TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies
Department of Software Science

Jekaterina Viltšenko 179556IAIB

Analysis of Semantic and Kinematic Features of the Clock Drawing Test

Bachelor's Thesis

Supervisor: Sven Nõmm
PhD

Tallinn 2021

Tallinna Tehnikaülikool
Tarkvaraarenduse Instituut
Infotehnoloogia teaduskond

Jekaterina Viltšenko 179556IAIB

Kella joonistamise testi semantiliste ja kinemaatiliste parameetrite analüüs

Bakalaureusetöö

Juhendaja: Sven Nõmm
PhD

Tallinn 2021

Author's declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author:	Jekaterina Viltšenko
Date:	May 25, 2021

Abstract

Neurological disorder assessment tests, such as the Clock Drawing Test, have been used by neurology specialists as a manual tool for spotting signs of dementia and evaluating its development levels for many years. Recent technological breakthroughs allow neurology specialists to conduct such tests on tablet computers, expanding the possibilities to compute and analyze test parameters faster, discovering patterns previously unnoticed by a human eye.

The clock drawing test analysis with technological appliances has been a relatively prominent research topic in the academia. Previous works on the topic provide a thorough overview of the quantitative and semantic elements of the clock drawing test, introducing different approaches for extracting the elements of the test. Due to the rapid technological change and new data becoming available, the present work aims to enhance prior research with the contemporary methodology and additional kinematic analysis.

The primary goal of the current thesis is to analyze semantic and kinematic features found in the Clock Drawing Test and, according to our discoveries, describe the differences in the parameters describing the test process between the groups of healthy controls subjects, individuals diagnosed with Parkinson's disease and patients with mild cognitive impairment.

Achieving the main goal requires a classification system for the elements of the drawing. A combination of advanced deep learning and classical machine learning techniques is used for the test element extraction. The final result is assessed through performance of the classifiers, trained and validated on the element features with the highest discriminatory power between the subjected group.

The thesis is written in English and contains 44 pages of text, 6 chapters, 13 figures, 18 tables.

Annotatsioon

Neuroloogiliste häirete hindamise teste, nagu kella joonistamise testi, on neuroloogiaspetsialistid juba aastaid kasutanud dementsuse tunnuste märkamiseks ja selle arengutaseme hindamiseks, kuid seda tehti alati käsitsi. Hiljutised tehnoloogilised läbimurded võimaldavad neuroloogiaspetsialistidel selliseid katseid teha tahvelarvutites, laiendades võimalusi testiparameetrite kiiremaks arvutamiseks ja analüüsimiseks, avastades varem inimsilmale märkamatuid mustreid.

Kella joonistamise testi analüüs tehnoloogiliste meetmetega on olnu akadeemilises ringkonnas üsna levinud uuringuteema. Varasemad tööd käesolevale teemale annavad põhjaliku ülevaate kella joonistamise testi kvantitatiivsest ja semantilisest taustast, tutvustades samal aja erinevaid lahendusi testelementide eraldamiseks. Tänu kiirete tehnoloogiliste muutustele ja uute andmete kättesaadavusele on käesoleva töö eesmärgiks saanud varasema lähenemise kaasaegsete tehnikate ja täiendava kinemaatiliste parameetrite analüüsiga parendamine.

Käesoleva lõputöö esmane eesmärk on analüüsida kella joonistamise testis leiduvaid semantilisi ja kinemaatilisi tunnuseid ning, vastavalt avastustele, leida ja kirjeldada parameetre, mis võimaldavad eristada terveid inimesi Parkinsoni tõvega diagnoositud isikutest ja omakorda eristada viimaseid kerge kognitiivse häirega patsientidest.

Põhieesmärgi saavutamiseks on vajalik luua testielementide klassifitseerimissüsteemi. Testielemendi väljavõtmiseks edasise analüüsi jaoks on kasutusele võetud täiustatud süvaõppe, sealhulgas arvutinägemise, ja klassikaliste masinõppevõtete kombinatsiooni. Lõpptulemus on hinnatud klassifikaatorite toimivuse põhjal. Klassifikaatorid on treenitud suurima diskrimineerimise jõuga parameetrite peal ning valideeritud kasutades ristviitamise meetodikat.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 44 leheküljel, 6 peatükki, 13 joonist, 18 tabelit.

List of abbreviations and terms

MNIST	Modified National Institute of Standards and Technology, large data set of handwritten digits
YOLO	You Only Look Once algorithm
DBSCAN	Density-Based Spatial Clustering of Applications with Noise algorithm
PD	Parkinson's Disease
MCI	Mild Cognitive Impairment
CV	Computer Vision
CNN	Convolutional Neural Network
EM	Expectation–maximization algorithm

Table of Contents

List of Figures

List of Tables

1. Introduction

Dementia is a term used to describe a set of symptoms that cover a wide range of medical conditions caused by significant loss in brain function. The deterioration in cognitive abilities affects memory, thinking, comprehension, decision making and judgement – crucial aspects of our daily life. Alzheimer's disease is the most common form of dementia and contributes to around 60-70% of cases [1], while other types of dementia are distributed among the rest 30-40%.

Parkinson's disease is another neurodegenerative condition. Unlike Alzheimer's, Parkinson's disease mainly affects the motor system; however other dementia symptoms, such as slowness in thinking, memory loss and decrease in attention[2], can also occur in later stages of the disease.

Although there is no treatment currently available to cure such conditions, early diagnosis is proven to ease the progressive development of the disease by providing better care and condition management to patients and their caretakers.

For many decades, the main tools for screening and spotting signs of neurodegenerative disorders were pen and paper. Various tests - including Clock Drawing Test - were invented for analyzing the fine motor skills of a person and their understanding of the task. Tasks may vary, but most require the patient to draw a figure, a line or write a sentence. The tests are manually analyzed according to a scoring system and likely to base on a subjective judgement of the clinician.

Nowadays, the use of tablet computers and digital pens made it possible to conduct the assessments digitally, allowing neurology specialists to detect kinematic features and semantic elements, track the movement and discover patterns previously unnoticed by a human eye.

The focus of the present thesis is to develop a system for analysis of digitized Clock Drawing Tests, completed by the healthy control group, patients with mild cognitive impairment and individuals diagnosed with Parkinson's disease. The first sub-problem of this work is to recognize all drawing elements by applying modern deep learning algorithms with classic machine learning. The second sub-problem is semantic and kinematic features' extraction and analysis. The desired outcome is to distinguish one group of individuals from another. Feature analysis should provide an overview of differentiation between the groups and ultimately answer the main questions: can we separate one group of dementia patients from another; how vast is the difference between dementia and healthy control groups, based on a set of computed features.

1.1 Background

1.1.1 The Clock Drawing Test

The Clock Drawing Test (CDT) is a tool for cognitive impairment and dementia screening, measure and diagnosis. It requires the subject to draw a clock face with numbers and clock hands placed inside the contour. In some CDT cases, individuals are asked to draw a clock face from memory, completing a specific task along the way. For instance, they can be asked to a draw a clock indicating a certain time. Other tests might come with a pre-drawn contour.

Completion of the test requires the patient's attention, comprehension, memory and visual-spatial knowledge [3]. Since dementia affects precisely the aspects of cognitive abilities described previously, the CDT is considered successful in assessing dementia.

The CDT can be visually separated into three distinct groups of elements:

- Circle - contour of the clock face
- Digits - numeric representation of time
- Hands - time indicators

In order to assess the CDT, neurology and psychology specialists use scoring systems of different levels of complexity and thoroughness.

