Eye Tracking-Based Desktop Activity Recognition with Conventional Machine Learning

Ole Poeth
Supervisor(s): Guohao Lan, Lingyu Du
EEMCS, Delft University of Technology, The Netherlands

A Dissertation Submitted to EEMCS faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering

Abstract

Cognitive processes have been used in recent years for context sensing and this has shown promising results. Multiple sets of features have shown good performance but no set of features has been determined the best for classifying gaze data. This paper looks at different feature sets and the heterogeneity of gaze signals from subjects and hardware to determine what impacts the performance of the classifiers and what returns the best results. These results are compared with deep learning classifiers using the same data set to determine which performs better.

For the different feature sets, saccade features show great positive influence on the accuracy (88% accuracy) but fixation features show a significant lower ability to classify correctly (63% accuracy), a combination of some fixation and saccade features show the best results(95% accuracy). The way the data is split, has a huge impact on the performance, splitting the data on every activity gives an accuracy of 95%, while the splitting on subjects only reaches a maximum of 60% accuracy. Deep learning algorithms perform only slightly better at 97% accuracy but dropping down massively (38%) when splitting the data over subjects.

The main conclusions from this research revolve around feature selection and subject bias. Saccade features have the most impact on the classification of activity recognition using eye tracking data. Each subject performs each task in a significantly different way which drastically decreases performance when completely new subject data is tested on a trained classifier. Deep learning classifiers show similar results and back up the importance of the heterogeneity of the data. The evaluation of different types of hardware has not been accomplished in this research due to time constraints.

1 Introduction

In the recent past, research on visual behavior has emerged and context sensing through cognitive processes has seen convincing correlation[1]. Eye movement is one of the processes that has been shown to be related to different processes such as emotion[2] and visual memory[3]. Different eye trackers have seen some use in the assistance of gathering data for training purposes in professional gaming[4] and monitoring drivers while driving[5]. Information about the condition and doings of people while performing everyday tasks can be used in a better understanding of what people actually focus on during these tasks. Finding patterns in eye movement in order to classify a set of tasks that people are doing has shown great success, up to 74% [6]. In more recent years, advanced algorithms have emerged that are able to classify tasks without the restrictions of different types of hardware or subjects. GazeGraph [7] recognizes cognitive contexts, e.g. reading or watching a painting, live while wearing an eye tracker within seconds. It shows up to a 45% increase in recognition accuracy and a 80% decrease in system adaption time in comparison to existing solutions.

1.1 Research Questions

Extracting features out of an eye tracker data is a task that has shown different solutions throughout the years and different sets of features are shown to have a good accuracy score using different types of conventional machine learning algorithms, e.g. K-NN, Random Forest and SVM[6], [8]. Although these different sets show great accuracy, there has not been one set of features that is shown to be the best for classifying cognitive contexts. The first step for the research in this paper is trying the find the best feature extraction method to answer the following question: **How to design and implement different feature extraction methods for eye movement signals?**

Having these extracted features is the beginning of the optimization for the classification of desktop activities. After the extraction, the best features have to be determined in order to achieve the best results. This will done using feature selection methods to answer the question: **To achieve good recognition accuracy, what are the best features that need to be extracted and used for conventional machine learning algorithms?**

Heterogeneity in data is something that cannot be ignored [9]. Also in eye tracking data, there is a large difference between subjects and the way they perform the tasks at hand. The heterogeneity of eye tracking data can alter the outcome, especially when using features that negatively impact the performance[10]. Different kinds of eye trackers return different data which can also include a bias that is able to influence the performance. Using different feature sets and ways to split the data, this paper tries to answer the following question: **What is the impact of different subjects and sensing hardware on the recognition performance?**

Steps towards human activity recognition with deep learning models have been made over the last years[9]. Deep learning has the opportunity to better recognize and remember patterns in data, which can be applied on gaze data to match patterns for classification of human activities. Deep learning models are a lot more complex and take longer to train in comparison to conventional algorithms. The last part of this paper will investigate the following: **Compare deep and conventional machine learning algorithms on accuracy and robustness against heterogeneity among subjects.**

