixation's duration depends on the subject's tasks.

It is important to correctly identify the fixation points because it is during fixations that the brain processes the visual information (JUST; CARPENTER (1984)). Longer fixations times are usually assumed to be associated with more complex cognitive processes.



(a) Gaze points (b) Fixations obtained based on the gaze points (c) Saccades between the fixations

Figure 2.1: Example of an eye tracking recording gaze points, fixations and saccades

2.1.2 Saccades

Saccades are the rapid movement of the eyes that occur between fixations. Saccades can cover a wide spatial range, from very small attention shifts to nearly targets, but also to gaze targets previously in the peripheral region. A saccade lasts between $10ms$ and $100ms$ (SHEBILSKE; FISHER (1983)), depending on the length of the saccade. The saccades are usually visualized as arrows from the starting fixations to the destination fixations, as in figure 2.1c.

Little visual processing is possible during a saccade (FUCHS (1971)) because of its high velocity and short duration. That is why it is a necessary step of eye tracking data analysis to correctly label each gaze point as belonging to a fixation or a saccade.

2.2 Eye tracking devices

There are many kinds of eye tracking methodologies and devices, each with their own advantages and disadvantages. The methodology used for an experiment depends on what kind of information is necessary and how precise it must be.

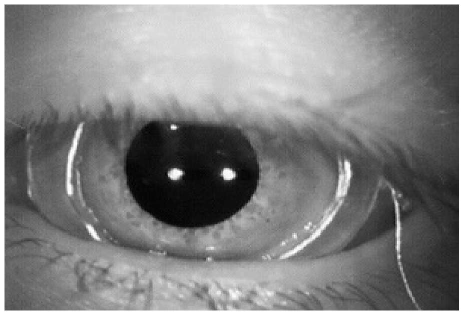
Earlier eye tracking measurements were made using a technique known as electro-oculography (EOG). EOG uses electrodes placed around the subject's eyes to measure the skin's electric potential differences. Figure 2.2a shows a subject with an EOG electrodes placed on his face.

Another technique for eye movements measurements involves the use of contact lenses worn by the subjects, as shown in figure 2.2b. It measures the position of the eye in relation to the head, so it is necessary to stabilize the head to obtain precise measurements in relation to a point of regard. Although it provides very accurate measurements, it is very intrusive and uncomfortable for the subjects.

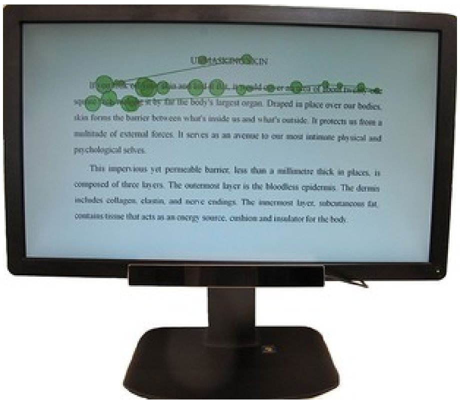
Contemporary eye tracking devices utilize video cameras and image processing techniques to analyse certain features of the eyes such as the pupil centre and Purkinje reflections (figure 2.3). This technique is known as video-oculography (VOG). These devices can either be fixed, as the one in figure 2.2c, or head-mounted. After proper calibration, these devices can accurately provide the subject's point of regard on a planar surface. By using a camera pointing to the subjects eyes and another pointing to the subject's facing direction (figure 2.2d), this methodology can be used for tracking the eye movements during dynamic situations. VOG is highly accurate and, unlike the previous techniques, is easy to set up and non-invasive, which makes it the currently most used methodology for eye tracking.

A video-based tracker was used in this work, however the methodology could also be applied to data recorded with any of the other eye tracking methods.

(a) Subject wearing an EOG device



(b) Subject wearing a coil lens. Source: IMAI et al. (2005)



(c) Table mounted eye tracker by SMI

(d) Head mounted device by Ergoneers

Figure 2.2: Eye tracking devices

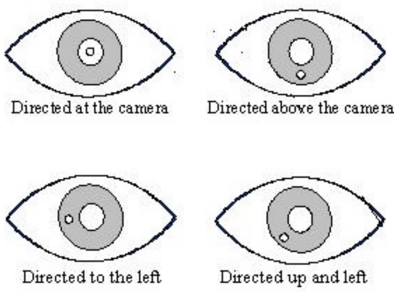


Figure 2.3: Picture of a Purkinje reflections formed on the eye according to the gaze direction in relation to the camera. Source: REDLINE; LANKFORD (2001)

3 EYETRACE

The Eyetrace software bundle (KUEBLER et al. (in press)) is the result of a cooperation between the Department of Computer Science at the University of Tübingen and the Department of Art History at the University of Vienna. It consists of two pieces of software, EyetraceButler and Eyetrace2014, and can be freely downloaded for research and educational purposes at `http://www.ti.uni-tuebingen.de/Eyetrace.1751.0.html`.

The main part of eye tracker's recordings analysis is the labelling of fixations points and saccades, and the analysis of statistics regarding these fixations (SALVUCCI; GOLDBERG (2000)). The proper identification of fixation points relies a lot on the type of task being performed and the proper choice of algorithms and their parameters.

Commercial eye tracking devices usually come with their own analysis software, each implementing their own algorithms. The implementation of these algorithms is usually not publicly available, which makes it hard to make a conscious decision for the algorithm's parameters. Furthermore, this makes it hard to replicate results when comparing data obtained with devices from different manufacturers. To complicate things even more, when some custom analysis is needed, the recording data must be exported and the calculations implemented in a third party software.

Eyetrace aims to address these problems by offering data import capabilities for multiple devices and well documented algorithms. The users are also able to easily change these algorithms' parameters. This is done by first pre processing the recordings using the EyetraceButler to convert the data to a standardized format. EyetraceButler currently offers conversion and import functionality for many major brand eye tracker devices such as SMI (SENSOMOTORIC INTRUMENTS GMBH (2015a)), Ergoneers (ERGONEERS GMBH (2015)) and TheEyeTribe (THE EYE TRIBE APS (2015a)).

This converted data can then be loaded in Eyetrace2014. It provides a series of algorithms for the identification and clustering of fixations, manual and automatic creation of areas of interest, and data quality analysis. It also offers various customizable visualization methods.

3.1 Gaze Points

The raw data obtained with eye tracking devices usually contains information about the gaze coordinates of one or both eyes in relation to a point of regard at determined instants. That is, the point (x, y) of the eye gaze in an instant t. Table 3.1 provides an example of this data. The rate at which these points are sampled is dependent on the sampling frequency rate of the tracking device, varying from $30Hz$ (THE EYE TRIBE APS (2015b)) in cheaper devices to $1,250Hz$ (SENSOMOTORIC INTRUMENTS GMBH

(2015b)) in some high end devices.

Timestamp (ms)	Left eye (x, y) coordinates	Right eye (x, y) coordinates
340	(1470,906)	(1452,900)
374	(1447,933)	(1459,899)
407	(1438,911)	(1471,891)
440	(1468,877)	(1453,913)
474	(1451,867)	(1450,883)

Table 3.1: Sample of an eye tracker recording

These gaze points can be visualized in Eyetrace2014 (figure 3.1). As can be observed in that figure, with this visualization it is already possible to identify some regions of interest where the concentration of gaze points is high. It becomes easier to interpret this data by generating a heat map, or attention map, of the gaze points, as the one in figure 3.2. This heat map shows which regions of the picture attracts the users interest.

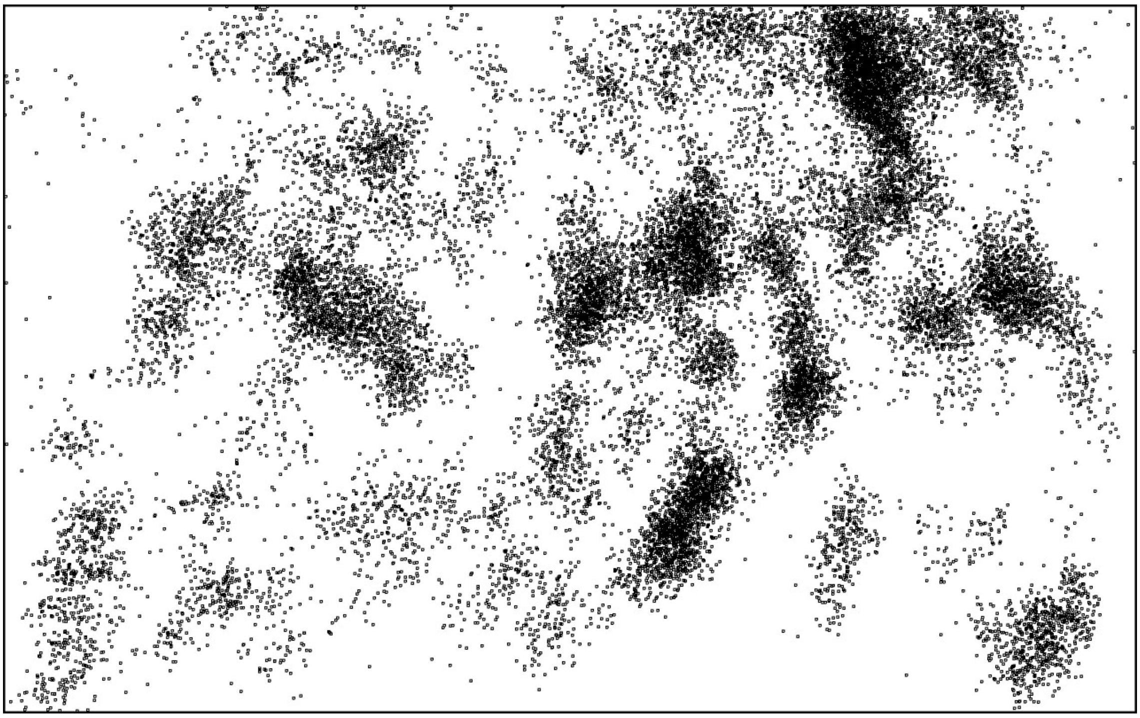


Figure 3.1: Gaze points visualization of the recording of a single subject for 1min 30s

3.2 Fixations

Although it is already possible to obtain some information about the user's cognitive patterns with the gaze points data and visualizations, it is necessary to reduce complexity and size of the data. This is done by identifying the fixations and saccades (SALVUCCI; GOLDBERG (2000)).

According to HOLMQVIST et al. (2011), fixations can be understood as an aggregation of gaze points on a specified area and timespan. Some of the regions from figure 3.1 that could be seen as a single fixation may actually contain more than one fixation, as they may not be temporally subsequent (BLIGNAUT (2009)).