1.1.2 Semantic Features in the Context of the Current Work

The semantic feature is an aspect of a unit that describes and provides meaning to it. Although the term originates from the field of linguistics, the present thesis uses it to describe an aspect of every element found in a clock drawing. Under the circumstance that CDT can form groups of elements by their meaning to the evaluation and the clock face in general, the use of the term is appropriate in the context of the current research. To provide a "meaning" to elements of the CDT, it has been has decided

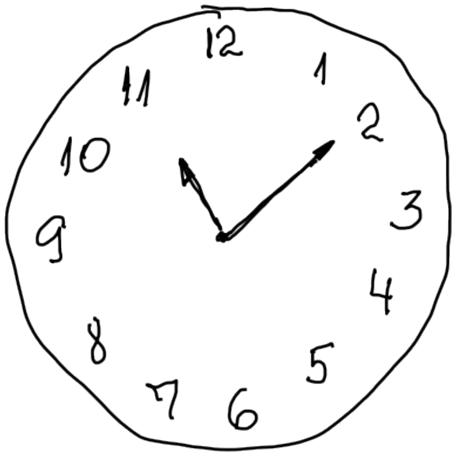


Figure 1. *Example of a Clock Drawing Test with all elements present.*

Figure 2. *Example of a Clock Drawing Test drawn by an MCI patient.*

to define the aforementioned features based on Freund's scoring scale.

Freund's Scoring Scale

Freund's scoring scale is a 7 points system for CDT assessment. This system is a good fit for the problem of defining semantic features since it focuses on three major categories: numbers, spacing and time setting. Semantic features of the elements are assigned and calculated according to their meaning and role in the evaluation process.

Table 1. *Freund CDT Scoring Scale[4]*

Time (3 points)	One hand points 2 (or symbol representative of 2) Exactly two hands Absence of intrusive marks, e.g., writing or hands indicating incorrect time, hand points to number 10; tic marks, time written in text (11:10; ten after eleven)
Numbers (2 points)	Numbers are inside the clock circle All numbers 1–12 are present, no duplicates or omissions
Space (2 points)	Numbers spaced equally or nearly equally from each other Numbers spaced equally or nearly equally from the edge of the circle

1.2 Problem Statement

Until recently, the CDT was used as a manual tool for screening and spotting signs of dementia. Nowadays, technological breakthroughs allow neurology and psychology specialists to conduct various tests on tablet computers, expanding the possibilities to compute and analyze test parameters faster and with higher accuracy.

Systems that would recognize different components of the drawing and automatically evaluate the result of a CDT are not available for public use just yet. A system developed by Ilja Mašarov for his master's thesis in 2017 [5] is considered a pilot version of what can potentially solve the problem stated above. The author was able to successfully implement a digitized version of CDT and extract its elements. Since the time from the initial development has passed, the system needs an update.

The main goal of the present research is to describe the differences in kinematic and semantic parameters describing the CDT process between the groups of PD patients, subjects affected by dementia and similar by age and gender distribution group of healthy controls subjects. Achieving the main goal requires solving the following sub-problems:

1. Extract and recognize different elements (digits, hands, and circle) from the drawn test;
2. Extract semantic and kinematic features describing each element;
3. Perform statistical hypothesis testing to answer the question of distinguishability;

4. Among the distinguishable parameters, select those possessing the highest discriminating power to distinguish between different groups;
5. Evaluate the performance of simple classifiers.

For the first sub-problem, goodness is evaluated if at least one element is recognized and extracted. Sub-problem 2 is the set of numeric calculations that do not require to be validated. For sub-problem three, let the level of significance be $\alpha = 0.05$, which is usual for similar types of problems. Considering the sample size is relatively small, only two or three parameters possessing the highest discriminating power should be selected for each classifier. Finally, the last sub-problem goodness threshold is set to be 0.7 for the accuracy, precision, recall and f1 scores. This value is most commonly reported in the literature.

A complete description of the approach and methods can be found in Methodology and Data section.

1.3 Workflow

The workflow of the current work if separated into two major parts.

The first part of the research is the classification of the CDT elements. The classification is achieved by a combination of advanced deep learning and classical machine learning methods (Table 3, DL methods are marked with a blue color, ML is marked with a yellow color). The analysis of the clock drawing test elements for correct diagnosis of a cognitive impairment has been described in the earlier research [5], [6], [7]. The uniqueness of the current work lies in implementation of the deep learning methods for classification of the elements.

The second part of the research is the analysis of classified CDT elements by their semantic and kinematic features. The main goal of the feature analysis is to explore and define the significance of the difference between the subject groups. The discovery of features with high discrimination power makes it possible to build high-performing classifiers for predicting the diagnosis - or its absence - by the digitized CDT sample.