1.2 Structure

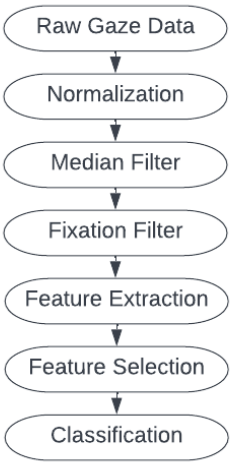
This paper will start by explaining the methodology applied for the experiments in section 2. The section will cover the data set, preprocessing of the data, feature extraction and selection and the classification. Section 3 will continue with the implementation of the methodology for the experiments. It will cover the implementation of the different filters applied on the data, extracted fixation and saccade features, classification results, subject bias calculations and the comparison with deep learning classifiers. Subsequently, a part on responsible

research will show how this paper dealt with privacy and responsibility of the data in section 4. Lastly, in the section 5 all the results from the paper will be discussed together with the limitations of this paper and future work.

2 Methodology

A number of different steps are applied on the raw data to get to the classification of an activity. This section will follow the pipeline from figure 1, explaining the reasoning behind the steps. The specific implementation together with the results of the steps can be found in section 3. In 2.1, the data set used in for this research will be elaborated on, followed by the preprocessing part which consists of the normalization and filtering of the data. Then feature extraction methods and selection of the extracted features will be explained. The last part of this section will cover the classifiers used in this research.

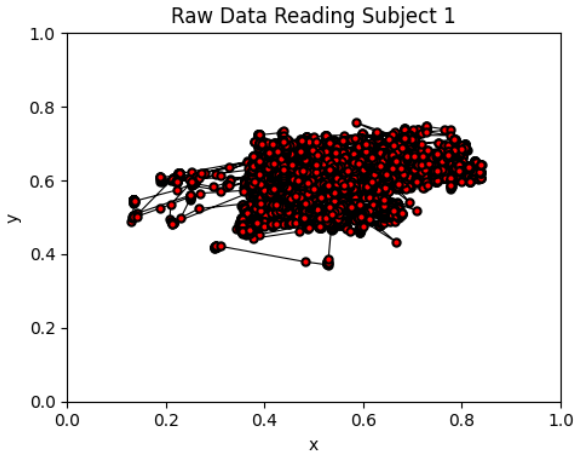
Figure 1: This figure shows the pipeline starting with the raw gaze data down to the classification. In between the data is normalized, filtered and after feature extraction and selection the data is ready to train or test the models.



2.1 Desktop Activity Dataset

The dataset used in this research, is a dataset obtained in the research from G. Lan et al[7]. The gaze data set is sampled from a Pupil Core eye tracker [11] on eight participants (four male, four female, fluent in English and between the age of 24 and 35). Each participant performs six common daily activities on a computer: reading three different types of texts, writing an essay, watching two short videos, freely browsing public websites, e.g. news sites or blogs, playing two different online games (Classic Super Mario and Agario) and searching the internet using a search engine, given a set of predefined questions. All the activities were performed during a time frame of 5 minutes, the participants were able to freely move their head while doing the tasks. The data was sampled at rate of 30 Hz, resulting in a data set of activities containing 9000 data points where each data point has a x and y coordinate representing the gaze relative to the other gazes at a certain time.

Figure 2: This plot shows the raw data points from the reading activity from subject 1.



2.2 Preprocessing

Gaze data contains noise from, e.g. blinking or hardware inaccuracies and this noise can affect the eventual outcome in a negative way. The data can also contain gaps where new data points need be sampled to achieve enough workable data. A filter that can solve these problems is a median filter. The median filter is non-linear filter used in signal processing to remove noise and filter out outliers. The median filter used in this paper takes a window of data points, calculates the geometric distance of all points and samples the point with the lowest distance. The geometric distance is the sum of the distances to all other points in the window.

Two distinct gaze features have shown to be a profound basis for gaze based activity recognition [6]–[8]: fixations and saccades.

Fixation. A fixation is a point in the data which represents a moment in time where the subject is looking at one specific part of the screen for an extended period of time. To extract fixations from the data, the fixation filter from [12] will be used. This filter gives the control over the average window size and peak threshold, together with the distance between the fixations, the radius. For every point the filter takes the mean of a window from before and after the point and calculates the difference. The filter compares these differences of consecutive sliding windows, if there is a sliding window that has a higher difference than the sliding window before and after, it is considered a peak and the point is added to a list as a potential fixation. Then for every peak in the list, it is checked whether any peaks are within a sliding windows' distance of each other. If this is the case, the peak with the highest value is selected. Following, only the peaks that are above the threshold will be used to determine the fixations. Lastly, the rest of the remaining points have to be at least the length of the radius parameter apart, otherwise they get merged into

one fixation. This will return a list of points where each point resembles fixation in the data.