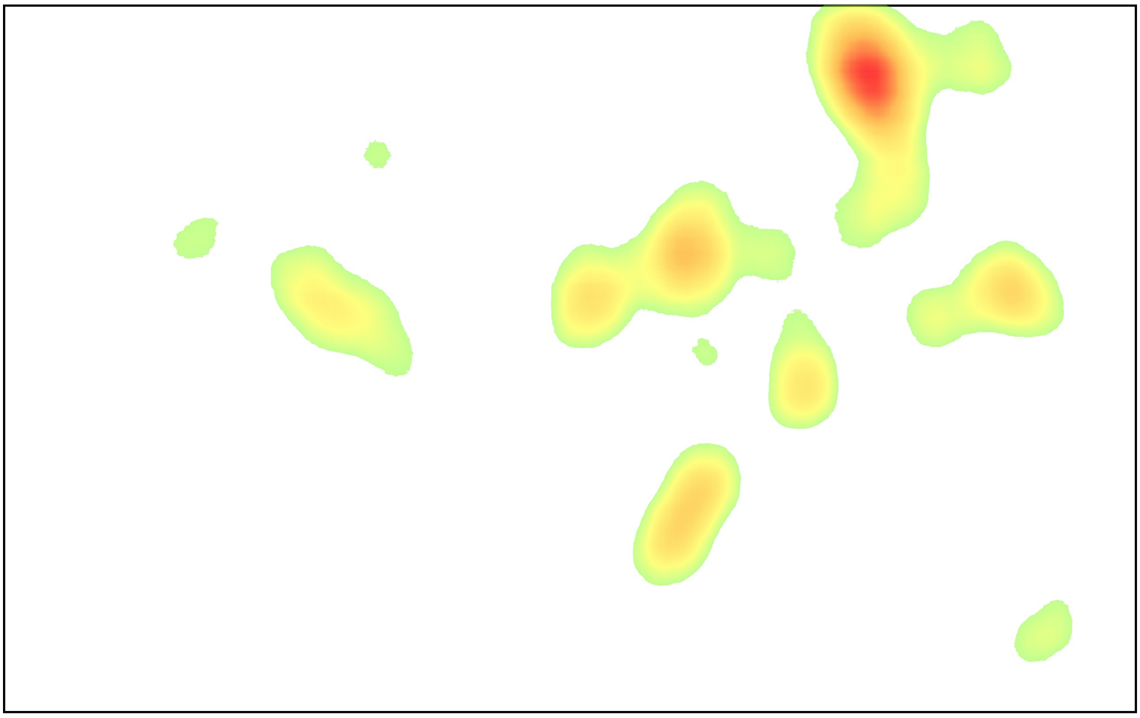


Figure 3.2: Heat map generated based on the gazepoints from figure 3.1

The algorithm chosen to identify the saccades and its parameters can change drastically the results obtained. This choice must be made by the researches according to the type of data and its subjects. Eyetrace2014 implements three kinds of methodologies for fixation identification:

- Dispersion based: This method, first proposed by WIDDEL (1984) takes advantage of the fact that gaze points of fixations are located near each other, since the eye is moving at low velocities during a fixation. The identification is made by setting thresholds for the dispersion of the fixation's gaze points and for its duration. These parameters are interdependent and must be chosen carefully by the users (SALVUCCI; GOLDBERG (2000)).
- Gaussian mixture model: This method, proposed by TAFAJ et al. (2012), takes advantage of the assumption that the distance between the gaze points of a fixation forms a Gaussian curve to derive the fixation points.
- Velocity based: As the eye movement speed is low during a fixation, and fast during a saccade, a velocity threshold can be set. By comparing the timestamp between two subsequent gaze points, the velocity of the this eye movement can be determined. If it is below the threshold, the point is considered as belonging to a fixation, otherwise it belongs to a saccade.

Fixation are usually visualized as circles in the picture, as shown in figure 3.3. This visualization reduces the visual clutter that happens when visualizing all the raw gaze points. The size of these circles may also be changed to be dependent on the fixation duration. Points that do not belong to any fixations are considered as belonging to a saccade.

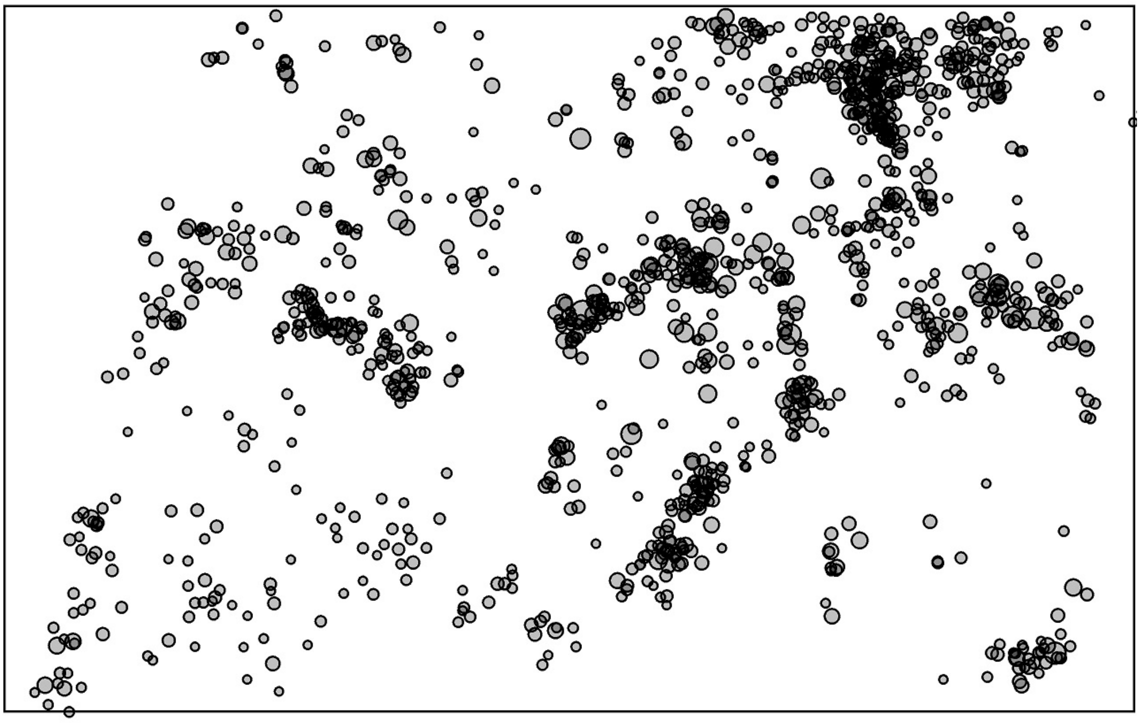


Figure 3.3: Fixations found by applying a dispersion based algorithm on the dataset from figure 3.1, with bigger circles for fixations of longer duration

3.3 Saccades

A saccade represents the rapid movement of the eye between fixations. During this motion, visual information is supressed (HOLMQVIST et al. (2011)). That is why it is important to correctly differentiate the gaze points belonging to fixations and belonging to saccades. After identifying the fixations, we can visualize saccades as arrows going from one fixation to another, according to the order which they occur, as shown on figure 3.4. Because of this fact, saccades were often discarded from eye tracking data analysis and thought as not containing any useful information about cognitive processes.

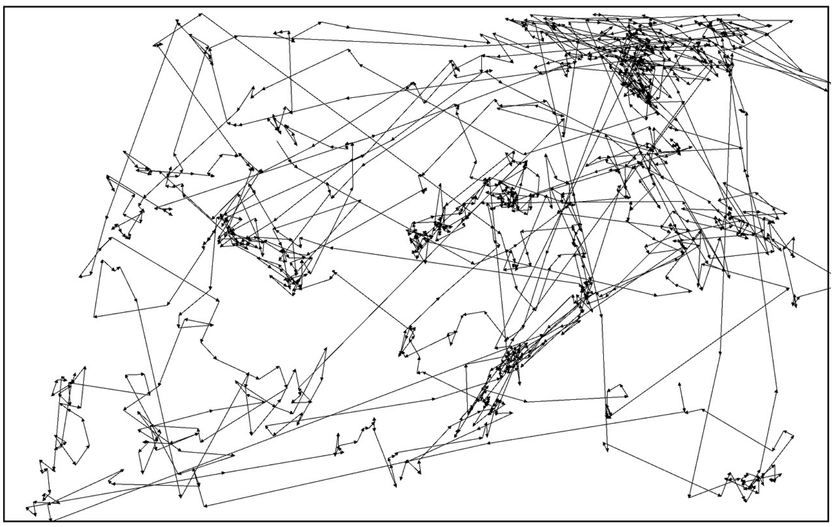


Figure 3.4: Saccades shown as arrows, showing the scanpath of the subject for the fixations from figure 3.3

However, saccades contain the essential information of the sequence in which gaze is distributed over an image. Studying saccadic patterns may allow us to gain insights in which paths our eyes travel when viewing a certain scene and may reveal which factors influence the choice of path.

When viewing an artwork, the artists intention in the composition of the elements of the painting can be expected to influence the viewer. The arrangement and distribution of objects over the image may either induce a certain gaze path through the image for all observers or result in a more unordered viewing behaviour where each observer chooses his own way through the image.

It is possible to make a statistical analysis and visualization of the saccades by creating an anglestar visualization. Three different effect sizes can be quantified when creating an anglestar:

- Quantity of saccades
- Duration of saccades
- Length of saccades

The size of each slice of the anglestar reflects the chosen parameter of the saccades that occurs on the same orientation as the slice. This way it is possible to see, for example, on which direction the saccades happen more often. The number of slices of the anglestar can be changed by the user. Figure 3.5 shows an anglestar generated from the saccades from 3.4 and the quantity of saccades as a parameter.

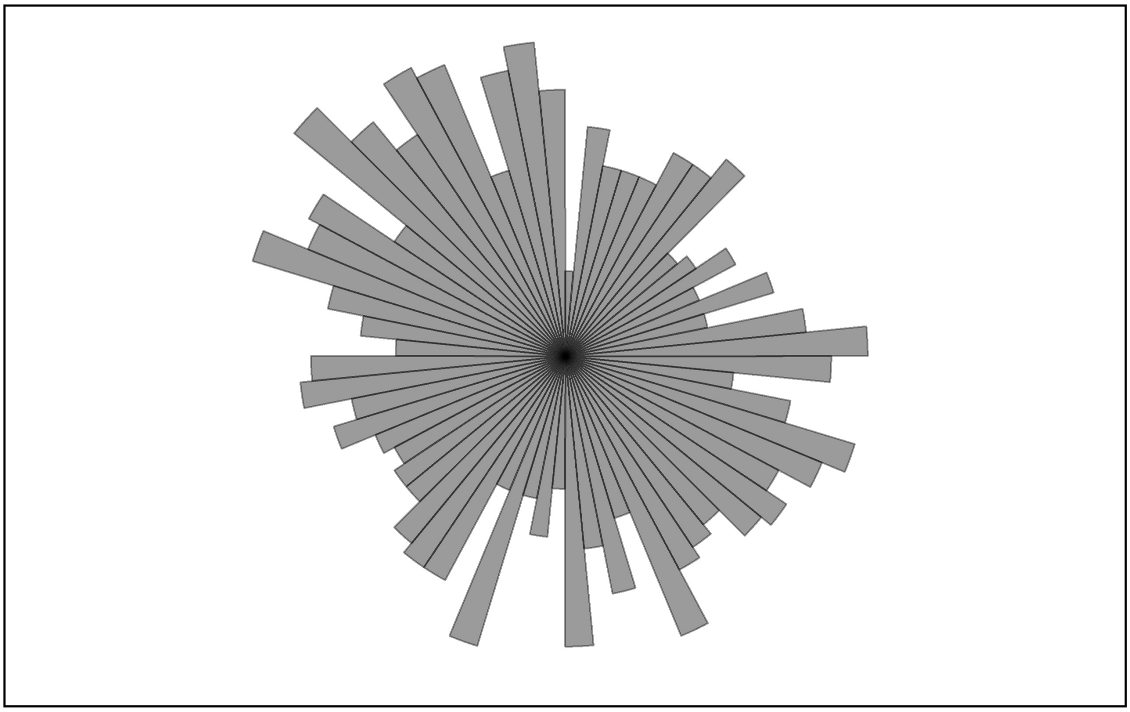


Figure 3.5: Anglestar with 64 slices based on the quantity of saccades from figure 3.4

It is also possible to create a heat map visualization of the saccades, such as the one in figure 3.6. This visualization shows which regions of the picture contains more saccades.

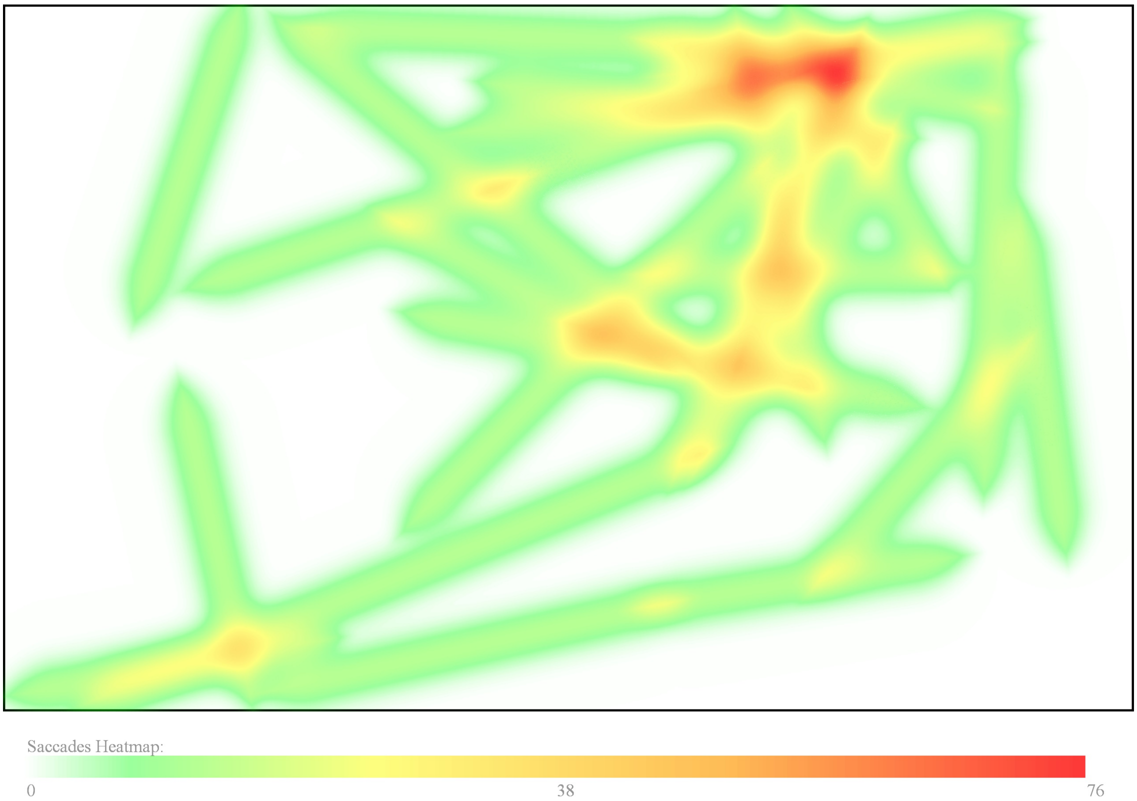


Figure 3.6: A heat map generated with the saccades from figure 3.4

4 CLUSTERING OF SACCADES

A simple arrows visualization of saccades may get too cluttered for a big quantity of saccadic movements, making it harder for the users to make a proper analysis of this information. Figure 4.1 shows one such recording of a single subject. One way to simplify the saccades data is by creating clusters of similar saccades. Although there are methods to group gaze fixations, to our knowledge no similar method is available for saccades. Therefore we propose the grouping of similar saccades with an hierarchical clustering algorithm.