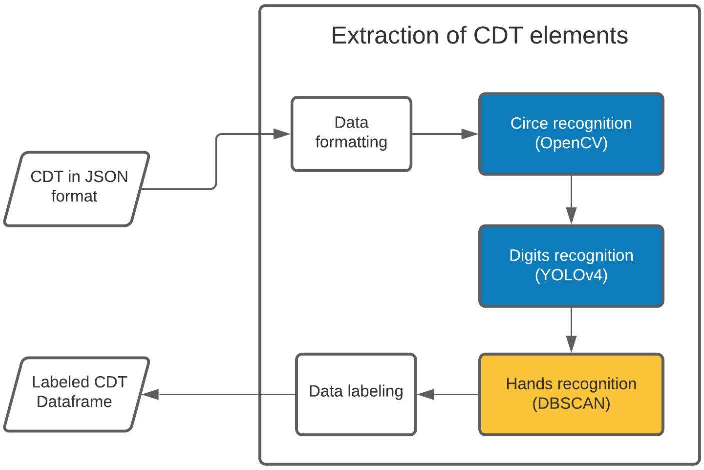


Figure 3. *Workflow: Extraction and Classification of the CDT elements.*

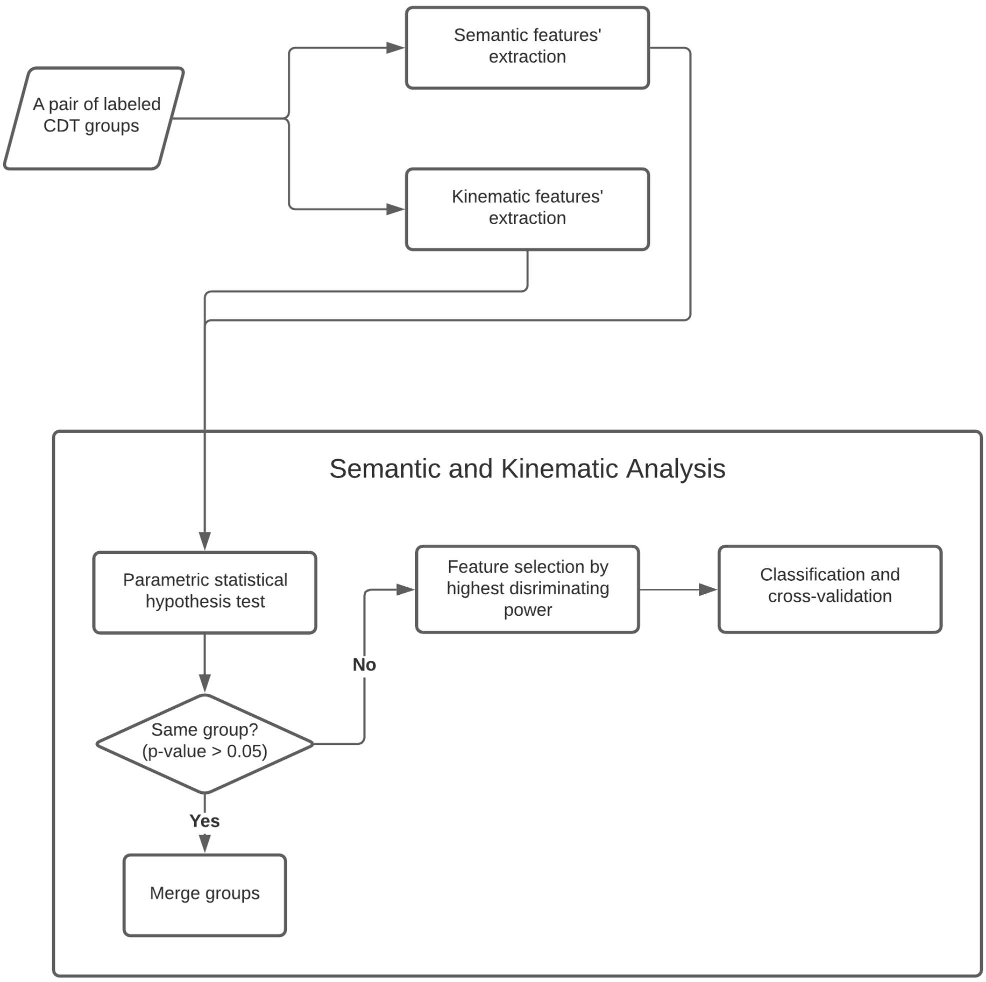


Figure 4. *Workflow: Semantic and Kinematic Analysis*

2. Related Works

2.1 Digital Clock Drawing Test Implementation and Analysis

The master's thesis "Digital Clock Drawing Test Implementation and Analysis" [5], written by Ilja Mašarov in 2017, is the main inspiration for the current research. The author implemented a digitized version of the clock drawing test and a pilot version of the CDT elements' classification system.

The approach to classify elements of the test is based on the motion-mass parameters of the strokes, for the exception of the digits, which were classified by a convolutional neural network. The current work is based on the combination of newer deep learning and classical machine learning methods for stroke classification.

2.2 Analysis of Interpretable Anomalies and Kinematic Parameters in Luria's Alternating Series Tests for Parkinson's Disease Modeling

The master's thesis "Analysis of Interpretable Anomalies and Kinematic Parameters in Luria's Alternating Series Tests for Parkinson's Disease Modeling" [8], written by Konstanti Bardõš in 2015, is used as a basis for the extraction of kinematic features. The original analysis in the work was build on Luria's alternating series test. In the current work, a similar approach is based on the clock drawing test.

3. Data

The current chapter will introduce the data that was used for the research and analysis.

3.1 Digitized CDT

The present subsection answers the question of what is a digitized clock drawing test and what set of data was used for the research.

A digitized CDT is a JSON file that contains an array of strokes drawn on a tablet computer with a digital pen. In essence, the stroke is represented as a vector of points $[p_1, p_2, ..., p_{n-1}, p_n]$, where n is the number of points in the stroke. Each stroke starts when a pen touches the surface of a tablet and ends when it detaches from the screen. The position on the screen (x and y, in Cartesian system), pressure, azimuth and altitude angles of the pen and a timestamp is described for each point in the vector. The pressure is estimated as the force of the touch, where 1.0 is the force of an average touch [5].

For this research, two distinct sets of CDT were subjected for the analysis: one set is gathered from the healthy controls group and PD patients from Estonia, the other - from the healthy controls group and MCI patients from Spain.

3.1.1 Addressing Differences in Data

It is necessary to point out that both data were collected independently, using different tablet computers and, essentially, data acquisition software. Nevertheless,

Table 2. *A sample from the unified CDT dataframe*

x	altitude	azimuth	y	pressure	timestamp	stroke_id
372.1719	0.872218	0.590648	-244.1602	0.333333	5.315888e+08	0
370.5156	0.872218	0.590648	-245.4961	0.333333	5.315888e+08	0
368.5938	0.872218	0.590648	-246.7500	0.333333	5.315888e+08	0
367.2656	0.872218	0.590648	-247.8164	0.333333	5.315888e+08	1
365.9375	0.872218	0.590648	-249.2695	0.333333	5.315888e+08	1

the difference between sets is insignificant and limited to the JSON structure and naming convention of stroke parameters.

4. Methodology

In the present chapter, an introduction and description of the methodology and tools used to solve the thesis can be found.

4.1 Tools

The development language of the current research is **Python** due to the wide range of machine learning and data science libraries available for and compatible with this language. Additionally, most of analysis was done in **Jupyter** environment, which is an interactive computational notebook for Python.

All the plots were made with **Matplotlib** library; **NumPy** and **Pandas** libraries were used for most of the computations, data handling and maintenance. **Scikit-learn** library was used for classical machine learning algorithms. For object detection and recognition from an image, **OpenCV** library and **YOLOv4** algorithms were used, the latter being trained on **MNIST** handwritten-digits dataset, gathered with the help of the **Keras** API module from **Tensorflow** library.

4.2 Classification of CDT Elements

In order to analyze the drawing, all elements need to be separated from each other. The workflow consists of three major components whereas these components constitute the combination of classical machine learning techniques and advanced deep learning methods.

First step is the separation of the stroke - or strokes - that form a circle from the initial drawing. Second step is the separation of the digits. Both methods use advanced computer vision and deep learning techniques. As for the remaining strokes, a classical data clustering algorithm was applied to identify potential strokes that form clock hands and separate them from the remaining strokes - outliers. For the time being, outliers and other miscellaneous elements that could be found in the process are neglected, since analysis of such elements is outside of the scope of current research. The final result of the process it to label all elements of interest: circle points are labeled as "0", hands as "20" and digits labels lie withing the range from "1" to "12".

4.2.1 Object Recognition with OpenCV

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library [9]. OpenCV contains a comprehensive set of classic and state-of-the-art machine learning and computer vision algorithms that can be used to perform various task including, but not limited to, face detection and recognition, and identification of objects and shapes.

OpenCV works solely with images and video frames. The colors of the image have to be converted from the blue-green-red color space to gray, as it is necessary for the detection algorithm to correctly identify contours - areas of pixels with similar color and intensity. Each notable change in color or density implies that a new contour area has started. In order to find contours, however, the Canny edge detector has been applied as it helps to mark separate objects by making their border of the same white color and intensity, while keeping the background black as in Figure 7a.

4.2.2 Object Detection with YOLOv4

You Only Look Once (YOLO) is a real-time object detection system [10]. Object detection in YOLO happens by applying bounding boxes and probabilities around the objects using a Bounding Box regression [11] function. The bounding box wraps around the object of a pretrained class (or a cluster) and returns center points coordinates, width and height.

YOLO v4 is built on a single neural network applied to a full image that divides it into smaller regions by compressing features down through a CNN backbone. *CSPDarknet53* is the backbone for YOLO v4 and contains 29 convolutional layers 3x3, a 725x725 receptive field and 27.6M parameters [11]. YOLO is a one-stage object detector, meaning it predicts the object localization and classification at once.

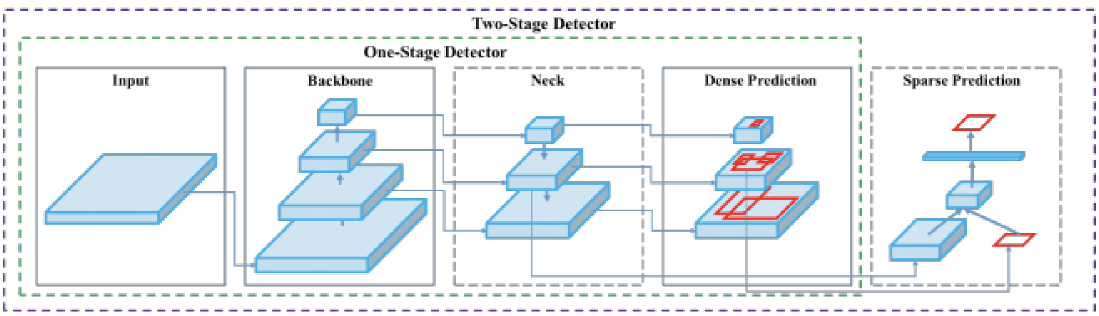


Figure 5. *Structure of an object detector[11]*

It is possible to train YOLO on a set of custom images to detect any object of choice. In case with YOLO, the backbone network's pretrained weights are adapted to identify relevant features out-of-the-box. In the process of training on custom data, they will be adjusted according to the features of new images and object detection settings.