Saccade. A saccade is the rapid movement between two fixations when a person switches focus. Due to the relatively low sampling rate of the data, the saccades are extracted by connecting the fixations. This results in all the saccades having the same duration which means that the duration cannot be used as a feature for classification. Fortunately, there is enough information left on the fixations and saccades to extract useful features to differentiate activities on.

2.3 Feature Extraction and Selection

From the fixations and saccades that were extracted during the preprocessing phase different types of gaze features can be extracted. In [8], a distinction is made between low, mid and high-level gaze features. Low-level gaze features are features that can be extracted directly from the fixations and saccades, e.g. fixation duration or saccade length. Mid-level features consist of patterns of multiple fixations or saccades, but these features show a significantly low increase in accuracy and were therefor omitted from the list of extracted features in this research. High-level features take their information from the Areas-of-Interest in the interface but are also beyond the scope of this research.

The features that are extracted from the fixations and saccades are shown in table 1, summing up to a total of 18 features.

Category	*Sub-Category*	*Features*
Fixation	Duration	fix-dur-mean, fix-dur-var, fix-dur-std
	Rate	fix-rate
	Slope	fix-slope
	Dispersion Area	fix-disp-area
	Radius	fix-radius
Saccade	Length	sac-len-mean, sac-len-var, sac-len-std
	Direction	sac-dir-nne, sac-dir-ene, sac-dir-ese, sac-dir-sse, sac-dir-ssw, sac-dir-wsw, sac-dir-wnw, sac-dir-nnw

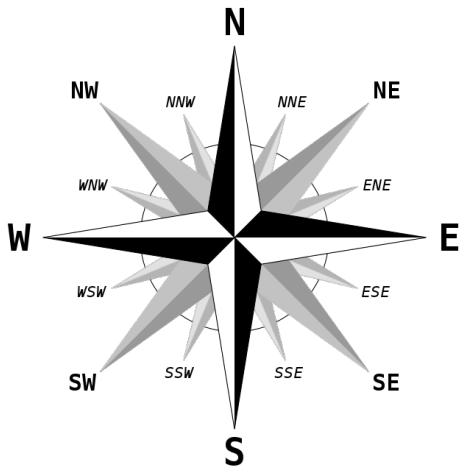
Table 1: Table containing features on fixations and saccades extracted from the data

All features are calculated over a list of fixations and saccades which can vary in length by using different sliding window sizes. The features are divided into the two distinct gaze features: fixation and saccade.

Fixation features. The fixation duration feature takes the duration length of the fixations and gives the mean, variance and standard deviation. The fixation rate is the number of fixations per second and the fixation slope is the slope of the best fitted two dimensional line of the given fixations, resembling the direction of the fixations. The dispersion area of the fixations is the area that 75% of the fixations in the window span. The 25% furthest fixations from the mean of all the fixations are not included in the calculation of the dispersion area. For the fixation radius the distances from the fixations to the mean are again used but now from all fixations and the largest distance is used to represent the radius that all fixations lie within.

Saccade features. The length of a saccade is calculated by the euclidean distance between the two fixations that determine a saccade. From all the saccade lengths in a window the mean, variance and standard deviation are used as features. The second category of saccade features is the direction. For the direction of a saccade the angle between two fixations is used and the angles are split in eight different directions following a compass. The sections the compass has been divided in are following the half-winds as the center of the section. When the angle of a saccade is between the cardinal north and inter cardinal northeast, the direction is set to the half-wind north-northeast (NNE). The total number of saccades per direction is summed up and normalized resulting in every direction feature being between 0 and 1.

Figure 3: This compass is a visualization of the direction the saccades have been divided in. If the angle of the saccade is between north and northeast, the direction is NNE, etc.



For the feature selection different metrics were used to determine the importance of all the different features. The features were tested under different circumstances in order to avoid bias to certain features for specific situations in the data. The results and feature selection methods can be found in section 3.2. The selected features can then be used in different conventional machine learning classifiers to obtain the accuracy on the data set.

