CycleCV - Detect, Analyze and Track Cyclists' Movements

Bachelor Project - Data Science

Author:
Simon Martin Breum, smbr@itu.dk (simon009@breum.dk)

Supervisors:
Bojan Kostic, boko@itu.dk
Michael Szell, misz@itu.dk

May 16^{th}, 2022 — Revised June 9^{th}, 2022

Abstract

Bicycle infrastructure is essential part of the growing focus on the environment and renewable energy. With Copenhagen having been ranked the most bicycle-friendly city in the world and Denmark having the second-highest levels of cycling, cyclist safety and the practicality of cycling is highly prioritized when it comes to Danish traffic infrastructure.
At the Dybbølsbro intersection in Copenhagen however, the bicycle infrastructure is suboptimal, and the Municipality of Copenhagen has proposed and approved a solution to the issues related to the intersection. This paper uses DBSCAN and Dynamic Time Warping (DTW) to time-independently analyze patterns in cyclist behaviour in the intersection. The solution proposed and approved by the Municipality of Copenhagen is then evaluated against the problematic patterns found during analysis, finding that the proposed solution looks promising because it targets all of the issues found. However, new potentially unaddressed issues are found, that might have consequences for the new infrastructure. Each of these have been described and given a potential solution.

Contents

1 Introduction

In a world with a growing focus on the environment and renewable energy, bicycle infrastructure is essential. In a city that does not properly support cycling, the only general means of transportation are walking or using a motor vehicle (public transportation or car).
For longer distances, walking is not feasible because of physical condition and/or time requirements, hence motor vehicles become the default choice. However, most of these distances could be traversed using a bicycle, either electric or not; both would be better for the environment. Bicycle riding is both a environmentally friendly form of transportation, and it simultaneously improves the physical health of the population (Green, 2020). Additionally, riding a bicycle requires no fossil fuels, unlike cars.
Even for longer trips where the physical condition of the cyclist might not suffice, there exist electrical bicycles to enable cyclists to cover longer distances using the same amount of physical effort. Electrical bicycles can be charged using renewable energy from solar cells or windmills to avoid unnecessary pollution.
To increase the ratio of cyclists to motorists, cities must improve their bicycle infrastructure. However, even cities which generally have great bicycle infrastructures might have issues.
With Copenhagen as the leading city in the world when it comes to bicycle-friendliness (Copenhagenize, 2019), and Denmark having the second-highest levels of cycling (Pucher & Buehler, 2007), it is not a surprise that we are one of the leading bicycle-countries in the world (Pucher & Buehler, 2007).
Being on the front lines of bicycle-friendly infrastructure, we highly prioritize the safety and practicality of cycling. Therefore, if some of our infrastructure was to falter, we try to rectify it.
The bicycle-friendliness of the Dybbølsbro intersection has faltered, and changes to the the infrastructure in the intersection have been proposed and approved.

1.1 Problem Definition

Using a Data Science approach, this paper aims to assess the proposal by analyzing tendencies in the intersection, focusing on traffic violations and otherwise dangerous situations. These tendencies will be used to evaluate and discuss the proposed changes, as well as try to predict potential future issues that might arise from said changes.

Including:

- Whether cyclists currently follow traffic regulations in the intersection
- Whether the intersection promotes easy-to-follow behavior
- Whether the approved proposal addresses any found issues
- Whether the approved proposal creates any new issues

Note: This paper will use data gathered using Computer Vision (hence the title CycleCV), but will otherwise *not* focus on this topic.

2 Terminology

Dybbølsbro will be used to refer to the bridge unless accompanied by the word "intersection", in which case it refers to the Dybbølsbro intersection.

Cardinal directions (north, south, east and west) will be used to indicate specific roads or directions. When a cardinal direction is used, it refers to the road or direction of the road most closely aligned with that cardinal direction. In practice, the cardinal directions refer to/towards the following roads:

- North: Skelbækgade
- East: Ingerslevsgade (towards east-northeast)
- South: Dybbølsbro
- West: Ingerslevsgade (towards west-southwest)

Traffic violation(s) refers (in this report) to any violation(s) of the Danish Traffic Laws for bicycles (The Danish Ministry of Transportation, 2021).

Source-destination cluster(s) refers to cluster(s) obtained from using DBSCAN (see 5.2 Initial Clustering Using DBSCAN).

Path refers to the time-series data of positions (image coordinates) for a sample in the data set (see 4.2 CV Road User Intersection Data).

Path cluster(s) refers to cluster(s) obtained from using Hierarchial clustering with Dynamic Time Warping distance (see 5.3 Deeper Clustering Using Dynamic Time Warping).

Median strip refers to a "spærreflade", the striped zone between some opposing car lanes (see figure 1). Other types of median strips exist (with trees etc.), but for the purposes of this paper, only the "spærreflade" is referred to using this wording.

Figure 1: An example of a Danish "spærreflade" [median strip].