Training Data Preparation

The custom detection requires some additional preparation of the data that YOLO will be trained on. Firstly, all the images have to be stored in one place. Secondly, there must be a label file that contains information about bounding boxes and classes related to the ground truth objects for every image. Each row is assigned to one object, and has the following structure: *<object-class> <x> <y> <width> <height>*. Positional arguments x and y of a bounding box predicted by YOLO can be described as a normalized relative position between first and last pixels, while width and height are calculated as proportion. The information on the location of image sets, the number of classes in total and their name is stored in the configuration file that is being read upon the start of training process.

4.2.3 Clustering with DBSCAN

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is a clustering algorithm that uses the individual data points in dense regions as building blocks after classifying them by their density [12]. DBSCAN method requires only information about the density τ and radius ϵ of the dense region. The density of a point is defined by the amount of points that remain inside of the ϵ of that point. The densities are used to distinguish *border*, *core* and *noise* points. The *core* has to contain at least τ points within the ϵ, while *border* need at least one *core* point to lie within its ϵ. Together they form a whole cluster.

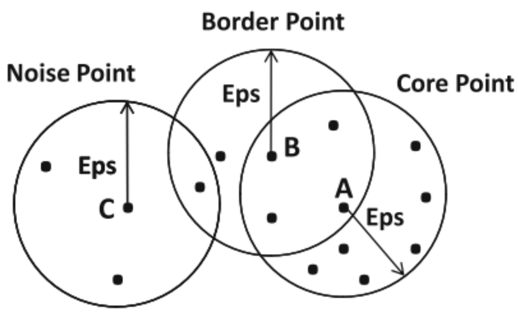


Figure 6. *Border, core and noise points[12]*

Major advantage of using DBSCAN algorithm for the current research is its ability to discover clusters of arbitrary shapes.

4.3 Analysis of CDT Distinguishability

The labeling of the CDT elements allows us to proceed with the analysis of semantic and kinematic features. Once features are extracted, classifiers should be build and trained on the attributes with the most discriminatory power. The classifier must be able to correctly distinguish the HC, PD and MCI groups by previously described parameters.

4.3.1 Identifying Parameters

Parametric Statistical Hypothesis Tests: Student's t-test

To quantify the difference between the means of two or more samples of data, a parametric statistical hypothesis test needs to be applied. The Student's t-test, one of such tests, determines whether the means of two independent samples are differ significantly. The hypotheses of t-test can be described by two hypothesis statements:

- H_0: the means of the samples are equal
- H_1: the means of the samples are not equal

The test results can be described through a calculated p-value. The p-value can be interpreted as the probability of observing the two data samples under null-hypothesis. If the p-value is below the significance level α (0.05), then the null-hypothesis is rejected, and the alternative hypothesis is accepted.

Fisher's Score

The Fisher score is designed to measure the ratio of the average interclass separation to the average intraclass separation between attributes [12]. The ratio is described with the following equation,

$$F = \frac{\sum_{i=1}^{k} p_i(\mu_i - \mu)^2}{\sum_{i=1}^{k} p_i \sigma_i^2} \tag{4.1}$$

where μ_i and σ_i are the mean and standard deviation of data points belonging to class i for a single feature, p_i is the subset of data points of class i, and μ is the global mean of the data on the evaluated feature. The greater the discriminatory power of the attribute, the higher is its Fisher's score.

4.3.2 Classifiers

K-Nearest Neighbors

K-nearest neighbor (KNN) is a supervised machine learning algorithm for classification. It operates by memorizing the training dataset and is considered fairly straightforward. KNN requires the knowledge about the number of k neighboring points and a distance metric. Then, it finds the k-nearest neighbors of the sample and assigns the class label according to the prevalent class among the neighbours.

Decision Tree

Decision Tree is another supervised machine learning classification and regression algorithm. The tree-like model predicts the class label of a sample by making a decision based on simple decision rules, generated by breaking the training data down into smaller subsets.

Random Forest

Random forests is defined as an ensemble of decision trees. Each individual tree in the model makes a prediction of a class label and the final prediction depends on the prevalence of votes. The class with the most votes is considered the model's prediction. The randomness has explicitly been inserted into the model building process of each decision tree [12], creating a low correlation between them.

Logistic Regression

Logistic Regression is a simple algorithm for preforming linear and binary classifications. Logistic regression can be viewed as a linear classifier, meaning that a linear hyperplane is used to separate the classes [12]. It preforms via the logistic function that exponentiates the distances between classes and a plane, defining the class localization.

Cross-Validation

The cross-validation is a method used for model evaluation by providing estimates of the model's performance on an unseen data. To perform cross-validation, the original dataset has to be split into training, validation and test sets. In the case of the K-fold cross-validation resampling technique, the set is split into k sets, where k - 1 folds are used for training and 1 for performance evaluation; the process is repeated until k models are evaluated and a performance estimates are generated.

5. Implementation

The current section describes the actual application of methods described in previous chapter and final implementation of the thesis solution. The classification of the CDT elements is achieved by the combination of deep learning and classical machine learning techniques, which is the defining characteristic of the present research when compared with other related works. The practical reason behind this approach is described further in the section.

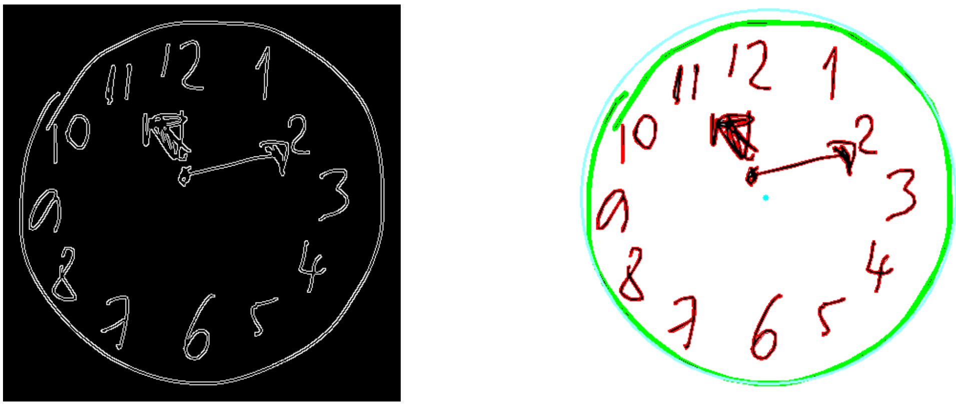
5.1 Circle Classification

As it has been implied earlier in the research, the detection of a circle in CDT happens by applying a computer vision algorithm for contour detection. The first step in the process of finding the circle is converting the JSON file with CDT data to an image file, as OpenCV only works with images. The image file is obtained from the plot as its parameters are useful to compare resolutions and translate images back to JSON.

Once the data is converted into a compatible format, the contour detection process can be started. Applying Canny edge detector to the grayscale image of a CDT will generate a set of all possible contours, however, there is only one circle contour that needs to be extracted.

Under the assumption that circle contour has the biggest area compared to the other elements of a drawing, it is reasonable to set an additional area threshold to ignore smaller items. Once the contour of interest is found, the minimum enclosing circle function is applied. It finds a circle that covers the object with minimum area and returns its central points (x, y) and a radius. These parameters are used to separate the corresponding points from the original CDT data by comparing closeness of each stroke center and length to the same parameters found in a contour.

The final result is visualized in the Figure 3b, where the circle contour is marked with the green colour, and contours that are left for further analysis are marked with red. The minimum enclosing circle and its center are painted with cyan colour. Separating the circle element from the CDT and labeling all stroke points as "0" allows us to proceed to the digit detection.