2.4 Classification

The machine learning algorithms used for determining feature importance and classifying the activities are the following: k-Nearest Neighbours (k-NN), Support Vector Machine (SVM) and Random Forest (RF). Although the k-NN has shown to have the lowest performance out of the three classifiers [8], probably due to the high complexity in the number of features, it is still a well-performing classifier to act as a benchmark. In section 3 the k-NN will also be tested with less

features to see if reducing the dimensionalilty of the features improves the performance. Multiple classifiers were selected to compare and find consistent results and to see which classifier performed the best using different feature sets on the data. For the training of the classifiers, k-fold cross validation was used to avoid as much bias in the data as possible.

3 Experimental Setup and Results

In this section the complete experimental setup with all the specific parameters and results from different experiments will be explained. Firstly, in the preprocessing part results on different parameter settings for the median and fixation filter will be discussed together with the final parameters that are used to create the filtered data for the feature extraction. Then the way the features are extracted is explained together with experiments on these features to determine feature importance and show the scale of the impact of every feature and if it has a negative or positive impact. After this, different feature sets will be tested using different classifiers and show which sets perform the best on which classifier. Then, three conventional machine learning classifiers are trained and tested using the best before determined parameters and features on two different splits regarding the data set. One where the data is split on activities and the other on subjects to test if the heterogeneity of the data has an impact of the performance. To conclude, the conventional algorithms are compared with deep learning algorithms on the same data set and splits.

3.1 Data Filtering

The filter used for removing outliers in the data is a median filter that uses a sliding window that moves over the data one by one and for every point in the data set, selects the geometric median[1] from the sliding window. The size of the sliding window has a great impact on the filtering of the data. If the sliding window is too big, the same point will be selected a great number of times in row, resulting in unnaturally high concentration of points, which also results in fixations lasting seconds and the fixations duration having a high variance. This will ultimately lead to losing characteristics from the data and the features extracted will not be accurate anymore. The final number of points used for the median filter is 6. This results in the following filtering of the raw data.

From figures 4 and 5 can be concluded that the median filter removes a lot of outliers and noise from the data while still maintaining the characteristics from the data in order to be able to retrieve accurate fixations and saccades.

The extraction of the fixations and saccades from the previously filtered data is done by using the fixation filter from [12]. The filter uses 3 parameters to extract the fixations: sliding window size, threshold and radius. The average fixation duration can range from 300 to 500 ms as stated in [13], where the average greatly depends on multiple factors which are not known or cannot even be determined, e.g. intend of the subject. This means the accuracy might have some small error but as long as the fixation duration lies within these

[1]https://en.wikipedia.org/wiki/Geometric_median

Figure 4: This figure shows the x and y position over time. This is the raw data from subject 1 doing the reading activity without any filtering applied.

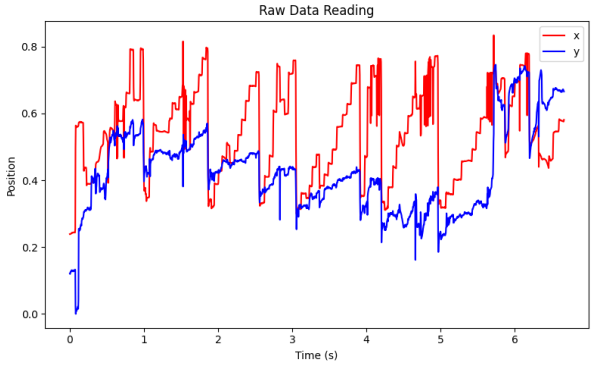
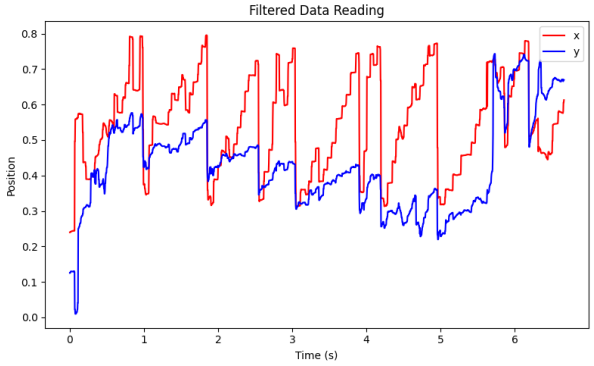


Figure 5: This figure shows the x and y position over time. This is the raw data from subject 1 doing the reading activity with a median filter applied using a sliding windows size of 6.



boundaries, the results are assumed to not suffer significantly from these factors. After trying different combinations of parameters a sliding window size of 7, a threshold of 0.003 and a radius of 0.001 resulted in the best fixations. Table 2 shows the average and standard deviation of the fixation duration for every activity. The results are averaged over the subjects.