3 Background & Related Work

3.1 The Dybbølsbro Intersection

The Dybbølsbro intersection lies at the crossing of Ingeslevsgade, Skelbækgade and the bridge Dybbølsbro (hence the name of the intersection). In 2019, Dybbølsbro (the bridge) was redesigned to accommodate a higher throughput of pedestrians and cyclists. In this context, the bicycle lane was extended to a 10 meter wide bi-directional bicycle lane, making it the widest in the city (Bach, 2018).
The new bi-directional bicycle lane was, however, on one side of the bridge, rather than following the standard of a bicycle lane on each side of a road or bridge. Because of this deviation from the norm, cyclists coming from the Dybbølsbro intersection must cross the road perpendicular to the bridge before crossing the bridge itself; an action which not very intuitive and increases the rate of accidents (Therkildsen, 2021). This lack of intuitiveness invites many dangers for especially cyclists that are not regulars in this intersection. Despite being a busy intersection with 20.400-22.000 cyclists crossing each day (Therkildsen, 2021; Bach, 2018), this intersection is infamous for its problematic design (Bach, 2018; Therkildsen, 2021; Nørd News [Nord News], 2021).
Even though the design of the intersection is dependent on the design of Dybbølsbro and could in theory be solved by altering both, it is not possible to simply "fix" the intersection and the bridge, since that would also require altering the two intersections on the other side of the bridge, because these rely on the design choices mentioned above.

Figure 2: Satellite image of the Dybbølsbro intersection. Source: Google Maps

3.2 DBSCAN

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) (Ester et al., 1996) is a fast, high-quality and widely used (SIGKDD, 2014; Schubert et al., 2017) clustering algorithm to group together data points in high density areas, while excluding points in low density areas. The algorithm takes any number of features and has two parameters; *eps* and *min_samples*.
The DBSCAN algorithm consists of four phases:

1. **Identify the core points in the data.** — A core point is a point that has at least *min_samples* samples within a radius of *eps*.

2. **Assign core points to clusters:**
   (a) Randomly assign an unassigned core point to a *new* cluster.
   (b) Assign all core-points within radius *eps* to the same cluster.
   (c) Repeat step 2b for each newly assigned core point until no new points are assigned to the cluster.
   (d) Repeat the above steps until no unassigned core-points remain.

3. **Assign non-core points to clusters**
   - For each cluster (in order), for each point in that cluster, if any non-core points are within radius *eps*, add that point to the cluster.

4. **Any unassigned (non-core) points are unclustered** (cluster -1)
   - For the purposes of this paper, any unclustered points are left out of analysis. This should not have a huge effect on the outcome (see 5.2 Initial Clustering Using DBSCAN).

This paper uses DBSCAN on the source- and destination coordinates of paths to find the most dense spatial source-destination combinations for cyclists. This method intentionally does not take into account how a path gets from the source to their destination. This allows us to cluster cyclists based on "goals" (to get from the same A to B). Each cluster is then hierarchically sub-clustered separately using Dynamic Time Warping distances between paths.

3.3 Dynamic Time Warping

Dynamic Time Warping (DTW) (Müller, 2007) is used to calculate the distance between two time-series of observations. The method allows warping time to account for variances in speed. Similar paths will thus have a low distance and very different paths will have a very large distance.
For any two given time-series of observations, the DTW-distance is the summed distance between pairs of data points such that:

- Each observation can be paired only with observations from the other time-series and must be paired with at least one data point (but can be paired with multiple).
- The start-points of the time-series are paired (they may be paired with other points too).
- The end-points of the time-series are paired (they may be paired with other points too).
- Pairings must be continuous — For any pairing such that observation o in the time series S_1 is paired with observation p in time series S_2, any observation in S_1 that is before o cannot be paired with an observation in S_2 that is after p, and vice versa.

This paper uses DTW distances to calculate similarity between paths with similar source and destination. The distance is then used to hierarchically cluster paths further, allowing for more efficient and effective manual inspection; of groups of similar paths, rather than individual paths.
This sub-clustering is very different from the DBSCAN clustering, since it takes the full paths into account, as well as ignores the timing of the path. This means two identical paths, where one has a lower average speed, will still be clustered together due to having 0 DTW distance.

4 Data

The data used for this paper is based on a 720p video of the Dybbølsbro intersection. The video spans a 10 hour time period with a frame rate of 30 frames per second. For this paper, the spatial data used has been created from this video using Computer Vision.
The video is sourced from a camera facing roughly north. The top of the image is thus referenced as "north" according to the specifications in 2 Terminology.

4.1 Intersection Video Sample

The full 10 hour video has not been made available. Instead, accompanying the data is a 1 hour sample of the video in 360p for GDPR reasons. This is used to make plots, perform manual inspection and for sanity checks on results.

4.2 CV Road User Intersection Data

This spatial time-series data set[1] used contains information on road users, where each line is a record of a single road user in the Dybbølsbro Intersections during a 10 hour time period. The data set consists of a total of 18842 samples, each with the following data:

ID	Unique identifier for the road user
Type	Road user type (pedestrian/cyclist/car/bus/truck)
Frames	Frames on screen; first frame, last frame and number of frames. 1 frame corresponds to $^1/_{30}{}^{th}$ of a second.
Positions	Image coordinates (x,y) as floating point pixel values on each frame (center of their bounding box during CV identification)
(Other)	*(Other data that is not used, either because it is imprecise or masked)*

Note: Many paths in the data are 'broken' due to occlusion or split into two samples due to missing connection for same object between two frames (two samples with individual paths exist for the same object; when one ends, the other begins.)
The problematic data is filtered out as described in 5.1 Preprocessing.

[1]Split into two files; one contains the data, but has incorrect frame numbers; the other file contains corrections to the first data file.