(a) *Canny edge detector output* (b) *Final output*

Figure 7. *Circle recognition process*

5.2 Digit Classification

For the current research, it has been decided to use YOLO for digit detection and localization. Although YOLO comes with an impressive set of pre-trained weights and is able to detect some objects upon the very first run, it is not possible to detect digits without training it on a custom dataset first.

5.2.1 Preparing Custom MNIST Dataset

The digit detection has been realized through training and tesing YOLO algorithm on the MNIST handwritten digits dataset. MNIST contains 60000 examples of training and 10000 of testing sets [13], respectively. Each example is a 28x28 image of a handwritten digit. This resolution, however, appears to be too narrow for YOLO to be trained. Training image's resolution should be at least 320x320 up to 608x608 with a step of 32 pixels. Therefore, the dataset has been modified according to the requirements, keeping the image resolution at 416x416.

It has been decided to train YOLO to detect 12 classes, a sequence of digits from 1 to 12, instead of 10, as there are single digits in total, from 0 to 9. This approach allows us to neglect additional checks of closeness of two digits to form a two-digit CDT class that would have been inevitable in the case of recognizing just one digit at a time. Since all the numbers in MNIST are single-digit, two-digit classes "10", "11" and "12" were generated from the combination of "0", "1" and "2" MNIST examples. The generated training data consists of chaotically distributed handwritten digits examples. A new position of digits on the training image is calculated upon every generation.

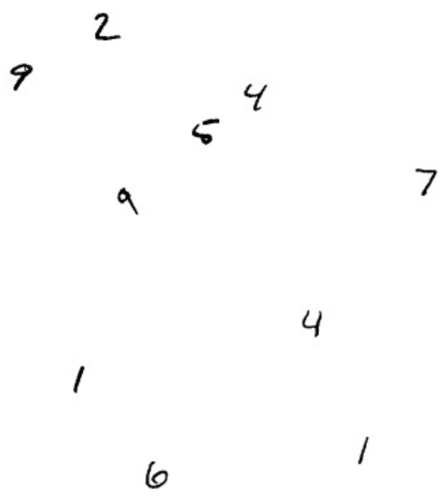


Figure 8. *YOLO compatible MNIST dataset*

```
6 0.78125 0.298077 0.067308 0.067308
1 0.211538 0.033654 0.067308 0.067308
0 0.668269 0.778846 0.067308 0.067308
3 0.579327 0.552885 0.067308 0.067308
0 0.165865 0.651442 0.067308 0.067308
3 0.478365 0.153846 0.067308 0.067308
8 0.25 0.334135 0.067308 0.067308
4 0.389423 0.216346 0.067308 0.067308
8 0.064904 0.112981 0.067308 0.067308
5 0.302885 0.824519 0.067308 0.067308
```

Figure 9. *YOLO darknet label file for MNIST object*

The final training dataset consists of 15816 generated data files, half of which is images (Figure 8) and the other half is the label files for ground truth objects (Figure 9). The central points, width and height of each bounding box are relative to the training image resolution. The width and height of single-digit classes are the same as in the original MNIST sample, while positional coordinates are generated at random and positioned without overlapping one another. The two-digit classes are generated similarly except for the bounding box being 1.5 times wider.

5.2.2 Training Process and Detection Results

According to the YOLO documentation, 2000 iterations for each object is sufficient enough to stop the YOLO training process. In the best interest of the research, after each 1000 iterations a weight file has been saved and tested. The best detection results were established after 10000 training iterations.

Although YOLO performance in identifying an actual digit object is significant, often with nearly 100% prediction accuracy, it is important to acknowledge that some of the detected classes did not resemble an actual digit object, despite being recognized as such. For instance, YOLO would label a smaller area in a clock hand as an object of class "1" (Figure 10). This has lead to applying an additional rule that checks whether all points of the stroke fit the bounding box or not. Another restriction is the exclusiveness of the element - each digit class can appear in the drawing only once. The rule is implemented by considering the element with the highest prediction confidence, if the same class label is predicted more that once on the same sample.

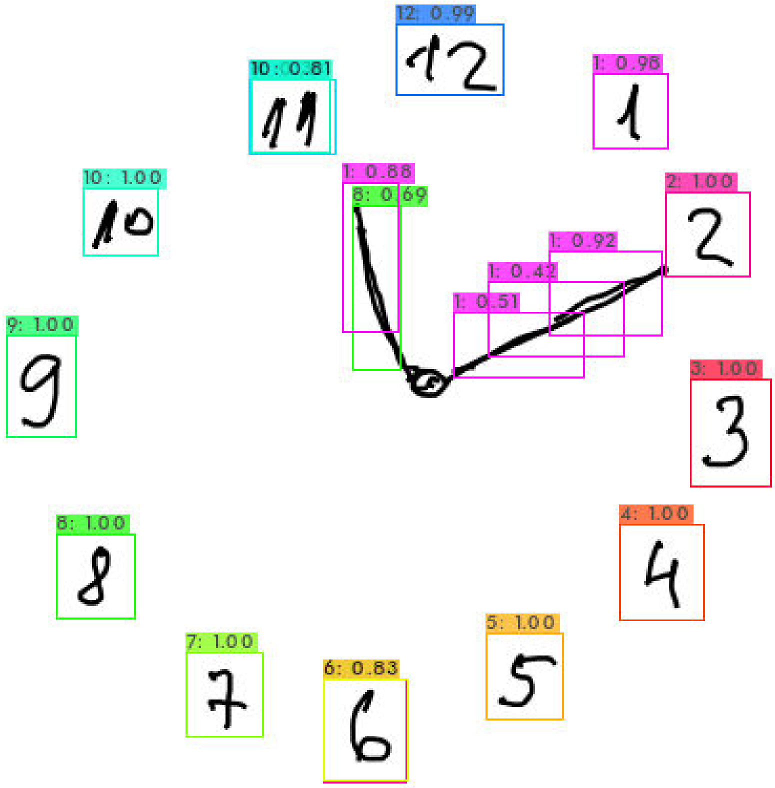


Figure 10. *YOLO predicted Digit classes*

In order to assign a correct digit-class label to the CDT strokes, bounding box parameters need to be converted into the Cartesian coordinate system, corresponding to original CDT metrics. Since the predicted parameters are relative to the image resolution, the conversion is done by multiplying the width and height by axes lengths and finding positional parameters by adding x and y offsets to the starting point of the corresponding axis.

The strokes are labeled as digits only under condition where (x_{min}, y_{min}) and (x_{max}, y_{max}) points of the stroke lie within the predicted bounding box area (Figure 11). Finally, when all potential digit strokes are labeled according to the YOLO predicted bounding boxes, clock hands, if they are present, should be labeled as well.

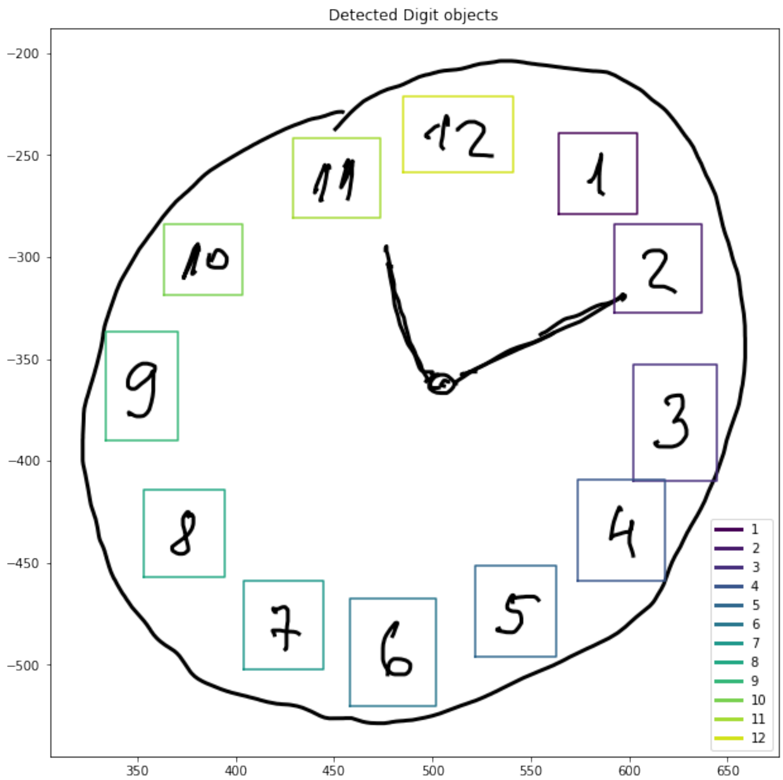


Figure 11. *Final Digit classes in the original CDT space*

5.3 Clock Hands Classification

The circle identification with a computer vision algorithm has been quite simple because of the triviality of the shape, and digit detection has been efficient due to the accessibility of MNIST handwritten digits dataset for training a CNN. However, it is not the case with the classification of the clock hands' element.