Activities	*Fixation Duration Avg*	*Fixation Duration Std*
Read	408,9	139,0
Write	426,5	162,5
Watch	437,3	185,9
Play	409,8	147,8
Browse	402,9	139,6
Search	411,8	153,3

Table 2: This table shows the average and the standard deviation of the fixations extracted by the fixation filter for every activity averaged over all subjects.

Now that the data is filtered and the fixations are correctly extracted, the fixations can be used to extract more meaningful features which can be used the determine the difference between activities.

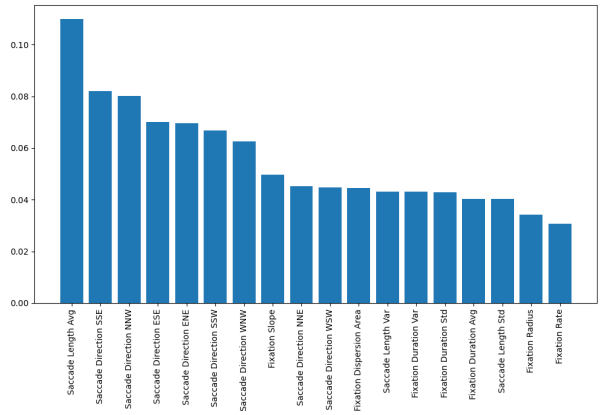
3.2 Fixation and Saccade Features

One of the downsides of the data set is the small number of subjects and the relatively small time window recorded for each activity. At five minutes per activity per subject, sampling the features per activity only results in eight feature samples per activity to train and test the classifiers on. To get more samples, a sliding window can be applied on the data of each activity where the features are extracted over this window. Different window sizes have been tested by [8] and a sliding window size of 105 seconds shows the best results regarding classifying accuracy.

Still 105 seconds only results in three times more samples as it is more than 1/3 of the total time per activity. By applying an overlap for the sliding windows, enough samples can extracted. Each sliding window uses a portion of the previous one and some new data to calculate the features over. An overlap of 95% shows good performance on both accuracy and stability of the classifiers and will be used in the latter of the experiments.

The full list of extracted features can be found in table 1, these features have shown to have good performance [8] in classifying gaze base activities across different classifiers, e.g. SVM and Random Forest. The paper mainly focuses on the addition of the mid-level features and although they do show feature important for a portion of the features used, no further information is given on the performance when omitting some of the features. In order to get the most ideal set of features, different sets of features have to be tested by different classifiers. The Random Forest classifier has the ability to return an importance value of each feature after training, where the higher the value, the more important the feature.

Figure 6: This figure shows the importance value of every feature after training the Random Forest classifier.



The Random Forest is trained with all the data, has no maximum depth for the leaves and uses 100 trees for its classification. A k-fold cross validation is used for dividing the data, with k=4. To avoid subject bias, the splitting of the k-fold cross validation is done on each activity from each subject where 3/4 of the data is used for training and 1/4 for testing purposes. Four different estimators are trained with each a different combinations of training and testing parts to avoid a bias. The results are averaged over the 4 estimators.

From figure 6 it can concluded that the features revolving around fixations have the least impact on the classification. Almost all saccade features have a good impact on the Random Forest classifier, with the saccade length average having the most. Training the classifier with different sets of features confirms the findings in the feature importance graph, results of these experiments are summarized in table 3. Using only features that revolve around fixations the performance massively drops by 0.29 to even 0.36 on the SVM, k-NN and Random Forest classifiers. Saccade features do a much better job with only a 0.03 to 0.07 decrease in accuracy. The feature set that performs the best is the one consisting of all features except the fixation radius. There is no degradation in performance and the k-NN classifier performs even slightly better.

