Identifying movement patterns from large scale WiFi-based location data

-A case study of TU Delft Campus

Balazs Dukai
Simon Griffioen
Matthijs Bon
Martijn Vermeer
Xander den Duijn
Yuxuan Kang

TU Delft

1
Introduction

1.1. Intro

Wireless Local Area Networks (WLAN) are widely used for indoor positioning of mobile devices within this network. The use of the Wi -Fi network to estimate the location of people is an attractive approach, since Wi -Fi access points (AP) are often available in indoor environments. Furthermore, smart phones are becoming essential in daily life, making it convincing to track mobile devices. This provides a platform to track people by using Wi -Fi monitoring technology. Knowledge of people's locations and related routine activities are important for numerous activities, such as urban planning, emergency rescue and management of buildings.

To understand the human motion behaviour many studies are conducted based on data collection of GPS receivers. The Global Navigation Satellite System (GNSS) is commonly used to track people in large scale environments. However due to poor quality of received signals from satellites in urban or indoor environments, GNSS receivers are not suitable in these environments. This led to the development of alternative technologies to track people's locations, including Bluetooth, Dead Reckoning, Radio frequency identification (RFID), ultra-wideband (UWB) and WLAN (Mautz, 2012). WLAN has the advantage of widespread deployment, low cost and with the use of a smartphone as a receiver, the possibility to track a large amount of people.

In general, there are four different location tracking techniques by using the Wi -Fi network: Propagation modelling, multilateration, Fingerprinting and Cell of Origin (CoO). Many of these methods rely on Received Signal Strength Indicators (RSSI) and/or previous set of calibration measurements. In comparison, CoO is the most straightforward technique and snaps the location of the mobile device to the same coordinate position as the access point it is connected to. For this project, CoO is used to track people's movement.

At the Technical University of Delft (TU Delft) a large scale Wi -Fi network is deployed across all facilities covering the indoor space of the campus. The network is known as an international roaming service for users in educational environments and called the eduroam network. It allows students and staff members from one university to use the infrastructure throughout the campus for free. This allows for easy collection of Wi Fi logs including individual scans of mobile devices. A continuous collection of re-locations of devices to access points for a long duration will return detailed records of people's movement. This ubiquitous and individual history location data derived from smartphones will present valuable knowledge on movement on the campus. For this reason, the project is carried out in request of the University's department of Facility Management and Real Estate (FMRE).

In this project, WiFi monitoring technology is used to discover movement patterns on the campus of TU Delft. Based on the relationship between activities and places, location history can be used to discover significant places, movement patterns and hotspots. FMRE can use this information to answer questions such "what is the relation between buildings", "where do people come from" and "how regular a trajectory occurs". This project will present a method for identification of movement patterns in a large scale indoor environments and between buildings. The method uses concepts of sequential pattern mining. Previous research has been done on sequential pattern mining, such as Zhao et al. (2014) to discover people's life patterns from mobile Wi -Fi scans, Meneses Moreira (2012) analysed place connectivity using the eduroam network and Radaelli et al. (2013) identifies indoor movement patterns by analysing a sequence of relocations. Individual movement can be identified as a sequence of relocations of a mobile device to different APs. Without any data between two subsequent re-locations, sequential analysis is a convincing way for identifying moving patterns from wifilogs.

1.2. Purpose statement

Identifying movement patterns has attracted significant interest in recent years. This report will explain how movement patterns can be identified using large scale Wi -Fi based location data. This report tries to contribute with three proposes. [label=*0*)]

A method for identifying movement patterns by analysing individual sequences of relocations from a large scale Wi -Fi network; This includes filtering the raw data and automatically create individual trajectories over a time interval as a sequence of relocations;

Restructure the association rule mining algorithm to use it in a large scale tracking environment, to discover locations that are commonly associated;

Investigate different visualization methods for showing movement, based on a large scale Wi-Fi network.

A method for automatically detect what entrances are used to enter and exit a building.
The contributions can be described in one research question for this project.

– How can movement patterns be identified from large scale WiFi-based location data of the eduroam network?

In order to answer the research question well, there are some sub questions:

– What movement patterns can be identified between buildings on TU Delft campus?

– What movement patterns can be identified between large indoor regions on TU Delft campus?

– What entrances are used to enter and exit a building on TU Delft campus?