There are plenty of possibilities to draw clock hands and the absence of an extensive data with exclusive examples makes it the most difficult CDT element to classify using deep learning methods. An attempt to train YOLO on a custom dataset of hands, drawn on an iPad resulted in very poor detection performance, hence was neglected. It has been decided to approach the problem of hands' classification with classical machine learning methods.

The DBSCAN algorithm has been chosen to preform the final classification due to its capability in discovering arbitrary shapes. The unclassified strokes are run through the DBSCAN model from **scikit-learn** library with a relatively high ϵ value and fairly low τ number of samples, as the hands shape is likely to be quite narrow.

As a result of performing DBSCAN clustering on the remaining unlabeled stroke

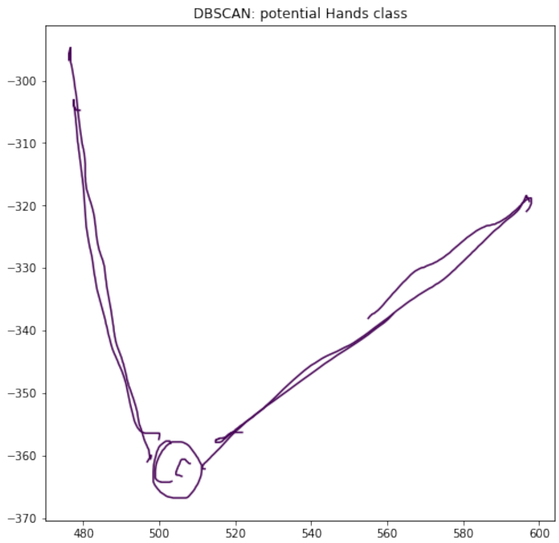


Figure 12. *Potential Hands cluster found with DBSCAN*

points, an array of corresponding labels is returned. In case of a correctly drawn clock, only one label must be assigned (Figure 12). In case if more than one unique label have been found, the closest cluster to the drawing's center by Euclidean distance is treated as the final clock hands object. The distance is described in the equation (5.1),

$$d(h, c) = \sqrt{(c_x - h_x)^2 + (c_y - h_y)^2} \tag{5.1}$$

where h and c are the central coordinates of a potential hand's cluster and the circle stroke, respectively.

The last step in the classification process is to remove all unlabeled points as they are not in the focus of the current research. At the end of the process, an additional *detect_ class* attribute of the original CDT data is filled with the corresponding values. The values are "0", "1" to "12" and "20", and stand for *circle*, *digit_ n* and *hands*, respectively. In the subsequent sections, the vector points $[c_1$, c_2, ... , $c_n - 1$, $c_n]$ of the CDT element, is referred to as ***class*** or ***element***.

5.4 Feature Setting

5.4.1 Semantic Features

The reason for analyzing the semantic features in the current research is the exploration of the distinguishability between healthy individuals and dementia patients by semantic features. If it occurs that the groups are distinguishable, the identification of discriminating power between the subject groups by semantic features has to be assessed as well. The semantic features are used in the subsequent classifier creation.

As it has been previously mentioned, some semantic features of a CDT are formed by following Freund's CDT scoring guidelines. The process can be though as defining features that can be used for an actual CDT evaluation per Freund. The process is different for each group of CDT elements.

In the following section, a semantic feature is explained through comparison of each element with the corresponding connotation in the Freund's scoring system. Some of the features are unrelated to the Freund system, but rather deducted from the interest in discovering patterns within different groups of tested individuals.

Circle Features

Exploring the Freund CDT scoring scale (Table 1), the following connection is found:

- Numbers spaced equally or nearly equally from the edge of the *circle*
- Numbers are inside the *clock circle*

A couple of conclusions regarding semantic meaning of a circle can be drawn from the excerpts above. Firstly, in order to state whether the numbers lie inside the circle or not, the radius and the center of a circle need to be found. Secondly, to evaluate the equality of the spacing between numbers and circle edge, identical set of values can be used. Provided this knowledge, the *radius*, *center_x*, *center_y* were extracted.

Additionally, the standard deviation of the circle coordinates have also been gathered, due to the significance of the coordinates deviation found specifically in the circle stroke. Although the standard deviation features are not connected to Freund's scale, it might positively impact the further analysis and classifier.

Table 3. *Description of semantic features found in a circle*

Circle Feature	Description
radius	radius of the circle element
center_x	x-coordinate of the circle's center
center_y	y-coordinate of the circle's center
std_x	standard deviation of the element by x
std_y	standard deviation of the element by y

Digit Features

Following the same procedure, the identified connection is:

- *Numbers* spaced equally or nearly equally from the edge of the circle
- *Numbers* spaced equally or nearly equally from each other
- *Numbers* are inside the clock circle

In the case of numbers - or digits -, the information about digit positions must be obtained. Moreover, the distance from the edge of the circle and from the neighboring class can be useful for building a well-performing classifier as it might detect patterns in spacing between the CDT elements in different groups of individuals.

Table 4. *Description of semantic features found in a digit*

Digit Feature	Description
radius	radius of the circle element
center_x	x-coordinate of the digit's center
center_y	y-coordinate of the digits's center
distance_from_edge	standard deviation of the element by x
distance_from_neighbor	standard deviation of the element by y

Hands Features

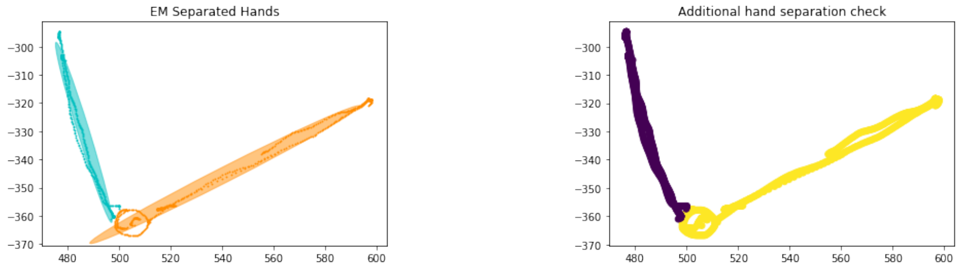
The process of finding semantic features in a clock hand is differs from the same process for the circle and digits. Mainly, it has occurred due to the importance of the hands in the evaluation per Freund scale.

- One *hand* points 2

- Exactly two *hands*
- *Hands* indicate correct time

An additional analysis of the hands class needs to be done in order to successfully proceed with the evaluation. The initial classification implied the hands class to be inseparable, hence the element could be analyzed as a whole cluster. The initial approach is not particularly suitable for the current route of the evaluation and semantic analysis. Hence, hands must be separated.