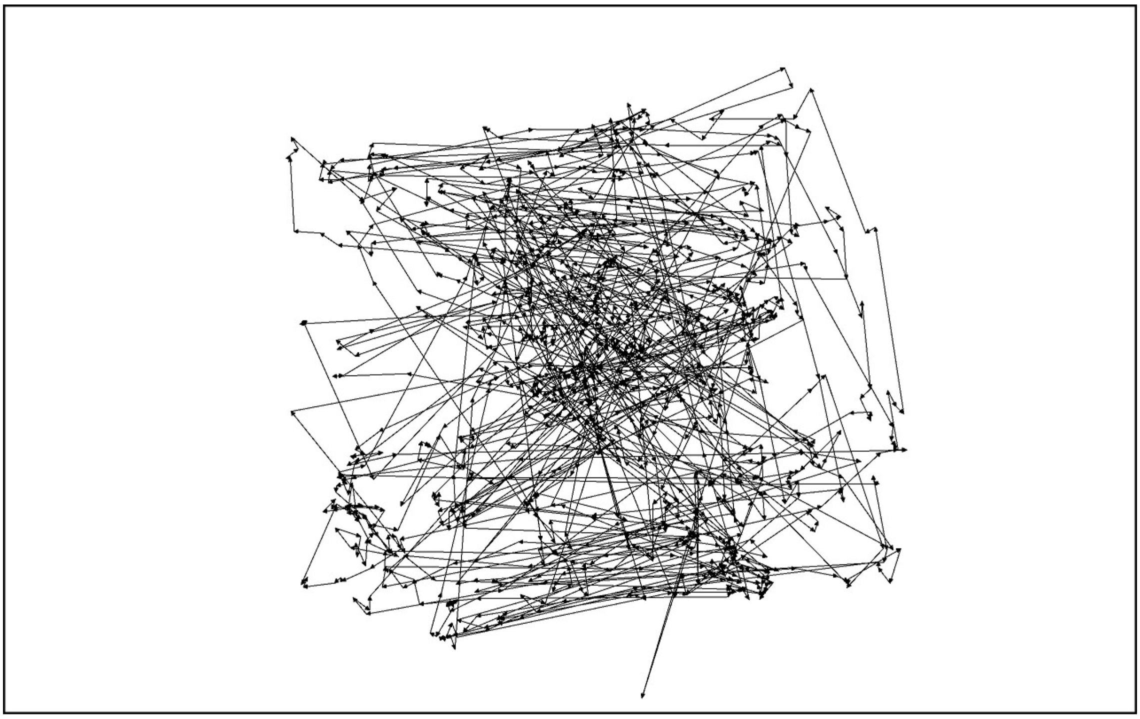


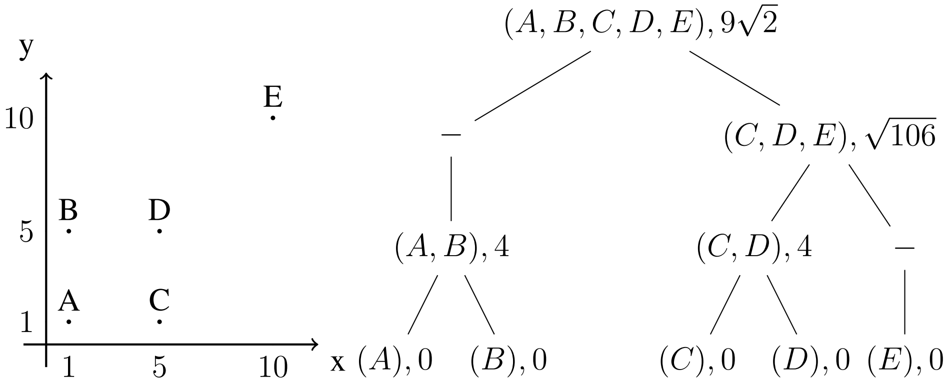
Figure 4.1: Saccades of one subject recorded during 1min 30s viewing time

This method was chosen, rather than some other common clustering algorithm like k-mean clustering, because it is independent of a random initialization and does not need the number of clusters to be known beforehand. The clusters found with the execution of this algorithm correspond to the most frequent paths taken by the eyes during the recording.

It is expected that many saccadic movements occurs between the regions of interest of an image. Furthermore, by grouping similar saccades of many subjects recordings, we may gain a better insight of the cognitive processes that are similar between them when realizing a specific task.

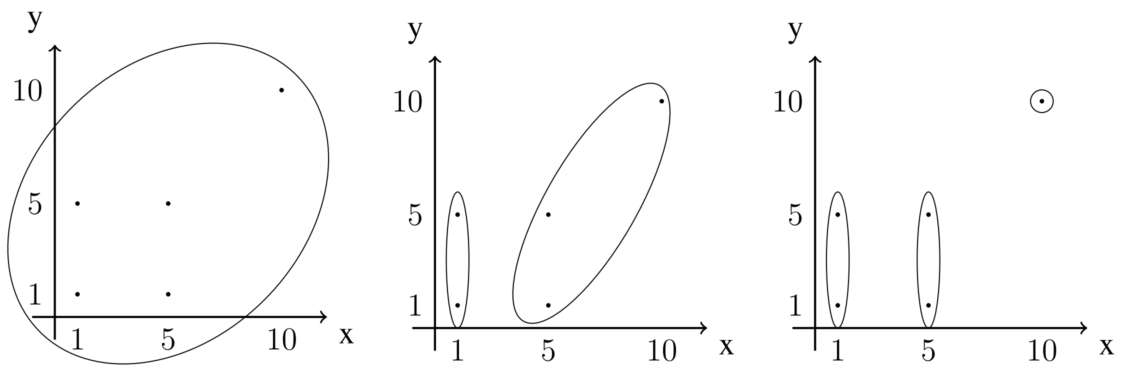
4.1 Hierarchical clustering

Given a set of objects to be grouped with an hierarchical clustering method, these objects may be put in a binary tree structure, where the value of each node represents the level of dissimilarity between the nodes children. This tree forms a hierarchy of the objects. Figure 4.2a shows a possible hierarchy tree for the points plotted on figure **??**, based on their distance to each other. Depending on where this hierarchy tree is cut, we get different segmentations of these clusters, as demonstrated in figure 4.3. At the lowest level of the tree we get clusters containing only one object each, and at top level we get a single cluster containing every single object.



(a) A possible tree hierarchy of the five objects from figure **??**

Figure 4.2: Plot and hierarchical tree of five 2D points



(a) Cluster at the tree's top level

(b) Clusters when cutting tree at the middle level

(c) Clusters when cutting tree at the bottom level

Figure 4.3: Different segmentations

In order to create this tree, we need a method to calculate the dissimilarity, or distance, between each object. For points on a graph, this measurement can be made by calculating the Euclidean distance between the points. This clustering methodology may be applied to any kind of object, as long as there is a way to calculate the distance between them.

In a bottom up approach, given a set of N objects to group (figure 4.5a), each object is assigned to a tree node without a parent and value 0 (bottom level of the trees in figure 4.6). This means that each object is assigned to its own cluster, with a dissimilarity value of 0. We create a $N \times N$ dissimilarity matrix (table 4.1a). Each cell contains the distance between two clusters.

After that, we search the matrix for the smallest value. This value is the distance between the two closest clusters that will be grouped together. On the matrix from table 4.1a, A and B will be grouped together. We link them on the binary tree with a node containing their dissimilarity value (node $(A, B), 1$ in the trees in figure 4.6). The joined clusters must now be removed from the matrix and the new cluster must be added to it. In order to do that, we need a criterion to calculate the distance between two clusters. Two popular criteria are maximum (or complete) linkage (equation 4.1a) and minimum (or single) linkage (equation 4.1b). Basically, maximum linkage will return the biggest distance between the elements of two clusters, and minimum linkage will return the smallest distance, as seen in figure 4.4. Others criteria could also be used to link clusters together. Table 4.1b shows the result of a minimum linkage on the matrix 4.1a an table 4.1c shows the matrix after a maximum linkage.

$$d = \max d(a, b) : a \in A, b \in B \tag{4.1a}$$

$$d = \min d(a, b) : a \in A, b \in B \tag{4.1b}$$

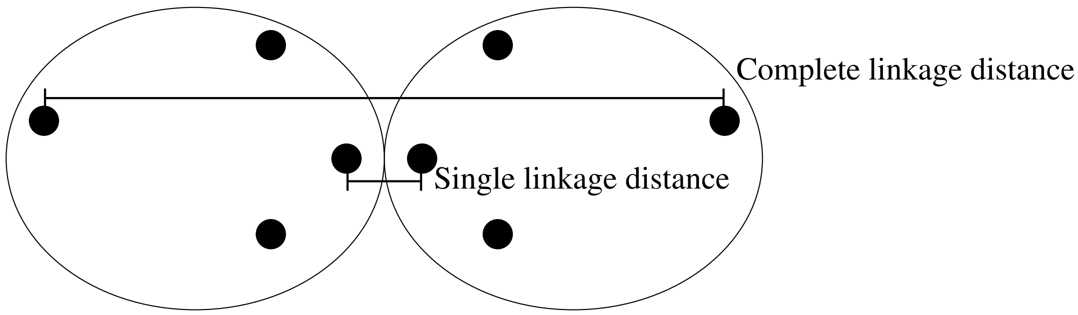
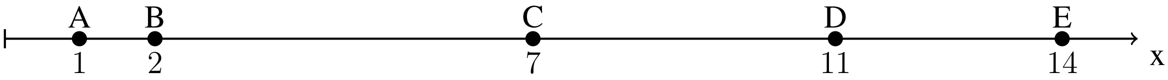


Figure 4.4: Distance between two clusters with both minimum (single) and maximum (complete) linkage

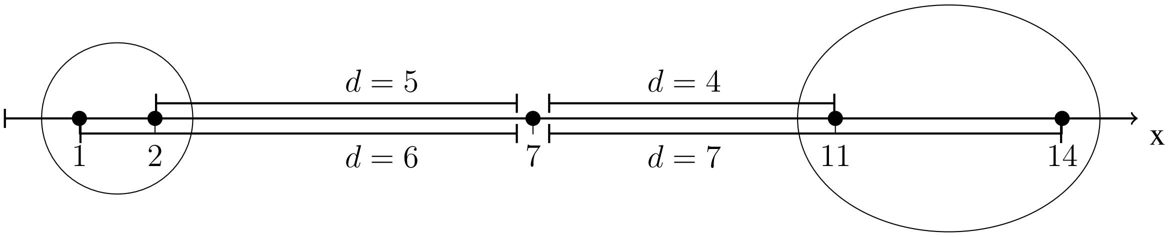
This process of looking for the most similar clusters and linking them together is repeated until we get a top-most tree node containing every single object to be grouped. Table 4.1 shows the grouping process of matrix 4.1a with both maximum and minimum linkage. Figure 4.6 shows the resulting hierarchy tree for both cases. Figure 4.5b shows why the hierarchical trees obtained are different for maximum (figure 4.5c) and minimum linkage (figure 4.5d). After the hierarchy tree is complete, it may be cut at some level to obtain different clusters. The threshold chosen for cutting the tree reflects how dissimilar each cluster element can be. A higher cut off value will provide fewer clusters, with more distant objects in it. The opposite happens with smaller cut off values.