Feature Sets	*All*	*Fixation*	*Saccade*	*Best*
Classifiers				
SVM	0.95	0.59	0.88	0.95
k-NN	0.84	0.49	0.81	0.86
Random Forest	0.92	0.63	0.86	0.92

Table 3: This table shows the accuracy of the three classifiers SVM(c=1000, kernel=rbf), k-NN(nn=20) and Random Forest using different sets of features. The second column is run using all features, the third column using only fixation features, the fourth column only saccade features and the last column the features with the best result based on the feature importance, which consists of all features expect the ones around fixation radius.

3.3 Conventional Classifiers

For the final classification of the activities, the SVM, k-NN and RF classifiers are used. After the feature selection, the following features showed to serve the best results and will be used in the latter experiments: Fixation rate, fixation slope, fixation dispersion area, fixation radius, saccade length and saccade direction. For the median filtering of the data a window size of 6, for the fixation filter a window size of 7, a peak threshold of 0.003 and a radius of 0.001. In order to make the classification work for every classifier, the features are scaled down to a value between 0 and 1. The final best results can be found in table 3, column 5. SVM, k-NN and RF scoring respectively 0.95, 0.86 and 0.92 accuracy for on the whole data set while this was split over activities.

3.4 Subject Bias

Humans perceive objects and activities differently, human visual behavior also differs based on personal interest [14], but also on visual stimuli [6], [8]. This heterogeneity can create a large subject bias in the data which could affect the performance of the classifiers significantly. The splitting of the data for training and testing sets in section 3.3 is done per activity. This means the classifiers have information on each subject for each activity, making classifying new data easier as the classifiers know for each subject how they perform each activity. If the data is split in a different way, where all the data

from six subjects is used for training the classifiers and two complete new subjects are used for the testing set, the performance drop is immense.

Classifiers	*Data Split On Activities*	*Data Split On Subjects*
SVM	0.95	0.60
k-NN	0.84	0.54
Random Forest	0.92	0.58

Table 4: This table shows the test set accuracy of three different conventional machine learning classifiers, SVM, k-NN and RF, when the data is split on the activities versus when the data is split on subjects.

Table 4 shows the significant drop in performance when the classifiers do not have any information on the new subjects and their activities. The performance drop off is 0.35, 0.30 and 0.32 respectively for the SVM, k-NN and Random Forest classifiers.

While the accuracy sharply drops when classifying activities from complete unknown subject, the feature importance does not differ from the feature importance when the data is split on the activities. Fixation radius still does not add much value to the training while the other features have the same impact on the training as found in section 3.2. This makes the heterogeneity of the data the strongest component when it comes to performance drop.

3.5 Deep Learning Classifiers

Another way to accomplish gaze based activity recognition is through deep learning algorithms [15]. During this research, two other papers focused on the same data set that is used in this paper but applying different deep learning algorithms to test their performance. A LSTM[2] and a CNN[3] were tested on the data set by respectively [16] and [17]. Both have split the data on activities and subjects as well resulting in a better comparison between deep learning and conventional machine learning algorithms.

	SVM	*LSTM*	*CNN*
Activities	0.95	0.95	0.97
Subjects	0.55	0.32	0.38

Table 5: This table shows the test set accuracy between the SVM, LSTM and CNN on the same data set, split over activities and over subjects.

From table 5 it can be concluded that there not is a large difference when comparing deep learning and conventional machine learning classifiers. When the data is split on activities and the classifiers have information on every activity on every subject, the deep learning classifiers do not outperform the conventional classifiers by much. On the contrary, both deep learning algorithms show substantial decrease in performance when the data is split over subjects, performing worse than the conventional algorithms. Once more does this show the influence of subject bias in the data on the performance of activity recognition using gaze data.

Nevertheless, the deep learning algorithms do show great potential, especially in the consistency of results where the conventional algorithms show a large variance in results with different hyper parameters for the specific features. Deep learning algorithms do not depend on features and only look at the data without having to know much about the data itself, resulting in more robust learning. One downside of the deep learning algorithms is the substantial training time which can take hours where the conventional machine learning algorithms only take around 30 seconds to train.

4 Responsible Research

Working with machine learning especially in combination with something privacy invading like eye tracking comes with responsibilities for the researcher. The gathering of the data alone is already intrusive, where the subjects need to put on an eye tracker on their head while performing the tasks. This results also into one of the limitations for this research due to the short amount of time you can put someone under these conditions. To ensure no sensitive information is published, the subjects are anonymous and only information about the diversity of the subjects is given and known. There is no possible way to determine which person belongs to which subject in the data which means from the data only the gazes are used and no other external information on the subject.