The hands separation is done by applying the Expectation-Maximization algorithm. It is a clustering algorithm that finds classes by estimating the missing samples, optimizing the model and repeating the process until full convergence of the data sample [14]. By setting the EM model to find two clusters of hands, we achieved the desired outcome of separating the hands.



(a) *Detected hand's clusters* (b) *Same-stroke rule applied*

Figure 13. *Clock hand's separation*

It is important to foresee that, although EM algorithm performs well in separating the n number of clusters, in some occasions only one clock hand is actually present on the CDT. The EM algorithm will do its best to create two-classed vector, resulting in a element that was not supposed to occur. Of course, it is considered a mistake by the scoring system, but even a single element is a very important bit of data for further analysis of the test groups.

To approach the problem of identifying the number of correctly separated hands, vector points' *stroke_id* attributes are set in comparison with the cluster labels of the same stroke. Provided the whole stroke has a unique label, if more that one label is assigned to the corresponding sample after prediction, then the prediction is considered inaccurate. In the case of inaccurately predicted labels, the statistical maximum label is assigned to the data sample. The final result is a set of labels with either one or two reoccurring values.

Once hands are processed, the following semantic features can be obtained (Table

5). The *rotation*, *max_x* and *max_y* features are calculated from the covariance between cluster points and the corresponding eigenvectors.

Table 5. *Description of semantic features found in a clock hand*

Digit Feature	Description
radius	biggest radius of the epsilon around the hand's cluster
center_x	x-coordinate of the hand's center
center_y	y-coordinate of the hand's center
max_x	x-coordinate of the hand's tip
max_y	y-coordinate of the hand's tip
rotation	rotation angle counter-clockwise
time_class	label of the digit, indicated by the hand (closest digit)
distance_from_digit	euclidean distance from the closest digit

5.4.2 Kinematic Features

Kinematic parameters of handwritten assets have proven to produce high level of discriminating power between Parkinson's disease patients and healthy subjects in the previous research [8]. In the favour of the current work, additional set of MCI patients is analysed alongside with the PD and HC group to support the statement of high distinguishability by kinematic features as well as potentially discover the significance between all subjected groups, especially interesting is significance between MCI and PD patients.

Kinematic feature set describes each vector of the CDT element separately. On the contrary to the semantic feature extraction process, additional separation steps are not required, as elements are treated equally, as a single arbitrary vector of points.

The higher-order kinematic parameters are *velocity*, *acceleration*, and *jerk*. For each high-order feature the *mean*, *mass* and *number of changes* are calculated and used for further analysis and classifiers. Additionally, the *duration*, *trajectory_length* and *pressure* features are being extracted to compare the current result with the previous research on the topic. The description of the kinematic features are inspired by K. Bardõš's master's thesis [8], where a similar problem was approached.

Table 6. *Description of kinematic features [8]*

Feature	Description
duration	time period between first and last point of the class
trajectory_length	sum of Euclidean distances between neighbouring points in stroke
velocity_mass	sum of all velocities in a class
velocity_mean	average velocity of a class
velocity_nc	number of velocity changes in a class
acceleration_mass	sum of all acceleration values in a class
acceleration_mean	average acceleration of a class
acceleration_nc	number of acceleration changes in a class
jerk_mass	sum all jerk values in a class
jerk_mean	average jerk of a class
jerk_nc	number of jerk changes in a class
pressure_diff_mean	average difference in pressure between neighbouring points
pressure_mass	sum of pressure values in a class
pressure_nc	number of pressure changes in a class

The process of features' selection is repeated for all available sets of CDT. In total, there are 7 HC and 17 PD samples from Estonia, and 12 HC and 11 MCI samples from Spain subjects.

5.5 Analysis of Features

The current section is dedicated to the description of feature analysis. The progress is similar for both semantic and kinematic features. The goal of the feature analysis is to examine the distinguishability between different groups of individuals. Sets of CDT samples are separated by the background of the test subjects.

5.5.1 Feature Extraction by Discriminating Power

The higher the discriminating power of a feature is, the better grounds it provides to a build classifier on.

There are multiple ways to assess distinguishability of the data. In the current research, it has been accomplished by calculating both p-value and Fisher's score of the independent samples.It is worth noting that the features selected by either of the metrics are ultimately the same, which is an expected outcome. For this reason, the main metric of reference will be p-value as it provides more clarity about the equality of samples. The two samples are thought to come from a different distribution if the p-value is below the significance level α (0.05).

The distinguishability is assessed based on opposing pairs of groups. The samples are compared by every CDT element separately, providing the context about the most distinguishable features and the elements they were found in. Each result table indicates what element is being analyzed. The indicator consists of the pair's group indicators and the CDT class label at the end. For instance, *pd_hc_0* means that the pair is formed from PD and HC samples, and the features belong to the *circle* element. Only 2-3 best features are selected for the classification model training.

5.5.2 Feature Extraction Results

HC vs HC

The HC group from Estonia and HC group from Spain were the first pair in the process, and for a good reason. Under the assumption that the HC groups are indistinguilshable, there is an urge to unite the two group into one, since it would have provided a bigger and more diverse set of samples for the subsequent analysis. However, there was no evidence the assumption is valid, therefore, before merging the groups, they have to be proved inseparable.

Obtaining the p-values from the Student's t-test proved that, while the minority of the elements did show some high discriminating level, overall it is safe to merge the groups into one, both by kinematic and semantic features. The example of p-values can be observed in Table 7. Values are sorted in ascending order.

feature	controls_1
jerk_mean	0.270795
jerk_mass	0.272199
acceleration_mass	0.272237

Table 7. *Example of low discriminating level (p-value)*

The subsequent reference to HC implies the merged group of Estonian and Spanish HC.

PD vs HC

Evaluation of the PD and HC groups supported the common knowledge that the CDT samples have high distinguishability by nearly every element of the drawing. The samples are kinematically most separable by the circle element, with selected features being *jerk_mass*, *velocity_mass* and *acceleration_mass*. Semantically, the circle element also differs significantly by *center_x* and *y_sdt* features.

feature	hc_pd_0
jerk_mass	6.94028e-05
velocity_mass	0.000228664
acceleration_mass	0.000276653

Table 8. *Kinematic features selected from PD/HC assessment*

feature	hc_pd_0
center_x	0.0390078
y_std	0.118333

Table 9. *Semantic features selected from PD/HC assessment*

MCI vs HC

Interestingly, the evaluation of distinguishability in MCI and HC features did not provide such dramatic results as in the previous pair. In fact, most of the p-values for the samples appeared to be high enough to assume that the data samples come from the same distribution. However, low p-value was found in the digit element of class "8". No significant semantic features were discovered for the pair.

feature	hc_mci_8
pressure_mass	0.0116476
duration	0.020174

Table 10. *Kinematic features selected from MCI/HC assessment*

PD vs MCI

It is already possible to make a conclusion about the significance of difference between PD and MCI pair from the previous results. The p-values for all of the elements indicated that the samples are unequal, implying the high distinguishability of the pair. According to the results, PD/MCI pair has the highest discriminating power out of all assessed pairs by kinematic features. However, by semantic features the pair is nearly inseparable, supporting the conclusion that, with the current approach, overall separability by semantic features is not as successful as it is by kinematic parameters.

feature	pd_mci_0
jerk_mass	5.56758e-06
velocity_mass	8.57844e-06
acceleration_mass	2.00937e-05

Table 11. *Kinematic features selected from PD/MCI assessment*

feature	pd_mci_9
distance_from_neighbor	0.0210383
center_y	0.0611824

Table 12. *Semantic features selected from PD/MCI assessment*

5.5.3 Classification Results

Following the feature extraction phase, it is now possible to build and train classifiers to predict the group label of the incoming sample. The classifiers are build by analogy with the feature extraction, pair-wise. The main classifiers of interest are build on the features that with highest difference perform quite well. Nevertherless, some models have shown better performance in another classes, trained on a different set of features.