4.3 Traffic Signal Lines Data

To assess whether cyclists in the intersection behave as intended, information on the signal lines must be used; this data consists of two data sets joined into one.

4.3.1 Traffic Lines Data (Spatial)

The Traffic Lines data set consists of two files; the real locations of traffic lines (car stop-lines, starts of pedestrian crossings, etc.) and their adjusted locations.
Coordinates of road users are given as the center of their respective bounding boxes. Because of this, the time that a cyclist's coordinates cross a signal line is different from when the cyclist physically crosses that signal line (as one wheel crossing the signal line counts as having crossed it). Therefore, signal lines all have an adjusted location such that when cyclists physically cross a signal line in the video, the adjusted signal line is also crossed by their coordinates at the same time (see figure 4). This adjusted signal line location allows for analysis on cyclists running red lights etc.
The data set contains the following information for each signal line in the videotaped area:

ID	Unique identifier for the signal line
Type	Intended road user type for the signal line (car/cyclist/pedestrian)
Location	Coordinates (x,y) for the points in the signal line as seen on the video.
Adjusted Location	'Location' adjusted according to the CV Road User Intersection Data.

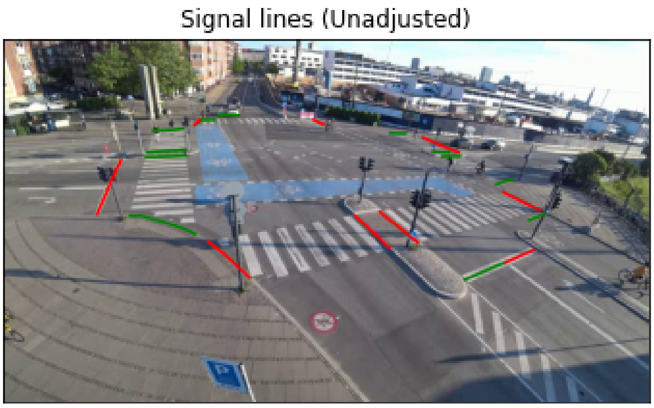


Figure 3: The unadjusted signal lines, colored by signal color (see section 4.3.2). Note that the unadjusted signal lines are exactly on top of the real signal lines.

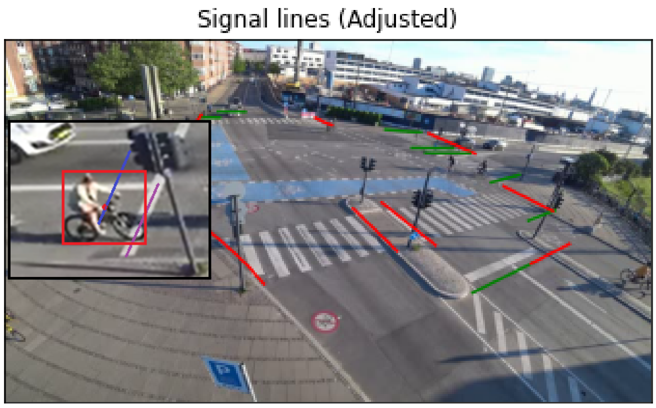


Figure 4: Signal lines adjusted to fit the coordinates of road users. Note that the cyclist crosses the real signal line (purple), as their coordinates (red dot; middle of bounding box) simultaneously cross the adjusted signal line (blue).

4.3.2 Traffic Signal Data (Time-series)

The Traffic Signal data set contains the ID and colors for each traffic light on each frame (see figures 3 & 4). The colors of the signal lines are referenced as follows:

Letter	Color	Comments
r	Red	—
y	Yellow	When light is turning from green to red
g	Green	—
o	Orange	Red+yellow; when light is turning from red to green

5 Methodology

The processing pipeline for the data and analysis in this paper contains several steps (see figure 5). Initially, the data is filtered such that only the most useful data is kept (complete bicycle samples). Next, data is clustered twice; first on path sources and destinations, then on individual paths. Lastly, incorrect intersection usage was detected; partially using manual inspection.

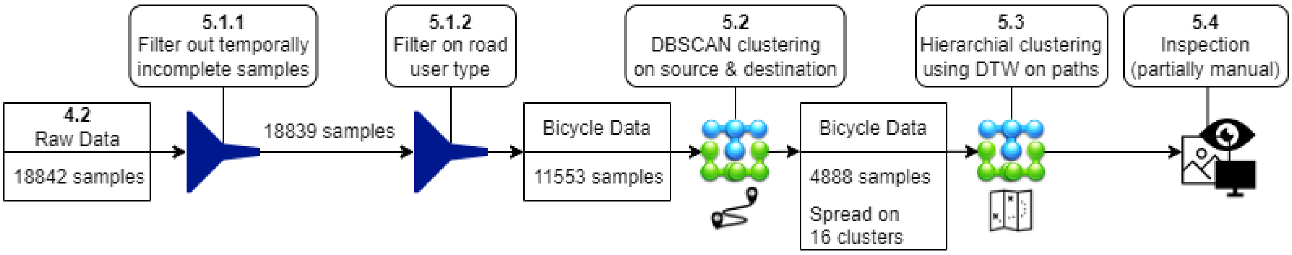


Figure 5: Flowchart of the processing pipeline. Each step is named and numbered according to the corresponding section in this paper.

5.1 Preprocessing

The data is separated in many files; some with corrections. This section describes the overall preprocessing steps made.