1.3. Methods

The Geomatics Synthesis Project (GSP) is a small research project that combines a literature study with practical research. This includes a case study of the TU Delft campus, using real-world data. Practical work includes data storing, processing, analysing, interpretation, visualization and validation. The project is carried out in a team of six students with a connection to a supervisor and stakeholders (FMRE). This involves interactive discussions between stakeholders as an important part of the research.

2
Top level requirements

To keep track of the progress of the project, it is necessary to monitor to which degree the project is meeting the top level requirements and if the project is still on schedule with these requirements. In the baseline review the requirements are specified using the MoSCoW rules and killer requirements. In this chapter these previous requirements will be discussed and possible changes will be explained.

The goals that *must* be achieved is on the level of detail of the campus. It's detailed specification as stated in the baseline review is shown below.

MUST campus level Main goal:

1. Identify which entrances are used to enter and exit a building;
2. Identify movement patterns and connectivity between building entrances by sequential pattern mining.

Relate entrances (place) of buildings to the corresponding APs (location). Find the stay places of each individual in order of the scan time. Find individual trajectories by taking a time interval from a sequence of stay places. Find the movement patterns, by deriving a sequence of common places shared by all trajectories. Visualize the movement patterns between buildings in static maps. A *killer requirement* for this level is: Identification of APs relating to an entrance of a building

Currently the project is progressed so far that it is possible to identify building patterns between buildings. The stay places of individuals and their trajectories have been found and this has been visualized in both static and dynamic maps and bar charts. But, until now there is no accurate map with the location of all access points of the campus. There is such a map for the faculty of architecture, but it is only one building and not very clear. Until this map of the whole campus becomes available, identifying entrances will be hard to do. Section 3.4 REFERENCE will go into greater detail about the progress that has been made so far with entrances.

The goals that *should* be achieved, focus on the building level, where buildings are divided into regions, but since there is currently no map with the locations of the access points, this level of detail is not yet reached. However, the way that the code is setup allows for easy transformation to higher levels of detail when such a map becomes available. How this code exactly works is explained in section 3.1 REFERENCE and 3.2 REFERENCE.

3
Progress

This document is intended to be both an example of the TU Delft LaTeX template for reports and theses, as well as a short introduction to its use. It is not intended to be a general introduction to LaTeX itself

Instructions on how to use this template under Windows and Linux, and which LaTeX packages are required, can be found in `README.txt`.

3.1. Movement between buildings

3.1.1. Bar chart

3.1.2. Maps

In order to get an overview about how people move on the campus and further more, find out movement patterns, a map visualization is essential. Map visualization consists of three parts:

1. base map: open street map is used as base map. There are many labels on open street map, providing more context of the environment, so it is more clear and readable compared to other base maps like satellite images.

2. building markers: building markers show the locations of the buildings. Google maps marker style is used since it is commonly used in many map application. Because the shape of the building is not useful in analyzing movement patterns between buildings, each building is regarded as a point instead of a polygon, thus a node in the network,

3. lines: lines are the most essential part in map visualization, they represent movements between buildings.

In the first stage of map visualization, only base map and lines are taken into consideration, building markers are not shown on the map. The line width represents the amount of movement and movements are aggregated daily regardless of the timestamp of each movement during a day. This map visualization gives an overview of the movements over a day and between which buildings there are the most movements. The following maps show the difference of the amount of movement between April 11th (weekday) and April 17th (weekend).

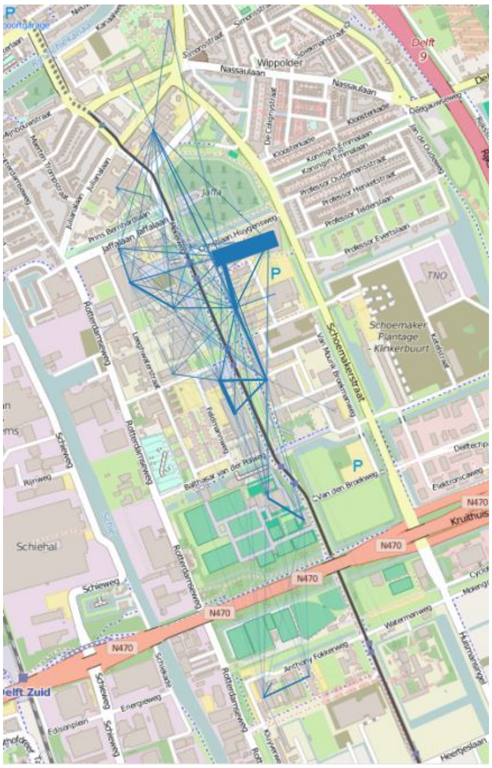

Figure 3.1: Movements of April 11th (week day)

Figure 3.2: Movements of April 17th (weekend)

It's clear that between Aula and library, there are the most movements and the amount of movements is totally different on weekday and on weekend.