The data used in this paper is extracted by a secondary resource [7], the gathering of the data was approved by the Institutional Review Board of the paper and only general data, e.g. age and gender was shared to show diversity which cannot be linked to the subjects directly. No data is left out in this research, the data is only altered to fit the data to the needs of this research in order to get useful results. All the altering done on the data and the reasoning behind this, can be found in this paper in the sections 2 and 3.

5 Conclusions and Future Work

The introduction of this paper explained the research gaps on optimal features, subject and hardware bias and deep learning algorithms. In this paper different feature extraction and selection methods are tested and compared, heterogeneity of gaze data is shown to have great impact on performance and deep learning models did not show improvement in comparison to conventional models. This section will analyse the four questions raised in the beginning of this paper, go over the results and answer each of them to conclude this paper. Finally, some limitations and future work will be discussed.

5.1 Research Conclusions

How to design and implement different feature extraction methods for eye movement signals? After reading different papers on feature extraction for gaze data, the two distinct features all of them have in common are fixations and saccades. The fixation filter by [12] gives great control over the

[2]https://en.wikipedia.org/wiki/Long_short-term_memory

[3]https://en.wikipedia.org/wiki/Convolutional_neural_network

way the fixations and saccades are extracted by being able to alter average window size, peak threshold and radius. From these fixations and saccades, different types of gaze features can be extracted to be able to differentiate activities with conventional classifiers.

To achieve good recognition accuracy, what are the best features that need to be extracted and used for conventional machine learning algorithms? The best features to use for activity recognition using gaze data revolve around saccades. Due to the fact that saccades inherently focus more on patterns, they perform significantly better than fixation features. Every person reading from left to right does this in kind of the same pattern, but not at the same speed or focus level. Fixation features have roughly the same values over different activities and they are not consistently different over subjects and activities. Saccade features on their own already show good performance but they do need some fixation features to work optimally. Fixation features on their own perform poorly and they need the combination with saccade features to perform well.

What is the impact of different subjects and sensing hardware on the recognition performance? The heterogeneity of the data under subjects show great difference in performance. The different ways each subject performs a task, is probably why fixation features work so poorly as well. When the data is split on each activity for training and testing purposes, the classifier knows how every subject performs each activity, which means it is easier for the classifier to determine what activity a subject is doing when presented with the testing data. When the data is split over subjects and the classifier is trained only on six of the eight subjects, the two unknown subjects differ too much in the way they perform each of the tasks, it becomes significantly more difficult to classify the activities correctly.

Compare deep and conventional machine learning algorithms on accuracy and robustness against heterogeneity among subjects. The deep learning algorithms used for comparison in this paper also show great performance regarding the accuracy of classification on the data set. On the contrary, they do both show poor robustness against heterogeneity among subjects as well as the conventional algorithms, where the conventional models perform even better than the deep learning models. On top of that, the deep learning algorithms take over 100 times longer time to train than the conventional machine learning algorithms, making them not the preferred option for the classification of this data set.

5.2 Limitations and Future Work

One of the limitations in this research is regarding the data set, this is relatively small, especially when trying the generalize over people. Due to the small amount of time every activity is recorded for every subject, there is not a lot of data to train and test the classifiers on. By applying a sliding window with overlap this is somewhat fixed when splitting the data over the activities, but the problem remains when trying to generalize over subjects. Then there are just not enough diverse subjects to train the classifier enough to be able to determine data from a new unknown subject. This problem can simply be reduced by gathering more data from different subjects, although due to reasonably high interference with the privacy of people, the actual gathering of the data might not be that simple.

Another limitation in the research revolves around feature extraction and selection. This paper tried to find the best features with the least amount of information as possible. Different papers [7], [8] have shown a variety of features to still be tested and improved upon in order to increase performance. More advanced patterns on fixations and saccades can be added to the features, but also gaze data in combination with visual recognition of the screen the person is looking at, can improve the accuracy of activity recognition. Using more external information can lead to greater privacy conflicts which is important to consider when building upon this research.

The last limitation in this research regards the difference of sensing hardware. Due to time constraints it was not possible to do any research on this topic and this can still be done in future works using the information from the research.