5.1.1 Filter out temporally incomplete samples

Given that our data originates from a slice of time, some sample paths are expected to be incomplete because they were either not tracked from they entered the frame (if they entered before the video started), or if they were not tracked when they exited the frame (if they exited after the video ended). Therefore, we filter out samples that are already in the frame when the tracking begins (when 'frame_in' == 0) or when it ends. Since we do not know the exact number of frames in the original video, we filter out all samples that have the latest exit time.

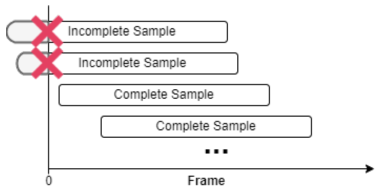


Figure 6: Samples filtered out because they enter the frame before tracking starts (at frame=0).

Listing 1: Filtering out incomplete samples.

```
# Drop incomplete paths
df.drop(index=df.index[df["frame_in"] == 0], inplace=True)
df.drop(index=df.index[df["frame_out"] == max(df["frame_out"])], inplace=True)
```

In total, 3 samples were filtered out, resulting in 18839 being left.

5.1.2 Filter on Road User Type (bicycle)

Most calculations and analyses in this paper are done using purely cyclist data. For these, a subset of the data is used, containing only cyclist samples:

Listing 2: Creating a cyclist subset of the data.

```
# Create a new DataFrame only containing cyclist samples
df_bike = df[:][df["class"] == "Bicycle"] #[:] makes sure that it's a copy and not a slice
```

The bicycle subset of the data contains 11553 samples.

5.2 Initial Clustering Using DBSCAN

The bicycle samples were initially clustered using DBSCAN (see 3.2 DBSCAN) on endpoints for each path. This gave general clustering of paths such that each cluster represented "from A to B" with the same A and B (source and destination), regardless of the pathing between A and B. These will be referenced as "source-destination clusters".

The best parameters were found using a narrowing grid-search-like approach on ranges of integers with manual inspection focusing on number and quality of non-duplicate intended[2] paths.

- Initial ranges of parameter values to check were selected based on very generous estimate ranges:

  $eps := \{10, 20, 30, ..., 100\} \qquad min\_samples := \{50, 100, 150, ..., 500\}$

- Ranges were gradually narrowed ($\xRightarrow{\text{Narrow}}$) such that the narrowing with largest impact on cluster number and quality[3] was performed first. The ideal clustering has one cluster for each intended source-destination pair in the intersection (with as many samples as possible within each).

The parameters were narrowed using the following ranges. For each iteration, the best value in the range is found (Noted under arrows; e.g. $\xRightarrow[\text{(Best:10)}]{\text{Narrow}}$), and the range narrowed around that value. If the lowest value of a range is the best, the narrowed range will also be given elements closer to 0:

$$eps : \{10, 20, 30, ..., 100\} \xRightarrow[\text{(Best:10)}]{\text{Narrow}} \{5, 10, 15, 20\} \xRightarrow[\text{(Best:10)}]{\text{Narrow}} \{5, 6, 7, ..., 15\} \xRightarrow{\text{Best}} 9$$

$$min\_samples : \{50, 100, 150, ..., 500\} \xRightarrow[\text{(Best:50)}]{\text{Narrow}} \{25, 50, 75, 100\} \xRightarrow[\text{(Best:25)}]{\text{Narrow}} \{5, 10, 15, ..., 50\}$$
$$\xRightarrow[\text{(Best:25)}]{\text{Narrow}} \{20, 21, 22, ..., 30\} \xRightarrow{\text{Best}} 25$$

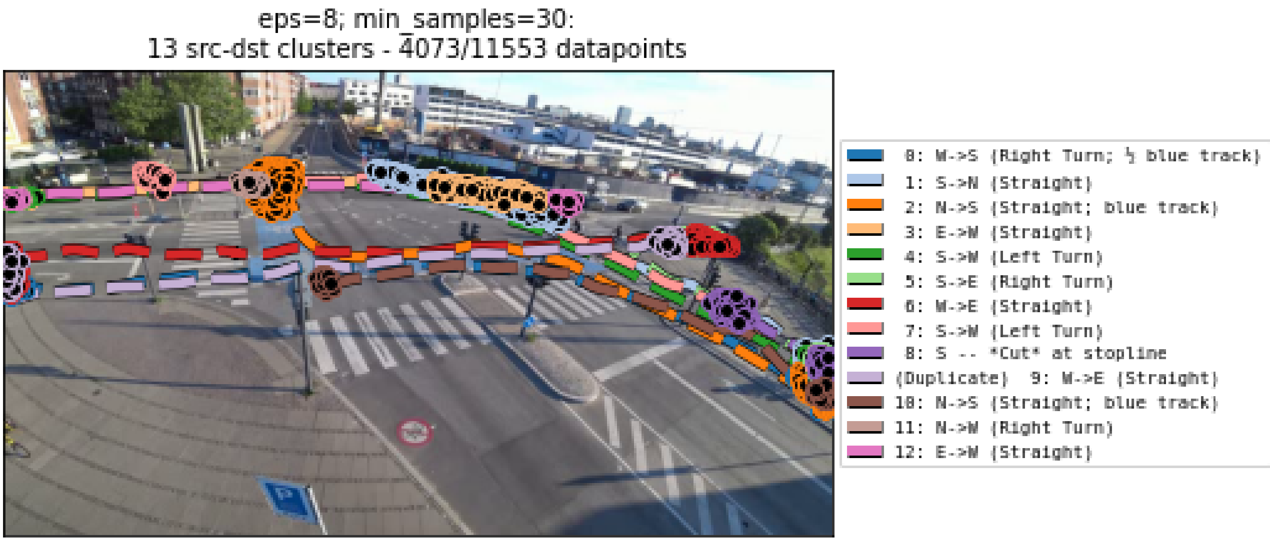


Figure 7: Example during step of the parameter search with eps=8 and min_samples=30. Note that there is only 13 source-destination clusters; eps is too low and/or min_samples is too high.