Regarding the movements are dynamic and time is also highly related to movements, a dynamic map visualization is created to display individual movement over a day with temporal information. The following screenshots of the gif file show how the movements look like at a certain time of a day:

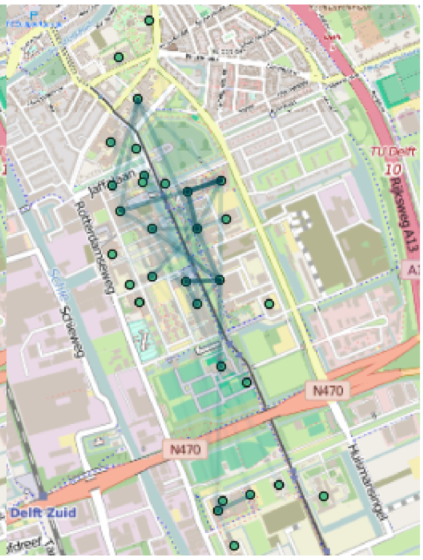

Figure 3.3: Movements of April 11th, 7:00 am

Figure 3.4: Movements of April 11th, 9:00 am

Figure 3.5: Movements of April 11th 11:00 am

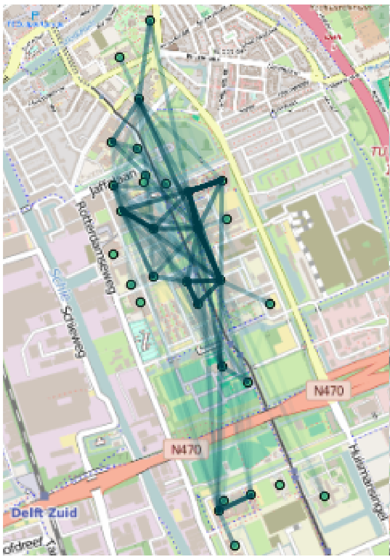

Figure 3.6: Movements of April 11th, 13:00 pm

Figure 3.7: Movements of April 11th, 15:00 pm

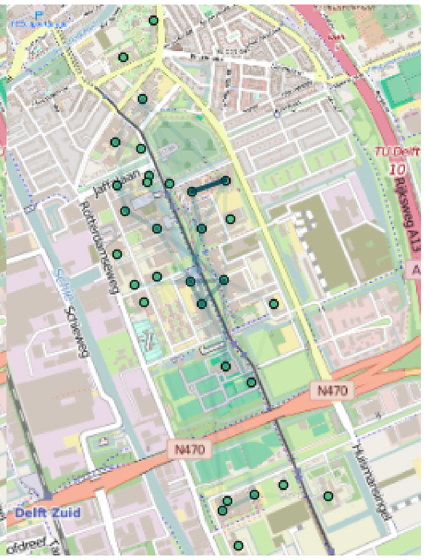

Figure 3.8: Movements of April 11th 16:30 pm

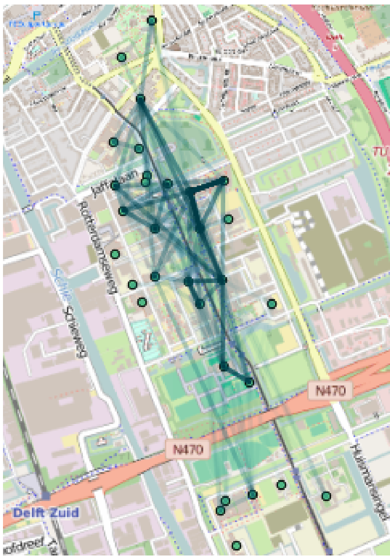

Figure 3.9: Movements of April 11th, 18:00 pm

Figure 3.10: Movements of April 11th, 20:00 pm

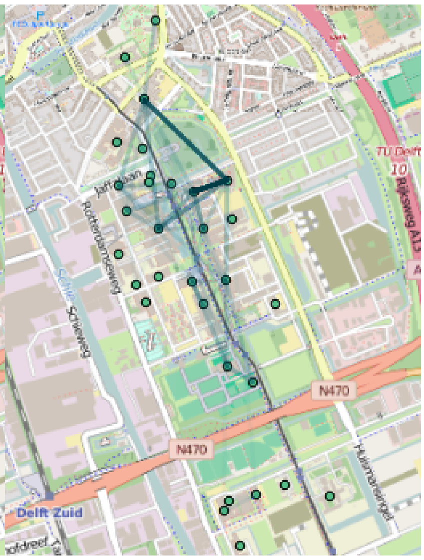

Figure 3.11: Movements of April 11th 22:00 pm

In these pictures, the more movements there are, the less transparent the lines are. So generally speaking, from 7:00 am to 20:00 pm, there are two peaks at 13:00 pm and 18:00 pm. Hence, it is possible to get some insights about movement patterns from the animation. However, the dynamic map visualization doesn't provide detailed information to dig into but only an overview. So in order to mine on movement patterns, it is necessary to create maps containing more information, including time, direction and so forth.

Because the amount of data is big, it is more convenient to generate maps automatically so that it will fasten the progress of finding movement patterns. According to the three components of map, there is some information needed to be collected before visualizing movement on map. The locations of buildings are collected manually on Google earth based the campus map. These locations are exported as KML file and imported into QGIS. After adding geometry columns x and y, the csv file is created and imported into database. By using $ST\_MakePoint$ function, a geometry column is created in database. In summary, the building locations are stored as the structure described in following table:

id	name	geometry	x	y
0	world			
3	science_ center	010100000042A7..	4.36939919846287	52.0072322181367
5	tnw_ bio	010100000043AE..	4.37120211221402	52.0086132164098
8	bk_ city	010100000077E3..	4.37053698152436	52.0056562098059
12	tnw_ dct	01010000007CA..	4.36891378927259	52.0040834950037
..	...	....	....	....

Table 3.1: Building data structure

There is a special 'building' called $world$ in the database. It is not an actual location, it is a virtual location which is used if someone is not scanned on the campus in a period of time. After storing the locations of buildings in the database, these locations will be extracted automatically from database to generate maps. There are two properties of lines used to deliver information:

1. width: line width is used to represent the amount of movements, but the amount is aggregated for both directions.
2. color: color is gradient from red to green. Red line means the movement is not symmetric that much more people move in one direction than the other, while green line means the movement is symmetric.

Based on this map visualization, users can choose certain dates and certain buildings to generate maps automatically. It makes it easier to find out movement patterns. Since not all buildings are chosen, the map will only display the movements between several buildings, which makes the map more readable:

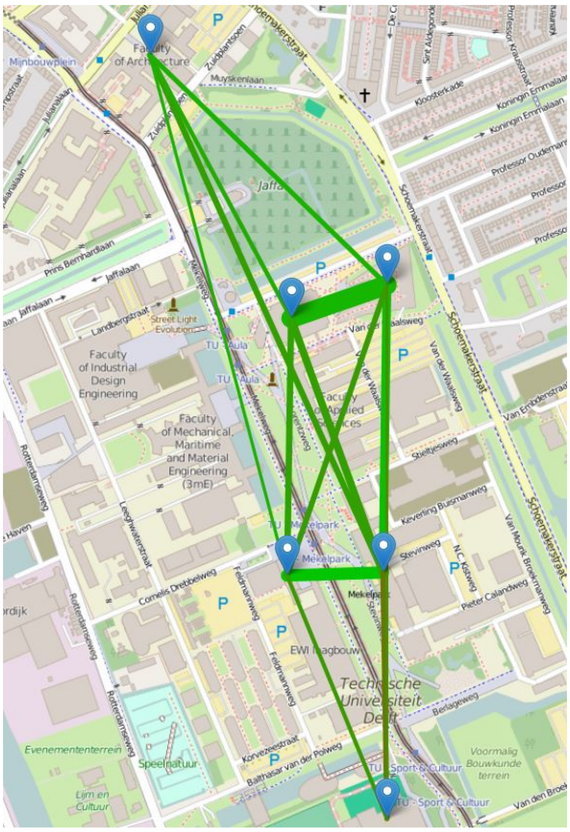

Figure 3.12: Amount of movements on April 25th

As shown in the map, the lines are in different colors, which shows the symmetry of the movements. If the user is willing to know more about the movement, it is also possible to click on the line to check the amount of the movements for each direction in detail, and there will be a pie chart showing how symmetric the movements are. With the map visualization, it is easy to focus on movements which are special or interesting:

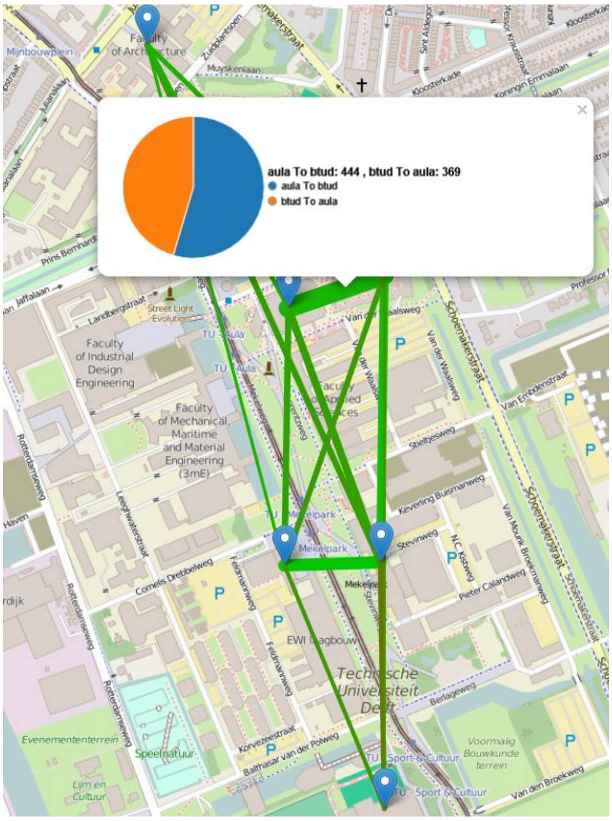

Figure 3.13: Amount of movements in pie chart

3.2. Sequences

This template will automatically generate a cover page if you issue the `\makecover` command. There are two formats for the cover page: one with a page-filling ('bleeding') illustration, with the title(s) and author(s) in large ultrathin typeface, and the other where the illustration fills the lower half of the A4, whereas title(s), author(s) and additional text are set in the standard sans-serif font on a plain background with a color chosen by the user. The last option is selected by the optional key `split`: `\makecover[split]` yields a page with the illustration on the lower half. All illustrations are bleeding, in accordance with the TU Delft style.

Before generating the cover, you need to provide the information to put on it. This can be done with the following commands:

- `\title[Optional Color]{Title}`
  This command is used to provide the title of the document. The title title is also printed on the spine. If you use a title page (see below), this information will be used there as well. As the title, subtitle and author name are printed directly over the cover photo, it will often be necessary to adjust the print color in order to have sufficient contrast between the text and the background. The optional color argument is used for this.

- `\title[Optional Color]{Subtitle}`
  This command is used to provide a subtitle for the document. If you use a title page (see below), this information will be used there as well. It possible to adjust the print color in order to have sufficient contrast between the text and the background – the optional color argument is used for this.

- `\author{J. Random Author}`
  This command specifies the author. The default color is `tudelft-white`, but this may be adjusted in the same way as the titles.

- `\affiliation{Technische Universiteit Delft}`
  The affiliation is the text printed vertically on the front cover. It can be the affiliation, such as the uni-

versity or department name, or be used for the document type (*e.g.*, Master's thesis). The default color is again `tudelft-white`, adjustable through the `color` option.

- `\coverimage{cover.jpg}`
  With this command you can specify the filename of the cover image. The image is stretched to fill the full width of the front cover (including the spine if a back cover is present).

- `\covertext{Cover Text}`
  If a back cover is present, the cover text is printed on the back. Internally, this text box is created using the LaTeX `minipage` environment, so it supports line breaks.

- `\titleoffsetx{OffsetX},\titleoffsety{OffsetY}` If the cover page contains a page-filling picture (i.e., `split` is not specified with the `makecover` command, the best position of the title depends a lot on the picture chosen for it. The lower left corner of the minipage containing title, subtitle and author is specified by these two commands. The offsets are measured from the top left corner of the page.

- `\afiloffsetx{AfilX}, \afiloffsety{AfilY}` specifies the lower left corner of the text containing the affiliation, measured from the top left corner of the page.

In addition to `[split]`, the `\makecover` command accepts several additional options for customizing the layout of the cover. The most important of these is `back`. Supplying this option will generate a back cover as well as a front, including the spine. Since this requires a page size slightly larger than twice A4 (to make room for the spine), and LaTeX does not support different page sizes within the same document, it is wise to create a separate file for the cover. `cover.tex` contains an example. The recommended page size for the full cover can be set with

\geometry{papersize={1226bp,851bp}}

after the document class and before `\begin{document}`.