Note that the choice of linkage criterion can change drastically the results obtained. With the data set such as the one from figure 4.7a, due to its homogeneous distribution of objects, every tree node will contain the same value when using a minimum linkage criterion. This means that it is only possible to have each point as a cluster or a big bundle containing every single object, depending on the tree cut off value chosen. By using maximum linkage, we obtain a different tree with evenly distributed clusters, depending again on the cut off value chosen, as show on the figures 4.7b and 4.7c.

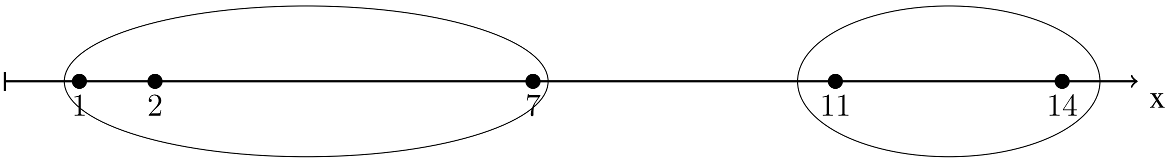
The linkage criterion selection must be made according to the type of data and result that we want to obtain. The cut off value must also be chosen according to how similar we want each cluster objects to be.



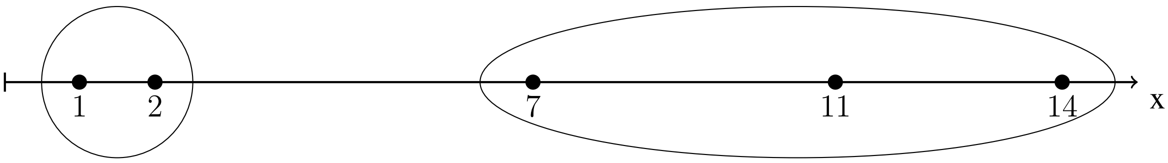
(a) Points to cluster



(b) Distance d of point C to the clusters (A,B) and (D,E) with maximum (bottom) and minimum (top) linkage criteria

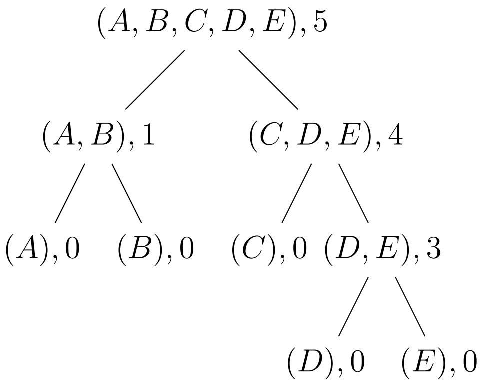


(c) Linkage using maximum linkage criterion

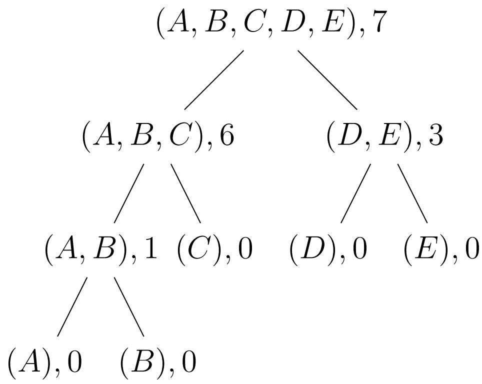


(d) Linkage using minimum linkage criterion

Figure 4.5: Different grouping of five points according to linkage criterion chosen



(a) Hierarchy tree with minimum linkage criterion



(b) Hierarchy tree with maximum linkage criterion

Figure 4.6: Hierarchy trees created with minimum and maximum linkage criteria

	A	B	C	D	E
A	-	1	6	10	13
B	-	-	5	9	8
C	-	-	-	4	7
D	-	-	-	-	3
E	-	-	-	-	-

(a) Starting dissimilarity matrix

	C	D	E	(A,B)
C	-	4	7	5
D	-	-	3	9
E	-	-	-	8
(A,B)	-	-	-	-

(b) Matrix 4.1a after a minimum linkage

	C	D	E	(A,B)
C	-	4	7	6
D	-	-	3	10
E	-	-	-	13
(A,B)	-	-	-	-

(c) Matrix 4.1a after a maximum linkage

	C	(A,B)	(D,E)
C	-	5	4
(A,B)	-	-	8
(D,E)	-	-	-

(d) Matrix 4.1b after a minimum linkage

	C	(A,B)	(D,E)
C	-	6	7
(A,B)	-	-	13
(D,E)	-	-	-

(e) Matrix 4.1c after a maximum linkage

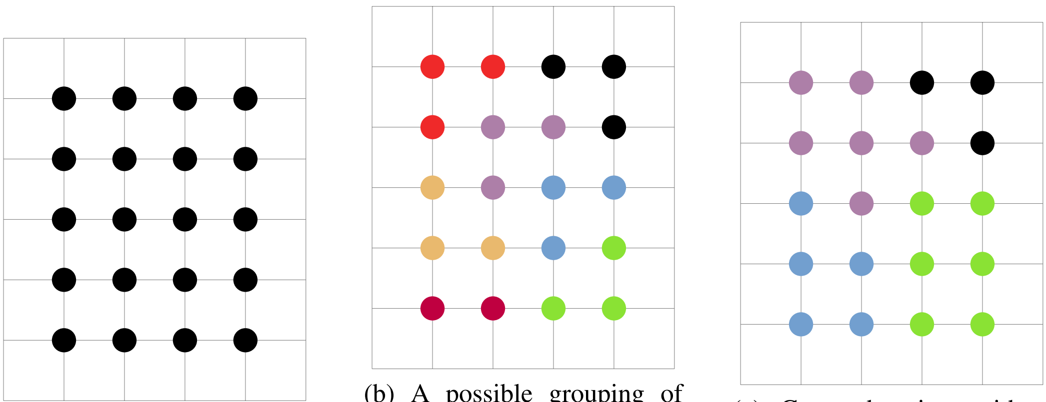
	(A,B)	(C,D,E)
(A,B)	-	5
(C,D,E)	-	-

(f) Matrix 4.1d after a minimum linkage

	(A,B,C)	(E,E)
(A,B,C)	-	7
(D,E)	-	-

(g) Matrix 4.1e after a maximum linkage

Table 4.1: Clustering in action



(a) 2D plot of evenly distributed points

(b) A possible grouping of the points with a strict cutoff value and minimum linkage criterion

(c) Grouped points with a more relaxed cutoff than in figure 4.7b

Figure 4.7: Clustering evenly distributed points

4.2 Implementation of hierarchical clustering for the grouping of similar saccades

In chapter 3, different ways of identifying fixations and saccades in eye tracker data sets were discussed. After this identification process it is obtained for each data set, or recording, a list of saccades ordered by time of occurrence. The hierarchical clustering method demonstrated in section 4.1 was implemented in Eyetrace2014 to bundle together the similar saccades in this list.

Two attributes were established as relevant when determining the similarity between two saccades: their orientation and their distance. That said, the clustering of saccades is made in two steps: first by clustering based on the orientation of the saccades, and afterwards by clustering each of the obtained clusters based on the distance between its saccades. An optional pre processing step may also be executed in order to get a smaller data set. Both clustering by orientation and by distance between saccades can be made with the same hierarchical clustering algorithm. The only difference between both stages is the choice of distance calculation and linkage methods.

4.2.1 Technique overview

The algorithm was implemented in a modular way, so that each stage of the clustering algorithm can be executed separately. This is useful so that after the hierarchical trees are created, the users may experiment with various cut off values, or change the linkage criterion, without having to calculate everything again.

When the user changes a parameter, only the affected stages of the clustering algorithm are recalculated. The stages are:

1. Do the pre processing
2. Create angle distance dissimilarity matrix with the chosen distance method.
3. Calculate hierarchy tree with the chosen linkage criterion.
4. Cut generated hierarchy tree at the chosen cut off value and remove clusters with less saccades than the set value.
5. Create saccade distance dissimilarity matrix for each cluster obtained in the previous stage with the chosen distance method.
6. Calculate hierarchy tree with the chosen linkage criterion for each dissimilarity matrix created in the stage 5.
7. Cut all generated hierarchy trees at the chosen cut off value.

In figure 4.8 the user interface for the saccade clustering in EyeTrace2014 is shown. All the parameters for the clustering method can be freely changed by the users. In this figure, after executing the clustering algorithm, the user changed the cut off value for the orientation hierarchy tree. This means that stages 4 to 7 have to be executed again.

4.2.2 Pre processing

Data obtained from many or long recordings may contain some thousands of saccades, and clustering all this data may be costly and unnecessary. In case we are only interested in analysing the longer saccadic movements, this data may be reduced in two ways: by merging subsequent saccades that deviate less than a given threshold, and by filtering out saccades smaller than a given value. This is useful when we are interested in saccades related to cognitive top-down processes, as these tend to cause longer saccades

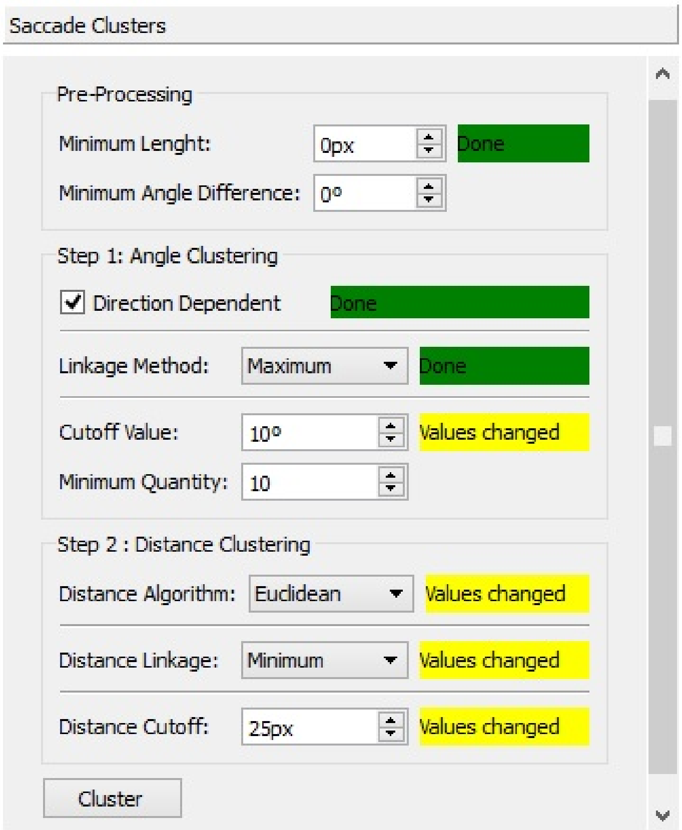
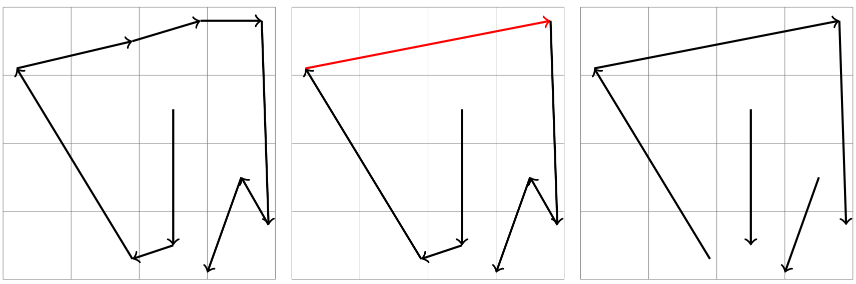


Figure 4.8: Saccades clustering user interface in Eyetrace2014

than bottom-up cognitive factors.

This process is shown in figure 4.9. In this example, the three topmost saccades where merged together, as they are temporally sequential and their orientation angle is similar, resulting in the saccades shown in figure 4.9b. This is done in Eyetrace by changing the value of **Minimum Angle Difference** in the pre processing options.