References

[1] A. Bulling and T. O. Zander, "Cognition-aware computing," *IEEE Pervasive Computing*, vol. 13, no. 3, pp. 80–83, 2014. DOI: 10.1109/MPRV.2014.42.

[2] K. Kassem, J. Salah, Y. Abdrabou, *et al.*, "Diva: Exploring the usage of pupil diameter to elicit valence and arousal," in *Proceedings of the 16th International Conference on Mobile and Ubiquitous Multimedia*, ser. MUM '17, 2017, pp. 273–278, ISBN: 9781450353786. DOI: 10.1145/3152832.3152836. [Online]. Available: https://doi.org/10.1145/3152832.3152836.

[3] A. Bulling and D. Roggen, "Recognition of visual memory recall processes using eye movement analysis," in *Proceedings of the 13th International Conference on Ubiquitous Computing*, ser. UbiComp '11, 2011, pp. 455–464, ISBN: 9781450306300. DOI: 10.1145/2030112.2030172. [Online]. Available: https://doi.org/10.1145/2030112.2030172.

[4] I. Grabska-Gradzińska and J. K. Argasiński, "Patterns in video games analysis – application of eye-tracker and electrodermal activity (eda) sensor," in *Artificial Intelligence and Soft Computing*, L. Rutkowski, R. Scherer, M. Korytkowski, W. Pedrycz, R. Tadeusiewicz, and J. M. Zurada, Eds., 2018, pp. 619–629.

[5] R. Coetzer and G. Hancke, "Eye detection for a real-time vehicle driver fatigue monitoring system," in *2011 IEEE Intelligent Vehicles Symposium (IV)*, 2011, pp. 66–71. DOI: 10.1109/IVS.2011.5940406.

[6] K. Kunze, "I know what you are reading: Recognition of document types using mobile eye tracking," *Proceedings of the 2013 international symposium on wearable computers*, 2013.

[7] G. Lan, "Gazegraph: Graph-based few-shot cognitive context sensing from human visual behavior," *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, 2020.

[8] S. Namrata, "Combining low and mid-level gaze features for desktop activity recognition," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 2.4*, 2018.

[9] J. Fan, F. Han, and H. Liu, "Challenges of big data analysis," *National Science Review*, vol. 1, no. 2, pp. 293–314, 2014. DOI: 10.1093/nsr/nwt032. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84919389078&doi=10.1093%2fnsr%2fnwt032&partnerID=40&md5=a8abe4f11ce84fd5f5d5632fa7cd5c25.

[10] J. Karolus, P. W. Wozniak, L. L. Chuang, and A. Schmidt, "Robust gaze features for enabling language proficiency awareness," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, ser. CHI '17, 2017, pp. 2998–3010, ISBN: 9781450346559. DOI: 10.1145/3025453.3025601. [Online]. Available: https://doi.org/10.1145/3025453.3025601.

[11] M. Kassner, W. Patera, and A. Bulling, "Pupil: An open source platform for pervasive eye tracking and mobile gaze-based interaction," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, 2014, pp. 1151–1160, ISBN: 9781450330473. DOI: 10.1145/2638728.2641695. [Online]. Available: https://doi.org/10.1145/2638728.2641695.

[12] O. Pontus, "Real-time and offline filters for eye tracking.," 2007.

[13] C. L. E. Thimothy A. Salthouse, "Determinants of eye-fixation duration," *The American Journal of Psychology*, vol. 2, pp. 304–34, 1980.

[14] Y. Li, P. Xu, D. Lagun, and V. Navalpakkam, "Towards measuring and inferring user interest from gaze," in *Proceedings of the 26th International Conference on World Wide Web Companion*, International World Wide Web Conferences Steering Committee, 2017, pp. 525–533, ISBN: 9781450349147. DOI: 10.1145/3041021.3054182. [Online]. Available: https://doi.org/10.1145/3041021.3054182.

[15] J. de Lope and M. Graña, "Deep transfer learning-based gaze tracking for behavioral activity recognition," *Neurocomputing*, vol. 500, pp. 518–527, 2022, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2021.06.100. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231222006403.

[16] K. Vaessen, "Gaze-based activity recognition with a lstm."

[17] B. Brockbernd, "Cognitive activity recognition by analyzing eye movement with convolutional neural networks."