The classifiers are evaluated by precision, recall and f-score values, as well as accuracy, generated during cross-validation of the model.

Classification by Kinematic Features

The most well-preforming classifiers by kinematic features are the following:

	precision	recall	f-score	accuracy	std
K-Nearest Neighbors	0.633333	0.625	0.619048	0.88	0.160000
Decision Tree	0.500000	0.500	0.466667	0.59	0.352704
Random Forest	0.633333	0.625	0.619048	0.75	0.232379
Logistic Regression	0.250000	0.500	0.333333	0.44	0.048990

Table 13. *Classifier build for differentiating PD and HC by kinematic features of a digit (class 4)*

	precision	recall	f-score	accuracy	std
K-Nearest Neighbors	0.866667	0.6	0.633333	0.766667	0.200000
Decision Tree	0.900000	0.8	0.819048	0.933333	0.133333
Random Forest	0.900000	0.8	0.819048	0.800000	0.163299
Logistic Regression	0.900000	0.8	0.819048	0.866667	0.163299

Table 14. *Classifier build for differentiating MCI and HC by kinematic features of a digit (class 8)*

	precision	recall	f-score	accuracy	std
K-Nearest Neighbors	0.925926	0.888889	0.895726	0.960000	0.08000
Decision Tree	0.925926	0.888889	0.895726	0.960000	0.08000
Random Forest	0.925926	0.888889	0.895726	0.960000	0.08000
Logistic Regression	0.604938	0.777778	0.680556	0.626667	0.03266

Table 15. *Classifier build for differentiating PD and MCI by kinematic features of a circle (class 0)*

Classification by Semantic Features

The most well-preforming classifiers by semantic features are the following:

	precision	recall	f-score	accuracy	std
K-Nearest Neighbors	0.687500	0.666667	0.657143	0.635714	0.150509
Decision Tree	0.757143	0.750000	0.748252	0.585714	0.171429
Random Forest	0.757143	0.750000	0.748252	0.610714	0.228236
Logistic Regression	0.611111	0.583333	0.555556	0.642857	0.255551

Table 16. *Classifier build for differentiating PD and HC by semantic features of a circle (class 0)*

	precision	recall	f-score	accuracy	std
K-Nearest Neighbors	0.904762	0.714286	0.757143	0.733333	0.161589
Decision Tree	0.904762	0.714286	0.757143	0.616667	0.323179
Random Forest	0.928571	0.857143	0.874459	0.733333	0.161589
Logistic Regression	0.904762	0.714286	0.757143	0.633333	0.113039

Table 17. *Classifier build for differentiating MCI and HC by semantic features of a digit (class 3)*

	precision	recall	f-score	accuracy	std
K-Nearest Neighbors	0.714286	0.714286	0.714286	0.83	0.235797
Decision Tree	0.510204	0.714286	0.595238	0.53	0.297658
Random Forest	0.510204	0.714286	0.595238	0.68	0.211187
Logistic Regression	0.510204	0.714286	0.595238	0.62	0.112250

Table 18. *Classifier build for differentiating PD and MCI by semantic features of a digit (class 4)*

According to the received results, most of the classifier models were able to reach the threshold of 0.7 accuracy rate for at least one element of the CDT, despite being trained on a relatively small set of data. However, there is a noticeable difference in results of the classifiers build on kinematic features and semantic features. The classifiers for differentiating the samples by kinematic features is performing significantly better. The reason for such difference might be that kinematic features are extracted and measured directly from the arbitrary vector points of the CDT element. In the kinematic analysis the measures that cannot be visually extracted from the test are taken into account, while semantic features are build on the visual comprehension of the drawing test.

Nevertheless, the results of the classification allow to state that, despite the small amount of available data, it is possible to build well-preforming classifiers for distinguishing a healthy subject from a person diagnosed with Parkinson's disease, a healthy individual from a person diagnosed with mild cognitive impairment, and a Parkinson's disease patient from a dementia patient.

6. Summary

The goal of the current research was to discover whether there is a distinguishability between the healthy subjects, Parkinson's disease patients and people diagnosed with mild cognitive impairment through the analysis of the semantic and kinematic features found in a digitized clock drawing test. In order to achieve the goal, a program to detect and classify the elements of a drawing was created using the combination of advanced deep learning and classical machine learning techniques.

In the first part of the research, the elements of the drawing needed to be classified. It has been discovered in the process of development that the current implementation is preforming very well on the visually clear examples of the drawings. However, there is a room for improving the cases of not so successful drawings. In the current research, plenty of elements were neglected by the detector, creating an obstacle for the further detection and classifying the visually present classes as outliers, which leads to decrease in the data subjected for analysis.

In the second part of the work, the classified elements of the test were analyzed by the kinematic and semantic features. The kinematic features were extracted directly from the clock elements and calculated according to the physical equations. The semantic features were defined for each group of the drawing independently, concluding the definition from Freund scale directive.

The results of the analysis concluded that the elements, found in the drawings from different groups, are distinguishable between one another both by kinematic and semantic features. The highest distinguishability is found between the Parkinson's disease and dementia patients, while dementia patients and health control subject are the least different. Although the discriminatory power is higher in kinematic features than in semantic features, the current research has opened new possibilities for further research of semantic feature extraction and analysis. Perhaps, a better feature definition and closer analysis of the semantics of a clock drawing test could make a greater difference in the future works.

Bibliography

[1] World Health Organization. *Dementia*. URL: https://www.who.int/news-room/fact-sheets/detail/dementia.

[2] National Parkinson Foundation. *Parkinson's Dementia*. URL: https://www.parkinson.org/sites/default/files/PD%20Dementia.pdf.

[3] Freedman MI et al. In: *Clock Drawing. A Neuropsychological Analysis*. Oxford: Oxford University Press, 1994.

[4] Barbara Freund et al. "Measures. Table 1". In: *Drawing Clocks and Driving Cars: Use of Brief Tests of Cognition to Screen Driving Competency in Older Adults*. Journal of General Internal Medicine, 2005.

[5] Ilja Mašarov. In: *Digital Clock Drawing Test Implementation and Analysis*. 2017.

[6] Mohamed Bennasara et al. "Feature Selection Based on Information Theory in the Clock Drawing Test". In: *17th International Conference in Knowledge Based and Intelligent Information and Engineering Systems - KES2013*. 2013, pp. 902–911. DOI: https://doi.org/10.1016/j.procs.2013.09.173.

[7] Sven Nõmm et al. "Interpretable Quantitative Description of the Digital Clock Drawing Test for Parkinson's Disease Modelling". In: *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*. 2018, pp. 1839–1844. DOI: 10.1109/ICARCV.2018.8581074.

[8] Konstantin Bardõš. In: *Analysis of Interpretable Anomalies and Kinematic Parameters in Luria's Alternating Series Tests for Parkinson's Disease Modeling*. 2015.

[9] *OpenCV*. URL: https://opencv.org/about/.

[10] *YOLO*. URL: https://pjreddie.com/darknet/yolo/.

[11] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. "YOLOv4: Optimal Speed and Accuracy of Object Detection". In: (2020). URL: https://arxiv.org/abs/2004.10934.

[12] Charu C. Aggarwal. In: *Data Mining: The Textbook*. Springer International Publishing, 2015.

[13] Yann LeCun and Corinna Cortes. *MNIST handwritten digit database*. 2010. URL: http://yann.lecun.com/exdb/mnist/.

[14] Jason Brownlee. *A Gentle Introduction to Expectation-Maximization (EM Algorithm)*. URL: https://machinelearningmastery.com/expectation-maximization-em-algorithm/.

Appendices

Appendix 1 - Lihtlitsens lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina, Jekaterina Viltšenko

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose "Kella joonistamise testi semantiliste ja kinemaatiliste parameetrite analüüs", mille juhendaja on Sven Nõmm
   (a) reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
   (b) üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

25.05.2021