After that, the length (Euclidean distance between starting and ending points) of each saccade is calculated. If the length of a saccade is smaller than the threshold set by the user in **Minimum Length**, it is removed from the list of saccades to cluster, as shown in figure 4.9c.



(a) Some saccades (b) Merged saccades (c) Filtered saccades

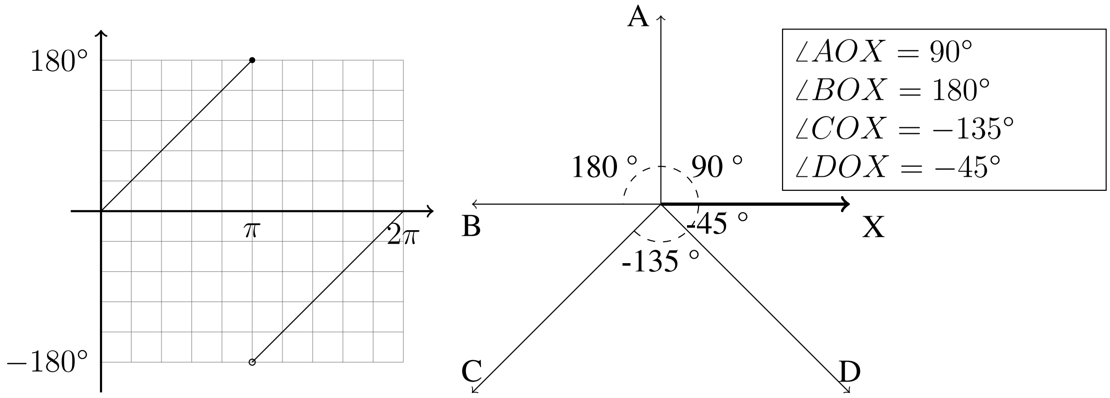
Figure 4.9: Pre processing stage

4.2.3 Clustering by saccade orientation

In order to apply the hierarchical clustering method based on the orientation of the saccades it is necessary to calculate the angle difference between each saccade.

First, the angle in degrees of every saccade in relation to the positive x-axis plane is calculated. Given an arbitrary saccade from point $A(x_a, y_a)$ to point $B(x_b, x_a)$, its angle is calculated with equation 4.2. This returns values in the interval $(-180, 180]$, as shown in the graph in figure 4.10a. Saccades going upward will have positive values, and saccades going downward will have negative values. Figure 4.10b has some examples of angles calculated for four different orientations of saccades.

$$angle(\overline{AB}) = \frac{atan2(y_b - y_a, x_b - x_a) * 180}{\pi} \tag{4.2}$$



(a) Domain of $arctan2$ function (b) Angle of saccades

Figure 4.10: Domain and example of saccades orientation

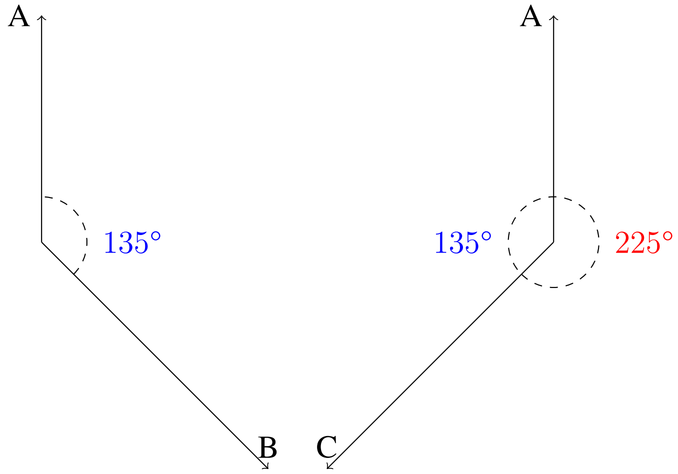
The angle distance between two arbitrary saccades A and B can now be calculated according to the equation 4.3. This distance is used to initialize the dissimilarity matrix of the hierarchical clustering algorithm. Figure 4.11 show one example for each conditional from this equation, with the results shown in blue.

$$d(A,B) = \begin{cases} |angle_a - angle_b| & \text{if } |angle_a - angle_b| \leq 180 \\ 360 - |angle_a - angle_b| & \text{if } |angle_a - angle_b| > 180 \end{cases} \tag{4.3}$$

Furthermore, if the direction each saccade points to is irrelevant, that is, a saccade from point A to point B is considered equal to a saccade from point B to point A, we use equation 4.4 instead. Figure 4.12 show one example for each conditional from this equation, with the results shown in blue. The user may use this distance calculation method by unchecking the **Direction Dependent** checkbox in the clustering options UI.

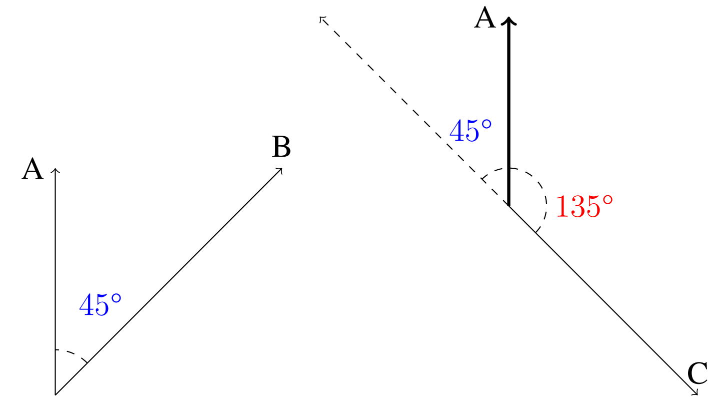
$$d(A,B) = \begin{cases} |angle_a - angle_b| & \text{if } |angle_a - angle_b| \leq 90 \\ |180 - |angle_a - angle_b|| & \text{if } |angle_a - angle_b| \in (90, 270] \\ 360 - |angle_a - angle_b| & \text{if } |angle_a - angle_b| > 270 \end{cases} \tag{4.4}$$

After initializing the dissimilarity matrix, the hierarchy tree is calculated using the linkage criterion chosen by the user in the **Linkage Method** combo box. A maximum

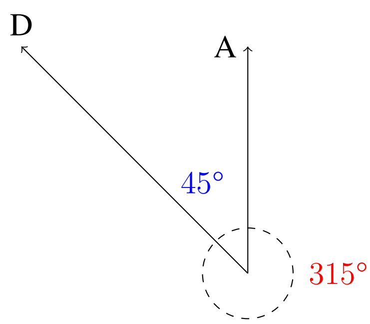


(a) $|angle_a - angle_b| \leq 180$ (b) $|angle_a - angle_b| > 180$

Figure 4.11: Examples of the two conditionals from equation 4.3. The result are shown in blue.



(a) $|angle_a - angle_b| \leq 90$ (b) $|angle_a - angle_b| \in (90, 270]$



(c) $|angle_a - angle_b| > 270$

Figure 4.12: Examples of the three conditionals from equation 4.4. The results are shown in blue.

linkage criterion is recommended for this stage to avoid creating a big cluster containing every saccade.

In figure 4.13 are shown the biggest clusters obtained from figure 4.1 with maximum linkage criterion and direction dependent distance calculation method. The cutoff value used in 4.13a was $1°$, and in 4.13b, $10°$. As can be seen in the figures, with the smaller cutoff values the clusters have less saccades, but their orientation is more similar.

With maximum linkage criterion, a cutoff value x means that each cluster will contain saccades no more distant than x.

4.2.4 Cluster by saccade distance

After clustering by orientation, a list of clusters is obtained, each containing a list of the bundled saccades. If a cluster contains too few saccades, it may be removed from this list. The minimum number of saccades per cluster is set by the user in the **Minimum Quantity** spinbox in the UI. Each one of the remaining lists of saccades is now independently clustered, but this time using the distance between the saccades to create the starting dissimilarity matrix.

The distance between a saccade from point A to B and a saccade from point C to D is given by equation 4.5

$$d(\overline{AB}, \overline{CD}) = \min(d(A, \overline{CD}), d(B, \overline{CD}), d(C, \overline{AB}), d(D, \overline{AB})) \tag{4.5}$$

In order to calculate the distance between a point P and a line segment $\overline{AB}$, the line that extends $\overline{AB}$ is parametrized (equation 4.6a), and the point P is projected into this line with equation 4.6b.

$$L = A + u(B - A) \tag{4.6a}$$

$$u = \frac{(P - A) \cdot (B - A)}{(B - A)^2} \tag{4.6b}$$

The distance from P to $\overline{AB}$ is given by equation 4.7. When $u < 0$ or $u > 1$ the closest point to P is in the extension of $\overline{AB}$, therefore the closest point to P on the edges of $\overline{AB}$ is used to calculate the distance. This can be seen in figure 4.14.

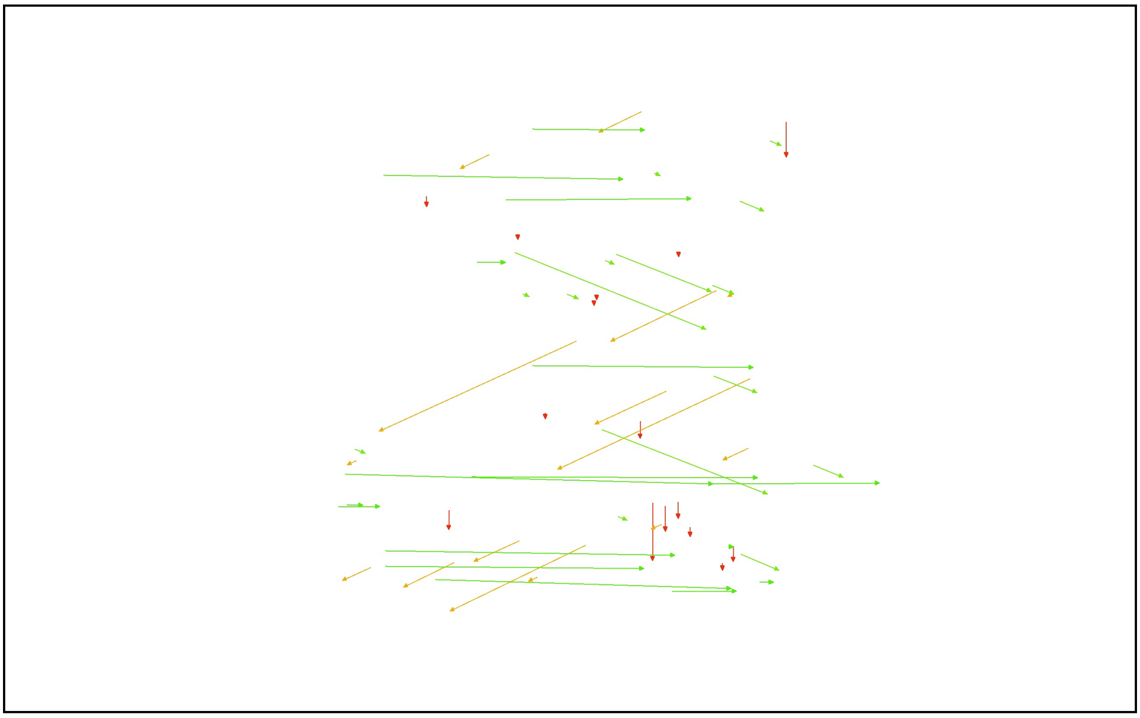
$$d(P, \overline{AB}) = \begin{cases} d(P, A) & \text{if } u < 0 \\ d(P, B) & \text{if } u > 1 \\ d(P, A + u * B) & \text{if } u \in [0, 1] \end{cases} \tag{4.7}$$

Finally, the distance between two points is obtained by calculating the Euclidean distance (equation 4.8a).