After clustering with DBSCAN using the best found parameters, 4888 samples were designated one of 16 clusters, and 6665 samples were not eligible for clustering. Although more than half of the given samples were filtered out, this should not have a negative effect on the outcome of the analysis since most are broken trajectories. The clustered samples are complete except for those in source-destinaion clusters 11 and 12 (tracking errors); spanning a complete path between a common source and destination.

[2] As per traffic regulations.
[3] During manual inspection.

5.3 Deeper Clustering Using Dynamic Time Warping

For each source-destination cluster[4], the samples were hierarchically clustered into 20 path clusters using DTW distances (see 3.3 Dynamic Time Warping). The 20 path clusters each consisted of a group of spatially similar paths, although independent of timing and speed.

The sub-clusters obtained with hierarchial clustering using the DTW distance of source-distance clusters will be referenced as "path clusters".

Note: In this paper, DTW is implemented in Python (see code exerpt below), however the DTW distance calculation was performed using DTAI's library for time series distances ("Wannesm" et al., 2022), as it allows using optimized (Salvador & Chan, 2004) and pre-compiled C code, which is much faster than my Python implementation. The following code excerpt is a simplified version[5] of my implementation. The algorithm is the same.

Listing 3: (Simplified version of) my implementation of the DTW distance calculation function.

```python
def dist(a, b):
    #Euclidean distance between points a and b.
    return np.sqrt(abs(a[0]-b[0])**2+abs(a[1]-b[1])**2)

def dist_DTW(ax, ay, bx, by):
    # Arguments 'ax', 'ay', 'bx' and 'by' are all arrays of x- or y-values.
    # Lengths of a- and b- components may differ in length between a and b (e.g. ax and bx), but must be same length for both components of itself (e.g. ax and ay)
    n, m = len(ax), len(bx)
    DTW = np.full(shape=(n, m), fill_value = np.inf)
    DTW[0, 0] = 0
    for i in range(n):
        for j in range(m):
            if i==j==0:
                continue
            cost = dist((ax[i], ay[i]), (bx[j], by[j]))
            DTW[i, j] = cost + min(DTW[i-1, j  ],    # insertion
                                   DTW[i  , j-1],    # deletion
                                   DTW[i-1, j-1])    # match
    return DTW[-1,-1]
```

[4]See 5.2 Initial Clustering Using DBSCAN

[5]For readability purposes. Simplification is done on the format of the arguments (input data)

5.4 Detection of Incorrect Intersection Usage

After deeper clustering of the data, each path cluster was visualized and manually inspected for path choices that go against the intended ways to cross the intersection, including whether it used the pedestrian crossing and whether it was off-course with respect to the intended pathing[6]. The following incorrect behaviors are looked for:

Corner-cutting Some cyclists are expected to cut corners. In this case, crossing corners of pedestrian crossings. The paths that cut corners over pedestrian crossings can be found by first defining the bounding boxes of each pedestrian crossing, then checking if each path overlaps any pedestrian crossing or bicycle signal lane at any point. This has the caveat of mislabeling in situations where cyclists are meant to cross pedestrian crossings. Therefore, any path that overlaps a non-bicycle signal line is labeled as corner-cutting instead. A path that crosses a pedestrian crossing sideways while not supposed to (without crossing signal lines) will be labeled corner-cutting during manual inspection.

Using the Pedestrian Crossing For a cyclist, there is two ways of using the pedestrian crossing; one legal, one not. Riding the bicycle on the pedestrian crossing is illegal, but getting off one's bicycle and pulling it across the pedestrian crossing is allowed. Because there is no distinction between these cases in the data, paths that utilize pedestrian crossings without simply cutting corners are identified with manual inspection of path clusters[7].
If a cyclist 'uses the pedestrian crossing', it is not evaluated as 'cutting corners' (see above).

Confusion When dealing with unintuitive intersections, it is expected that some cyclists get confused; they make a 180-degree turn, cross car lanes etc.
These paths are easiest to find with manual inspection, thus this is done for each path cluster.

Clear traffic violations Excluding running red lights (tracked separately), some cyclists commit clear traffic violations; they clearly cycle on the pedestrian crossing, cross a median strip etc.
These paths are easiest to find with manual inspection, thus this is done for each path cluster.

Running red lights Some cyclists are expected to run red lights (Pettinger, 2022). Crossing a signal line during a red light (not necessarily crossing the intersection) is a violation of the Danish traffic regulations (The Danish Ministry of Transportation, 2021). Therefore, the number of cyclists crossing a signal line during a red light are noted, whether or not they cross the intersection.

[6]The intended pathing is assumed to match path cluster 0 (and without making traffic violations; some paths in these path clusters cut corners over pedestrian crossings, which is not the intended pathing of the intersection.) for each source-destination cluster. These have been sanity-checked manually.