The other options `\makecover` accepts are

- `nospine`
  If a back cover is generated, the title will also be printed in a black box on the spine. However, for smaller documents the spine might not be wide enough. Specifying this option disables printing the title on the spine.

- `frontbottom`
  By default the black box on the front is situated above the blue box. Specifying this option will place the black box below the blue one.

- `spinewidth`
  If a back cover is present, this option can be used to set the width of the spine. The default is `spinewidth=1cm`.

- `frontboxwidth`, `frontboxheight`, `backboxwidth`, `backboxheight`
  As their names suggest, these options are used to set the width and height of the front (black) and back (blue) boxes. The default widths and heights are `4.375in` and `2.1875in`, respectively.

- `x`, `y`
  The blue and black boxes touch each other in a corner. The location of this corner can be set with these options. It is defined with respect to the top left corner of the front cover. The default values are `x=0.8125in` and `y=3in`.

- `margin`
  This option sets the margin between the borders of the boxes and their text. The default value is `12pt`.

For a thesis it is desirable to have a title page within the document, containing information like the thesis committee members. To give you greater flexibility over the layout of this page, it is not generated by a command like `\makecover`, but instead described in the file `title.tex`. Modify this file according to your needs. The example text is in English, but Dutch translations are provided in the comments. Note that for a thesis, the title page is subject to requirements which differ by faculty. Make sure to check these requirements before printing.

3.3. Pre-Processing

Each chapter has its own file. For example, the LaTeX source of this chapter can be found in `chapter-1.tex`. A chapter starts with the command

```
\chapter{Chapter title}
```

This starts a new page, prints the chapter number and title and adds a link in the table of contents. If the title is very long, it may be desirable to use a shorter version in the page headers and the table of contents. This can be achieved by specifying the short title in brackets:

```
\chapter[Short title]{Very long title with many words which could not possibly
fit on one line}
```

Unnumbered chapters, such as the preface, can be created with `\chapter*{Chapter title}`. Such a chapter will not show up in the table of contents or in the page header. To create a table of contents entry anyway, add

```
\addcontentsline{toc}{chapter}{Chapter title}
```

after the `\chapter` command. To print the chapter title in the page header, add

```
\setheader{Chapter title}
```

Chapters are subdivided into sections, subsections, subsubsections, and, optionally, paragraphs and subparagraphs. All can have a title, but only sections and subsections are numbered. As with chapters, the numbering can be turned off by using `\section*{...}` instead of `\section{...}`, and similarly for the subsection.

3.4. Entrances

3.4.1. \subsection{...}

\subsubsection{...}

3.4.1.1.1 \paragraph{...} Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

3.5. Static and mobile devices

The fonts used by this template depend on which version of LaTeX you use. Regular LaTeX, *i.e.*, if you compile your document with with `latex`, `pslatex` or `pdflatex`, will use Utopia for text, Fourier for math and Latin Modern for sans-serif and monospaced text. However, if you want to adhere to the TU Delft house style, you will need to use XeLaTeX, as it supports TrueType and OpenType fonts. Compiling with `xelatex` will use Arial for most titles and text, Courier New for monospace and Cambria for math. If you want to haf a sans-serif font for the main text, while using `latex`, `pslatex` or `pdflatex`, you can use the option `noroman` in the report style: `\usepackage[...,noroman]tudelft-report`. For document and part titles, TU Delft Ultra Light is used. For quotes, columns and text in boxes, you use Georgia. If you want to use XeLaTeX, but do not want to use the TU Delft house style fonts, you can add the `nativefonts` option to the document class. This will still use TU Delft Utra Light and Arial on the cover, but not for the body of the document. If you need to use these fonts for certain sections in the main text, they are available via `\tudrmfamily` (Georgia) and `\tudtitlefamily` (TU Delft Utra Light).

> You have to learn the rules of the game. And then you have to play better than anyone else.
> *Albert Einstein*

The corporate colors of the TU Delft are cyan, black and white, available via `\color{tudelft-cyan}`, `\color{tudelft-black}` (which differs slightly from the default `\color{black}`) and `\color{tudelft-white}`, respectively. Apart from these three, the house style defines the basic colors `tudelft-sea-green`, `tudelft-green`, `tudelft-dark-blue`, `tudelft-purple`, `tudelft-turquoise` and `tudelft-sky-blue`, as well as the accent colors `tudelft-lavendel`, `tudelft-orange`, `tudelft-warm-purple`, `tudelft-fuchsia`, `tudelft-bright-green` and `tudelft-yellow`.