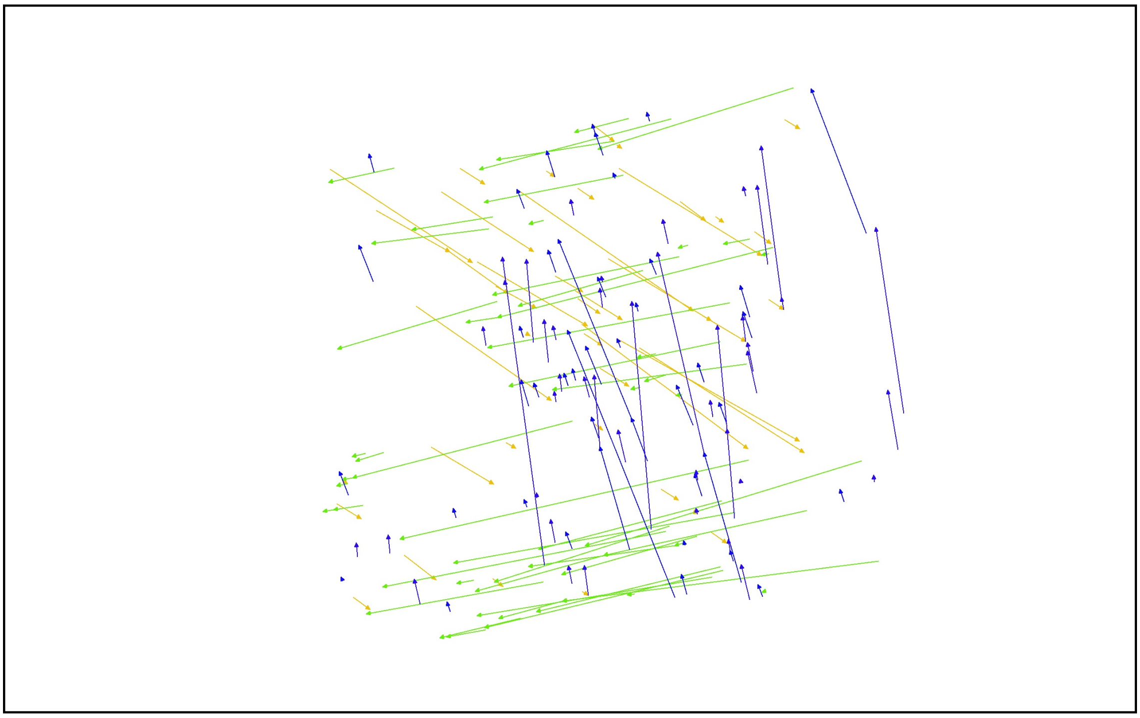
$$d(A, B) = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2} \tag{4.8a}$$

Again, different linkage criteria may be chosen to calculate the distance between clusters. The clusters obtained can vary significantly with the criterion chosen. Figure 4.15 has clusters obtained with maximum linkage and figure 4.16 has clusters obtained with minimum linkage.

Note how with minimum linkage we get a cluster of saccades that extends over longer distances. This is useful to cluster together all saccades in a path. With maximum linkage the clusters will tend to contain only saccades spatially close together.



(a) Clusters with a cutoff value of 1°



(b) Clusters with a cutoff value of 10°

Figure 4.13: Clusters obtained from the saccades from figure 4.1 with, direction dependent distance method and maximum linkage criterion, but different tree cutoff value. To avoid visual clutter only the clusters with most saccades are shown

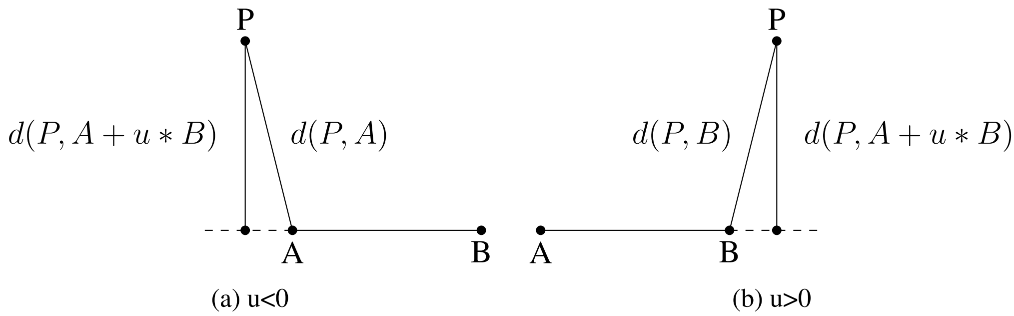


Figure 4.14: Examples of $u < 0$ and $u > 1$

The cutoff value also changes significantly the clusters obtained. If its value is too small, the clusters will not contain many saccades, but if it is too big, the clusters may not be significant. As cutting the tree is done in $O(n)$, the user can easily experiment with this value until an appropriate result is obtained.

This flexibility for selecting distance algorithm, linkage methods and cutoff values is important to adapt the clustering to the kind of data being analysed.

4.3 Algorithm complexity

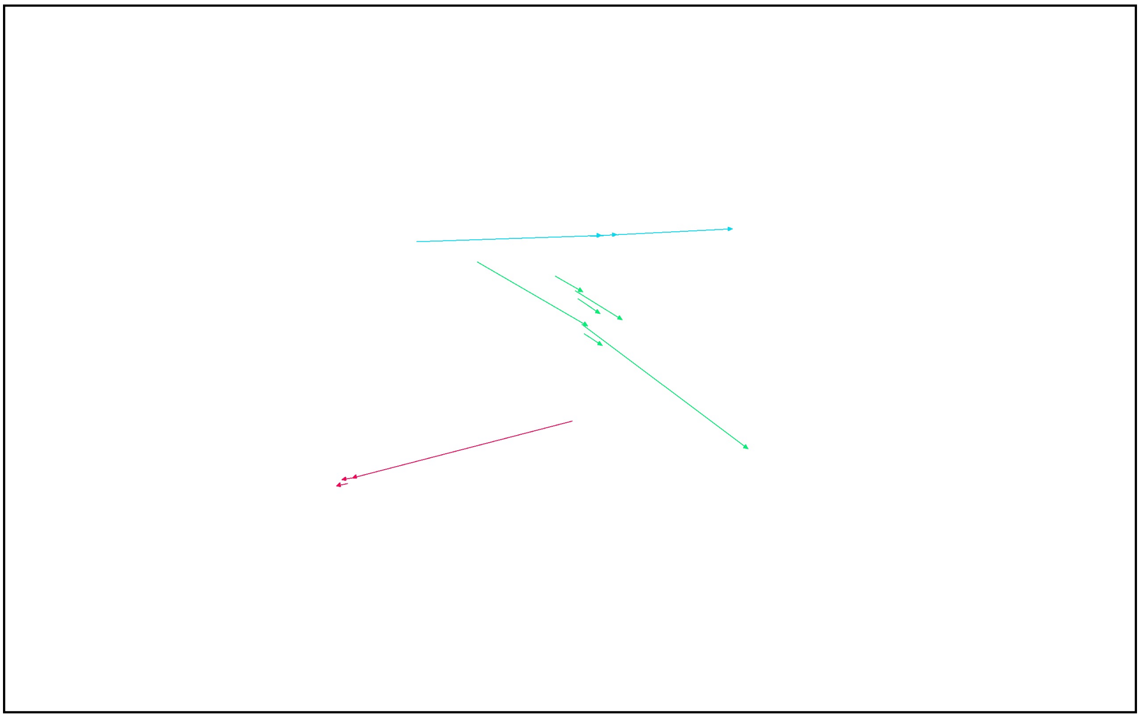
The implemented algorithm for creating hierarchy trees has complexity $O(n^3)$, which makes it slow and impractical for a high volume of objects to cluster. In table 4.2 we show the execution time of the algorithm, with an increasing number of recordings and saccades, in a device with an Intel i5-4200M 2.5GHz processor. The table shows separate running times for the orientation and distance clustering stages. The execution time for the clustering by distance stage is dependent on the cutoff value chosen for the first stage, as it works on the clusters obtained then. In these tests the cutoff value chosen was $10°$.

The running time at the second stage is considerable lower than at the first stage. This happens because, although we still have the same quantity of saccades, they are divided in smaller subsets. The bulk of the execution time is spent creating the first hierarchy tree.

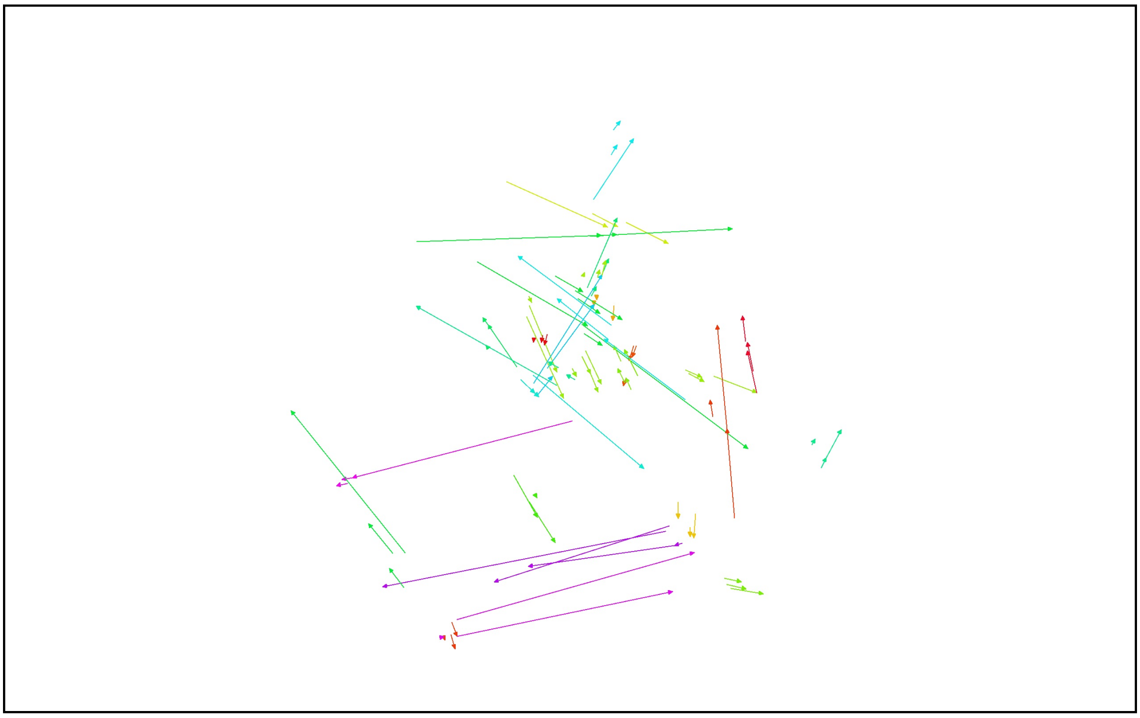
Table 4.2 shows the execution time for the clustering by saccade distance is way lower than for clustering for orientation. This happens because this stage works with smaller lists of saccades than the previous clustering stage. Therefore, after creating the hierarchical tree, new parameters can be tested much faster, as this tree does not need to be calculated again.

Number of subjects	Number of saccades	Execution time of clustering by orientation (s)	Execution time of clustering by distance (s)
1	657	4.341	0.04
2	1297	20.221	0.131
3	1870	58.831	0.26
4	2521	144.365	0.465
5	3143	269.015	0.837
6	3782	506.9	1.266
7	4402	812.629	1.589
8	4981	1151.46	2.109
9	5501	1730.19	3.085