[7]This could also be evaluated by checking if a path crosses both ends of a pedestrian crossing, but would leave out the paths who use e.g. 80 or 90% of a pedestrian crossing, but do not cross both ends

6 Results & Discussion

6.1 Source-Destination Clusters (DBSCAN)

Using DBSCAN on the data set yielded 16 source-destination clusters, one of which represents an artifact in the data set[8]. Most source-destination clusters represent a unique combination of source and destination in the intersection, e.g. all cyclists traveling from West to South are in source-destination cluster 0.

Note: East⇒North is not present in the clusterings. This is likely due to the tendencies in the data itself; these samples would be far from the camera and thus more unclear or indistinguishable, resulting in being unclassified or misclassified by the CV algorithm.

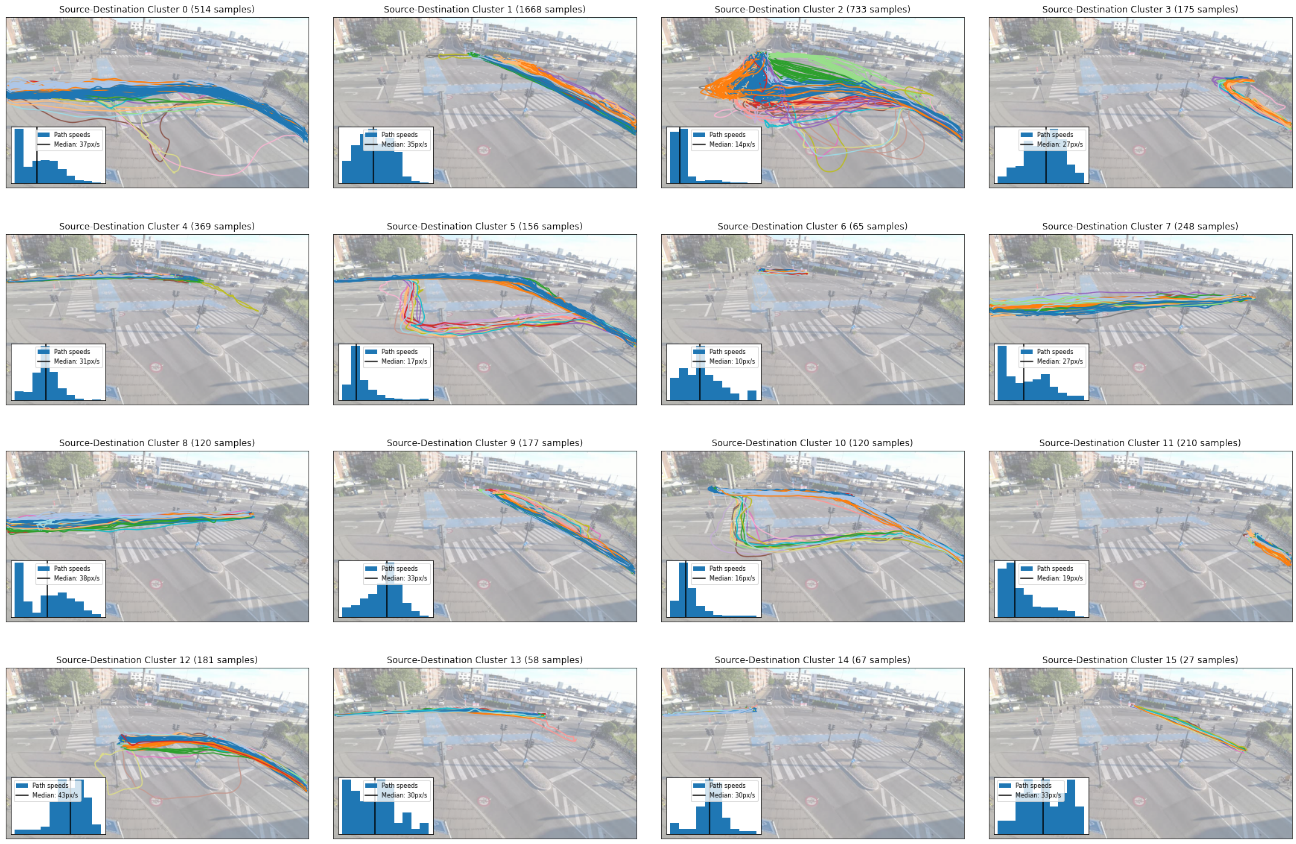


Figure 8: Source-destination clusters ordered by number and visualized (top left is 0, top right is 3, bottom right is 15). Each source-destination cluster has a histogram of path speeds and distinctly colored path-clusters. Note that the histograms are meant to be compared with each of their path clusters, not between source-destination clusters, hence the axes of the histograms are not equal.

Most source-destination clusters have either one or two general paths from source to destination. For those that cross the road more than once, there seems to be paths both clockwise and counter-clockwise. However, source-destination cluster 2 seems to have paths all over the intersection; some even crossing diagonally through the intersection[9]!