Table 4.2: Clustering execution time with data sets of different sizes

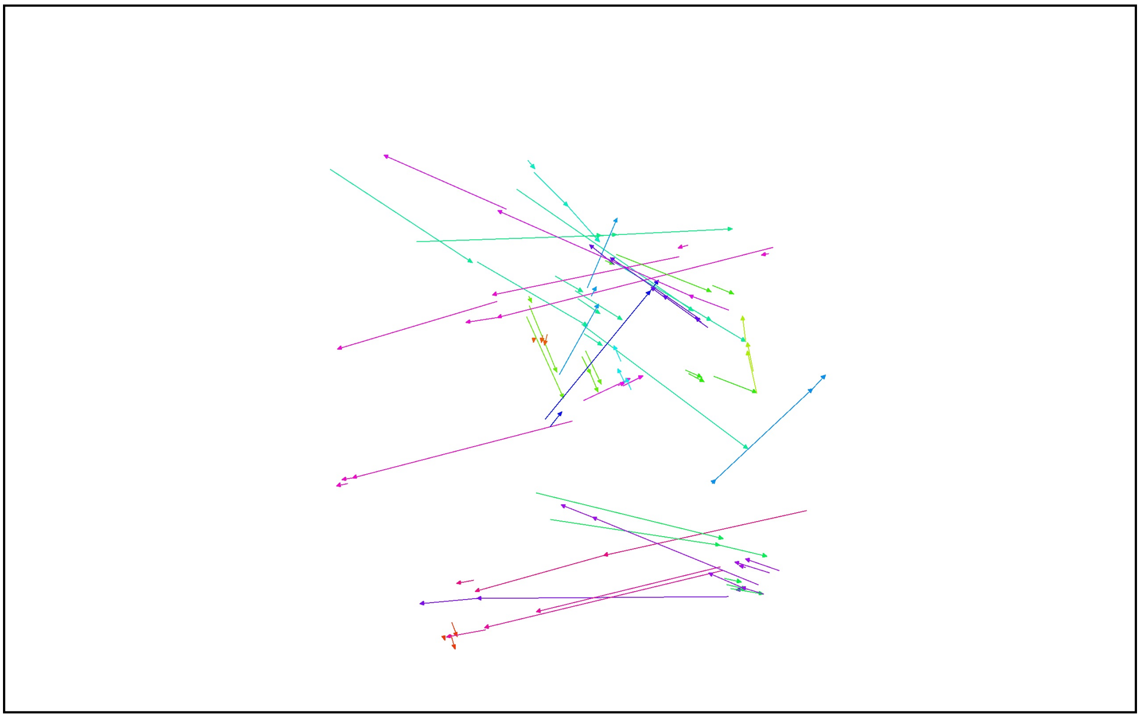


(a) Clusters with a cutoff value of 25 pixels

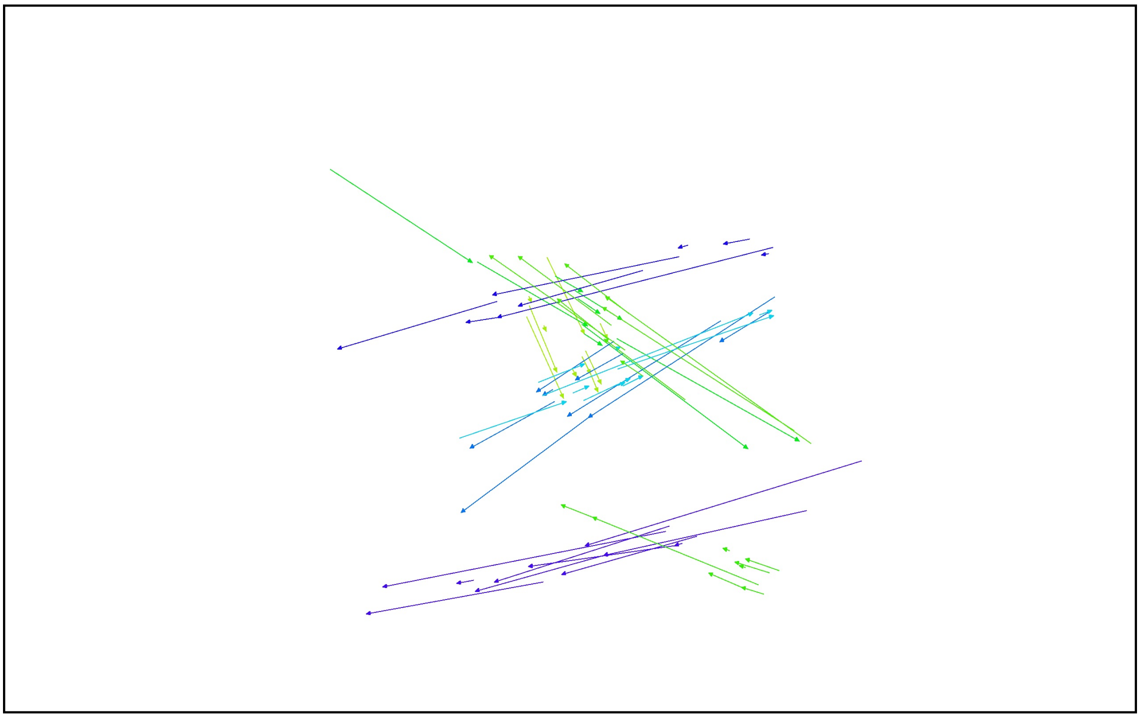


(b) Clusters with a cutoff value of 50 pixels

Figure 4.15: Final clusters of saccades obtained from the saccades from figure 4.1 by using maximum linkage. The orientation were clusters created with maximum linkage, direction dependent distance method, and a tree cutoff value of $10°$. Only the clusters with most saccades are shown.



(a) Clusters with a cutoff value of 25 pixels



(b) Clusters with a cutoff value of 50 pixels

Figure 4.16: Final clusters of saccades obtained from the saccades from figure 4.1 by using minimum linkage. The orientation clusters were created with maximum linkage, direction dependent distance method, and a tree cutoff value of $10°$. Only the clusters with most saccades are shown.

5 RESULTS IN PERCEPTION OF PAINTINGS

The arrangements made by artists in their works of art play an important role in how the subjects' eyes behave when appreciating and interpreting them. Artists use various composition elements in their paintings. These elements are used to influence how the picture is perceived.

One of such elements are the composition lines. Composition lines may be achieved by edges or changes of colors, and they create a path for the eye to transverse through. They are used to guide the eyes of the observers. As an example, composition lines were used by da Vinci in the Last Supper to guide the observer's eyes to Jesus' face (figure 5.1). This makes the analysis of saccadic movements patterns important when studying works of art (ROSENTIEL; KLEIN (2015)). The saccades will tend to form patterns specific to the painting being observed. We hope to find these paths by clustering the saccades of recordings of subjects observing paintings.

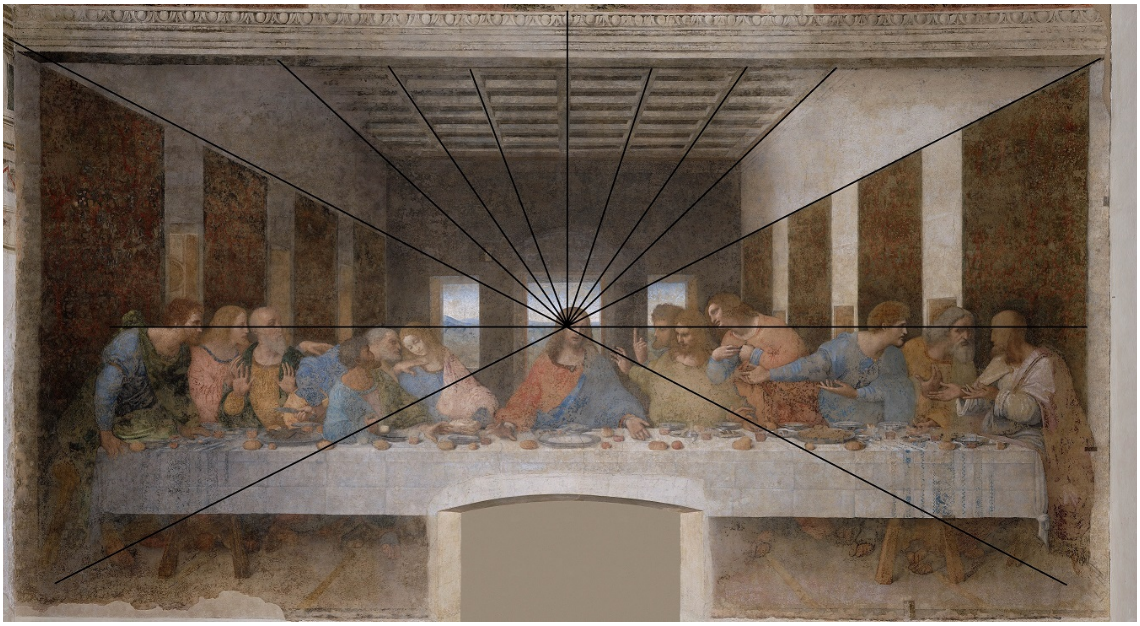


Figure 5.1: Composition lines in Leonardo da Vince's The Last Supper

We validated our clustering algorithm by applying it to three datasets of static image viewing tasks. In these recordings the subjects were instructed to simply observe the image for the duration of the experiment. We analyse the results to see if the clusters obtained are meaningful. No pre processing was done on the saccades data. The orientation clustering stage was done with maximum linkage criterion and direction dependency. The distance clustering stage was done with minimum linkage.

The clusters discussed in the following sections were obtained by first using restrictive cut off values, and progressively easing them until we obtained some clusters that stood out from the others by containing a relatively bigger quantity of saccades.

5.1 Experiment 1: The Conversion of Saul

This data set contains the recording of three subjects (figure 5.3) when observing the painting in figure 5.2, featuring a total of 3182 saccades. The hierarchy tree was calculated in $272.147s$. Figure 5.4 shows the three biggest clusters obtained, each containing at least 10 saccades.

We can observe in these figures, that although the subjects gaze scanned the entire picture, all three subjects gazes have common paths that coincides with the collumns of soldiers walking uphill. This makes sense, given that this is the are of the painting that contains more details and therefore attracts the subjects' attention.

5.2 Experiment 2: St. Denis Preaching the Faith in France

This second experiment was made with the recordings of two subjects (figure 5.6) when observing the painting from figure 5.5, containing in total 2292 saccades. The hierarchy tree for this dataset was calculated in $110.664s$. The biggest clusters obtained are shown in figure 5.7.

This painting has two main characters, the angel on the clouds, and the priest. It is expected that the most relevant saccadic movements occur from or to these two characters. In the scanpath it is only possible to see how the subjects gaze passed through the faces of everyone in the picture, but in the visualization of the clusters we can see three relevant clusters: from the angel to the priest, and from the sitting crowd to both the angel and the priest. Two other unrelated clusters were also found.

5.3 Experiment 3: The Art of Painting

This dataset contains the recording of 9 subjects (figure 5.9) observing the painting shown in figure 5.8. These recordings have a shorter duration and were made with a device with a lower sampling rate than the ones from the previous two experiments. It has a total of 959 saccades. The hierarchy tree was calculated in $8.359s$.

Even though this dataset contains less saccades, its result is expected to be more meaningful, as it contains much more subjects. Besides that, due to its shorter recording duration, the clusters obtained might show if there is a common eye movement between the subjects when first looking at this picture. Figure 5.10 shows the two biggest clusters obtained. The green cluster is of note, as it starts at the girls face and goes to the painter face.

Figure 5.2: The Conversion of Saul by Pieter Brueghel

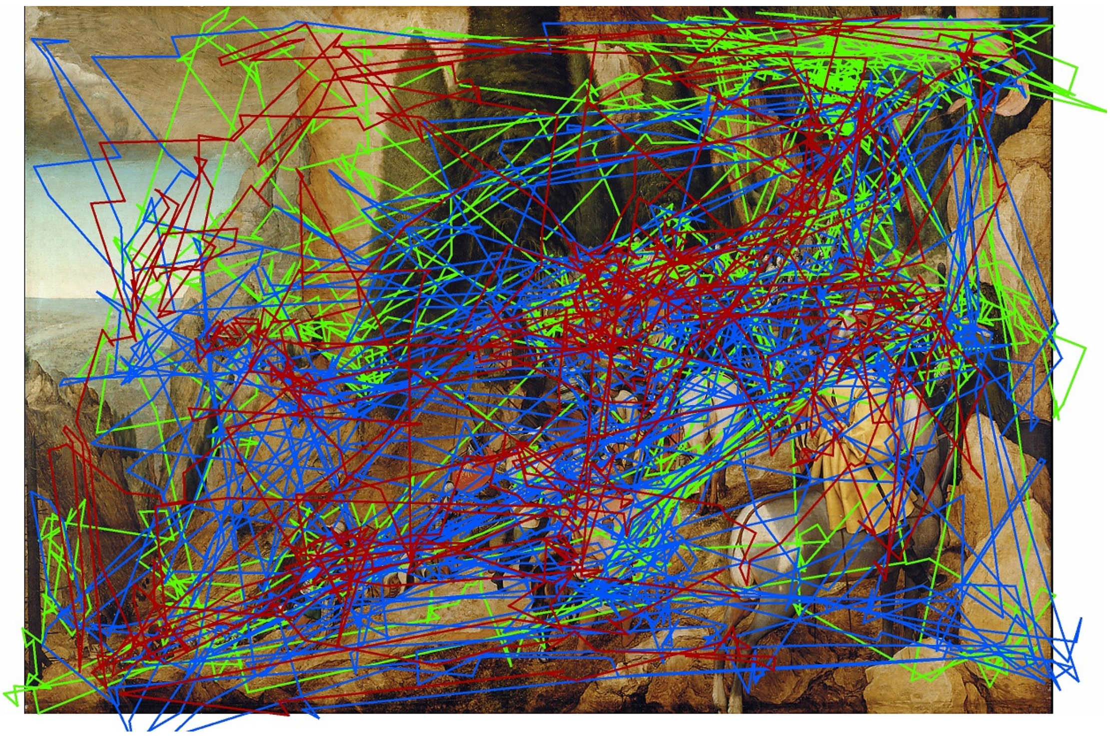


Figure 5.3: Scanpath made by three subjects, shown in different colors

Figure 5.4: Clusters with at least 10 saccades.

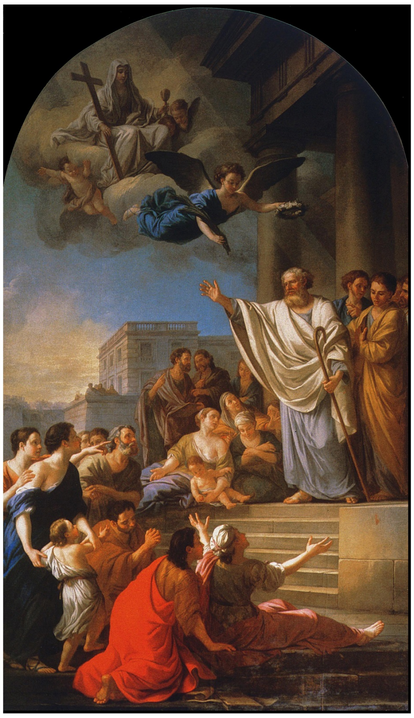


Figure 5.5: St. Denis Preaching the Faith in France by Joseph Marie Vien

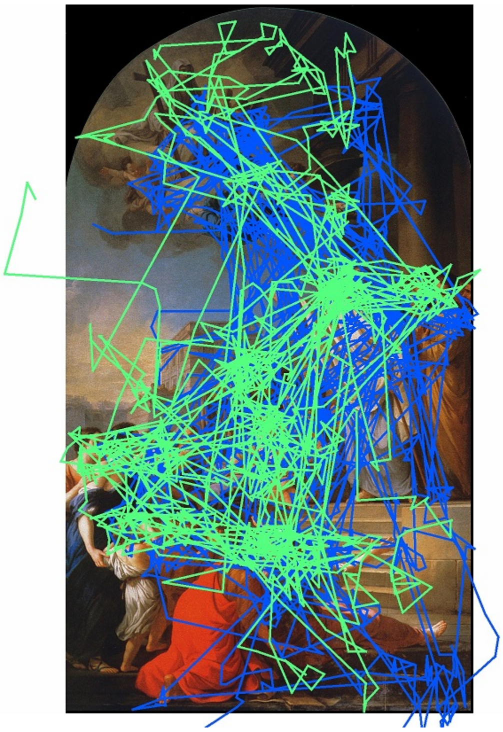


Figure 5.6: Scanpath made by two subjects, shown in different colors

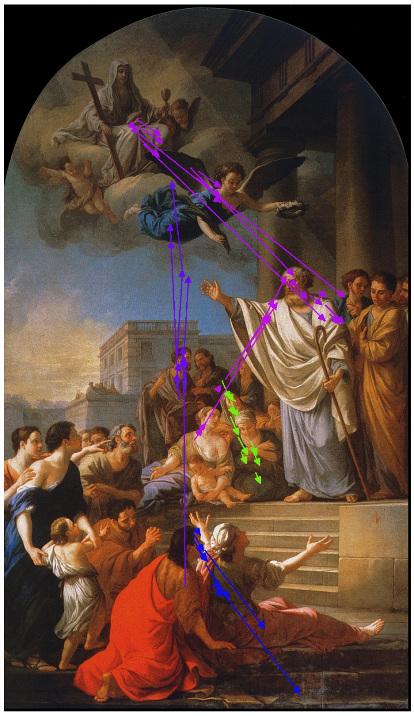


Figure 5.7: Clusters with at least 8 saccades.

Figure 5.8: The Art of Painting by Johannes Vermeer

Figure 5.9: Scanpath made by nine subjects

Figure 5.10: Clusters with at least 8 saccades.

6 DISCUSSION

With this clustering technique we are able to obtain more specific informations than the already existing saccades visualization techniques, namely anglestars and saccades heat map. As discussed in section 2.1.2, it is possible to create an anglestar visualization that shows the most common direction of the saccades. Unfortunately, this does not provide any positional information about these saccades. This problem is addressed with the clusters of saccades. Furthermore, it could be argued that a heat map generated with the saccades could be used instead. Although this kind of heat map will show where saccadic movements occcur more often, it does not takes the orientation of the saccades in consideration and neither permits the quantification of the results like in saccade clustering.

Although the implemented algorithm is fully operational, it was developed in a naive way in order to first validade its results. The current implementation has complexity $O(N^3)$, which makes it impractical for a higher volume of data. There are two hierarchical clustering algorithms that can be implemented to obtain a complexity of $O(N^2)$, although they are specific to a linkage criterion. The algorithms are SLINK (SIBSON (1973)) and CLINK (DEFAYS (1977). They perform hierarchical clustering with minimum and maximum linkage, respectively.

Very recently, PEYSAKHOVICH; HURTER; TELEA (2015) proposed a promising edge bundling method that can be used, by modeling the saccades as a graph, to obtain results similar to our method of saccade clustering. However, currently no implementation is available for testing. Furthermore, this graph based approach uses high-speed eye-tracking data and its implementation is associated with a massive computational load that can only be handled by the GPU, while the approach used in this thesis works on low-frequency data but can process large amounts of data on a standard CPU.

7 CONCLUSION AND FUTURE WORK

In this work a hierarchical clustering method was designed for the bundling of similar saccadic eye movements. Most analysis software focuses only on the eye fixations and performs at most a statistical analysis of the saccades. In these software the saccades are usually visualized as arrows, which creates a lot of visual clutter for a big quantity of saccades. By clustering similar saccades we seek to simplify this data and, as a consequence, be able to extract more meaningful information from it.

Although the experiments performed in this work were focused in the art history domain, it can be applied for a number of other applications in other knowledge domains, such as human-computer interaction, marketing, knowledge engineering and cognitive science.

The saccade clustering algorithm can be used in interface design by identifying the most frequent paths and, based on the obtained data, making sure that the access to the ui elements is not obstructed by any salient element. Furthermore, this information may also be used for a more strategic placement of advertisements.

It may also be useful in order to build efficient visualization techniques. Many times, in order to interpret complex information, various data visualizations are provided for the users. Frequent gaze path information can be useful to detect how efficiently the users are using the information available to them with these visualizations.

Eye tracking could also be used with domain specialist for the development of expert systems. Not only could it be used to help understand the specialist cognitive processes, it could be used to discover the differences in how experts and a laymen react to various stimuli, and which elements trigger gaze transitions and they how can they be influenced.

REFERENCES

BLIGNAUT, P. Fixation identification: the optimum threshold for a dispersion algorithm. **Attention, Perception, & Psychophysics**, [S.l.], v.71, n.4, p.881–895, 2009.

BUSWELL, G. T. **How people look at pictures**. [S.l.]: University of Chicago Press Chicago, 1935.

DEFAYS, D. An efficient algorithm for a complete link method. **The Computer Journal**, [S.l.], v.20, n.4, p.364–366, 1977.

DUCHOWSKI, A. A breadth-first survey of eye-tracking applications. **Behavior Research Methods, Instruments, & Computers**, [S.l.], v.34, n.4, p.455–470, 2002.

DUCHOWSKI, A. **Eye tracking methodology**: theory and practice. [S.l.]: Springer Science & Business Media, 2007. v.373.

ERGONEERS GMBH. **Ergoneers webpage**. [Online; accessed 27-July-2015], `http://www.ergoneers.com/en/`.

FUCHS, A. F. The saccadic system. **The control of eye movements**, [S.l.], p.343–362, 1971.

HENDERSON, J. M.; FERREIRA, F. Eye movement control during reading: fixation measures reflect foveal but not parafoveal processing difficulty. **Canadian Journal of Experimental Psychology/Revue canadienne de psychologie expérimentale**, [S.l.], v.47, n.2, p.201, 1993.

HOLMQVIST, K. et al. **Eye tracking**: a comprehensive guide to methods and measures. [S.l.]: Oxford University Press, 2011.

HUBEL, D. H. **Eye, brain, and vision**. [S.l.]: Scientific American Library New York, 1988. v.22.

IMAI, T. et al. Comparing the accuracy of video-oculography and the scleral search coil system in human eye movement analysis. **Auris Nasus Larynx**, [S.l.], v.32, n.1, p.3–9, 2005.

JACOB, R.; KARN, K. S. Eye tracking in human-computer interaction and usability research: ready to deliver the promises. **Mind**, [S.l.], v.2, n.3, p.4, 2003.

JUST, M. A.; CARPENTER, P. A. Eye fixations and cognitive processes. **Cognitive psychology**, [S.l.], v.8, n.4, p.441–480, 1976.

JUST, M. A.; CARPENTER, P. A. Using eye fixations to study reading comprehension. **New methods in reading comprehension research**, [S.l.], p.151–182, 1984.

KASNECI, E. et al. Driving with binocular visual field loss? A study on a supervised on-road parcours with simultaneous eye and head tracking. **PloS one**, [S.l.], v.9, n.2, p.e87470, 2014.

KUEBLER, T. C. et al. Analysis of eye movements with Eyetrace. **Biomedical Engineering Systems and Technologies**, [S.l.], in press.

NOTON, D.; STARK, L. Eye movements and visual perception. **Scientific American**, [S.l.], 1971.

PEYSAKHOVICH, V.; HURTER, C.; TELEA, A. Attribute-Driven Edge Bundling for General Graphs with Applications in Trail Analysis. **IEEE Pacific Visualization Symposium**, [S.l.], 2015.

PIETERS, R.; WEDEL, M. Attention capture and transfer in advertising: brand, pictorial, and text-size effects. **Journal of Marketing**, [S.l.], v.68, n.2, p.36–50, 2004.

RAYNER, K. Eye movements in reading and information processing: 20 years of research. **Psychological bulletin**, [S.l.], v.124, n.3, p.372, 1998.

REDLINE, C. D.; LANKFORD, C. P. Eye-movement analysis: a new tool for evaluating the design of visually administered instruments. **Proceedings of the Survey Research Methods Section of the American Statistical Association**, [S.l.], 2001.

ROSENTIEL, R.; KLEIN, C. The moving of the eye of the beholder: eye-tracking and the perception of paintings. In: HUSTON, J. P. et al. (Ed.). **Art, Aesthetics and the brain**. [S.l.]: Oxford University Press, 2015.

SALVUCCI, D. D.; GOLDBERG, J. H. Identifying fixations and saccades in eye-tracking protocols. In: EYE TRACKING RESEARCH & APPLICATIONS, 2000., 2000. **Proceedings...** [S.l.: s.n.], 2000. p.71–78.

SANTELLA, A.; DECARLO, D. Robust clustering of eye movement recordings for quantification of visual interest. In: EYE TRACKING RESEARCH & APPLICATIONS, 2004., 2004. **Proceedings...** [S.l.: s.n.], 2004. p.27–34.

SENSOMOTORIC INTRUMENTS GMBH. **SMI webpage**. [Online; accessed 27-July-2015], `http://www.smivision.com/en.html`.

SENSOMOTORIC INTRUMENTS GMBH. **SMI iVIew X Hi-Speed specifications webpage**. [Online; accessed 27-July-2015], `http://www.smivision.com/en/gaze-and-eye-tracking-systems/products/iview-x-hi-speed.html`.

SHEBILSKE, W.; FISHER, D. **Understanding extended discourse through the eyes**: how and why. [S.l.]: Lawrence Erlbaum Associates: Hillsdale, NJ, 1983.

SIBSON, R. SLINK: an optimally efficient algorithm for the single-link cluster method. **The Computer Journal**, [S.l.], v.16, n.1, p.30–34, 1973.

TAFAJ, E. et al. Bayesian online clustering of eye movement data. In: SYMPOSIUM ON EYE TRACKING RESEARCH AND APPLICATIONS, 2012. **Proceedings...** [S.l.: s.n.], 2012. p.285–288.

THE EYE TRIBE APS. **The Eye Tribe webpage**. [Online; accessed 27-July-2015], `https://theeyetribe.com/`.

THE EYE TRIBE APS. **The Eye Tribe specifications webpage**. [Online; accessed 27-July-2015], `https://theeyetribe.com/products/`.

WIDDEL, H. Operational problems in analysing eye movements. **Advances in Psychology**, [S.l.], v.22, p.21–29, 1984.

YARBUS, A. L.; HAIGH, B.; RIGSS, L. A. **Eye movements and vision**. [S.l.]: Plenum press New York, 1967. v.2, n.5.10.