[8] A simple tracking error in the CV algorithm, likely due to occlusion from the traffic signal pole.
[9] See 6.3 Deeper Analysis of Source-Destination Cluster 2

6.2 Incorrect Usage (by Path and Category)

Following the definitions of 5.4 Detection of Incorrect Intersection Usage, the source-destination clusters had the following incorrect usage statistics amongst cyclists. Numbers in parentheses (in the columns "Pedestrian Crossing" and "Confusion/Violations") indicate which path clusters contribute to the percentages. For source-destination cluster 2, these can be found in section 6.3 Deeper Analysis of Source-Destination Cluster 2. Additionally, plots of all path clusters are in the appendix.
Not all source-destination clusters have both confusion and violations. If only one is indicated in the table, the other is 0%. If a cell contains only a percentage, it represents both confusion and violations. The percentage of cyclists running red lights is the technical violation of crossing a signal line while the signal is red (see 5.4 Detection of Incorrect Intersection Usage).

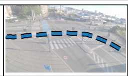
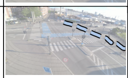
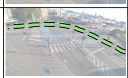
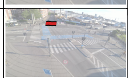
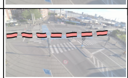
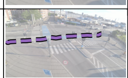
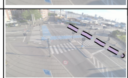
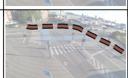
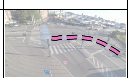
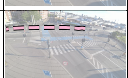
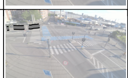
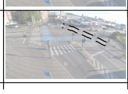
Src-Dst Cluster	**Src⇒Dst**	**Path (Visual)**	**Corner Cutting**	**Pedestrian Crossing**	**Confusion / Violations†**	**Crossing Red Light**
0	West⇒South		76.5%	2.7% *(3-5,8-12,14,17,19)*	1.0% Confusion *(9,10,13,17,18)*	86.8%
1	S⇒N		31.4%	0.1% *(11, 14)*	0.2% Confusion *(9, 10, 14, 15)*	20.0%
2	N⇒S		73.2%	8.9% *(0-3,5-10,12-16,18)*	**6.8% Confusion** **10.8% Violations** *(5,11-19)(4-5,11-19)*	85.1%
3	S⇒E		97.1%	0.6% *(13)*	0%	84.0%
4	E⇒W		5.7%	*Not visible*	*Not visible*	5.1%
5	S⇒W		71.2%	12.8% *(3,6-13,15-19)*	0.6% Violations *(13)*	48.1%
6	N⇒E *		1.5%	*Not visible*	1.5% *(15)*	15.4%
7	W⇒E		48.4%	0%	0%	54.8%
8	W⇒E ¤		23.3%	0%	0%	30.8%
9	E⇒S **		86.4%	5.1% *(7,9-12,16-18)*	0% or 100% **	89.8%
10	S⇒W		81.7%	10.0% *(3-5,8-10,12-15)*	0%	71.7%
11	*Tracking Err.*	—	—	—	—	—
12	N/W⇒S ¤		70.1%	3.9% *(4,12)*	0.6% *(17)(11)*	79.6%
13	E⇒W		22.4%	1.7% *(7)*	1.7% Confusion *(7)*	5.2%
14	N⇒W		10.4%	0%	0%	9.0%
15	S⇒N		14.8%	0%	0%	14.8%

Table 1: Summary statistics for traffic violations of the source-destination clusters. Parentheses refer to path clusters. Note that numbers for red light violations is abnormally high; likely due to uncertainty of centroids when adjusting the signal lines in 4.3.1 Traffic Lines Data (Spatial).

† Both equal unless otherwise specified. If only one is noted, the other is 0%.
* Path is not the intended way to go from the source to the destination.
** Path might not be the intended for this source and destination (no indication of intended path).
¤ In multiple clusters due to varying tracking sources/destinations.

6.3 Deeper Analysis of Source-Destination Cluster 2

As identified in 6.1 Source-Destination Clusters (DBSCAN), the source-destination clusters had little variance of path clusters. However, source-destination cluster 2 (North⇒South) had very variable path clusters among its 733 samples. They were almost spread out over the whole intersection; some even crossing the intersection diagonally (see figure 9). Many of the path choices seen in source-destination cluster 2 are path choices that do not match the intended pathing in the intersection and/or are clear traffic violations.
Since this source-destination cluster is so different from the others in terms of path cluster diversity and confusion-/violation-rate (see table 1), it prompts deeper analysis:

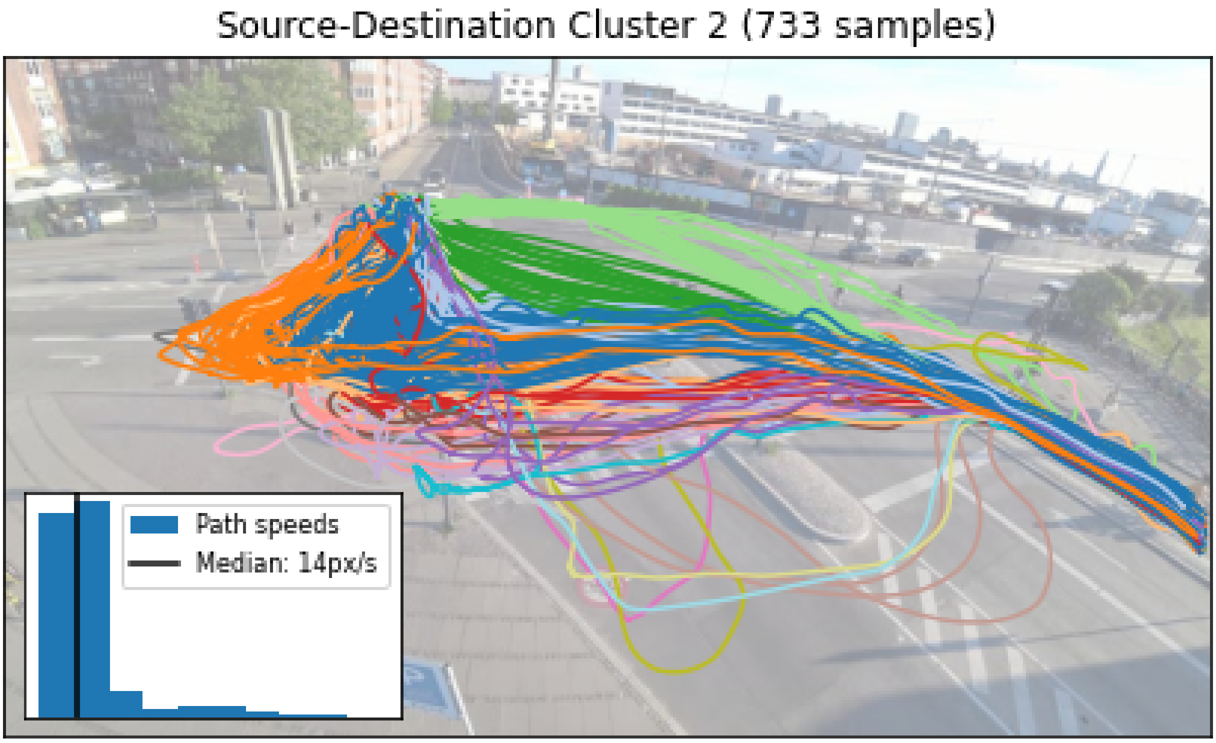


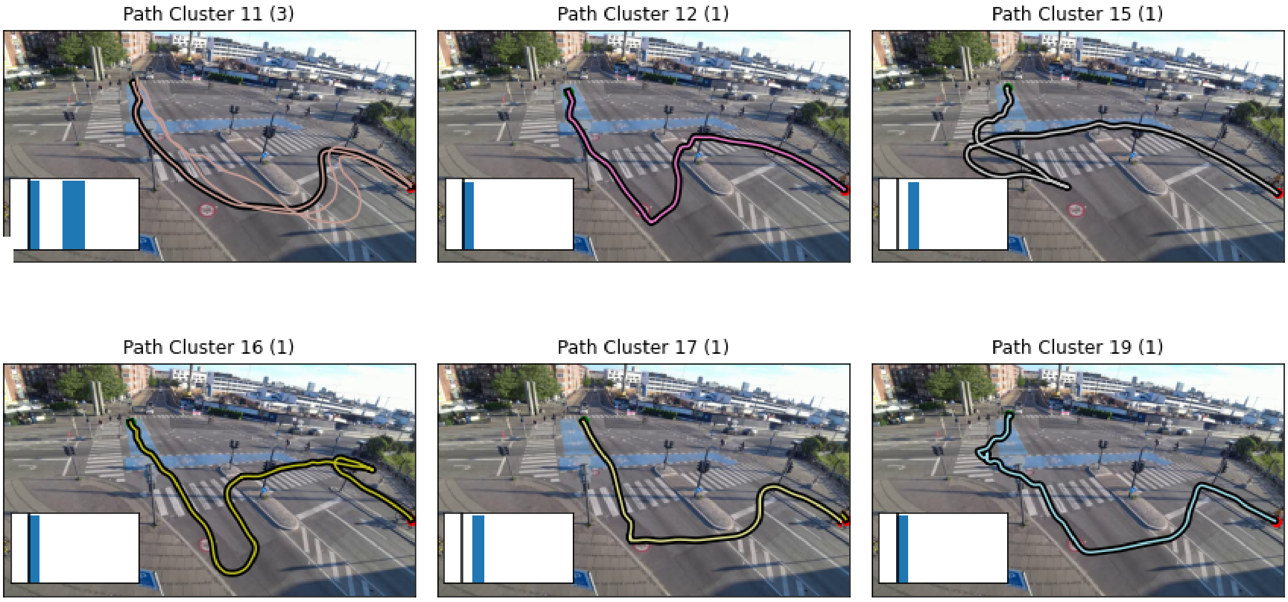
Figure 9: Source-destination cluster 2. Colors are the different path-clusters. Note that even though all paths go from north (top left of the image) to south (bottom right), the paths cover almost the entire intersection.

Path clusters in this source-destination cluster have signs of both confusion and corner-cutting, as well as what appears to be intentionally breaking traffic regulations (e.g. crossing diagonally through the intersection). On figure 9 is seen four main issues[10]:

- **Confusion** A few paths have traveled too far, crossing into the "no-bicycle zone" at the bottom middle of the image (see figure 3 for clearer marking).
- **Driving on Pedestrian Crossing for Alignment** One path cluster (orange) drives on the pedestrian crossing on the left to align themselves properly.
- **Clockwise Crossing** One path cluster (light green) crosses the intersection clockwise instead of the intended counter-clockwise that is indicated with lane markings.
- **Diagonally Crossing** One path cluster (dark green) cuts diagonally through the intersection instead of following the lane markings.

[10]Ignoring the use of pedestrian crossings since we do not know whether they are used correctly or not.

6.3.1 Confusion

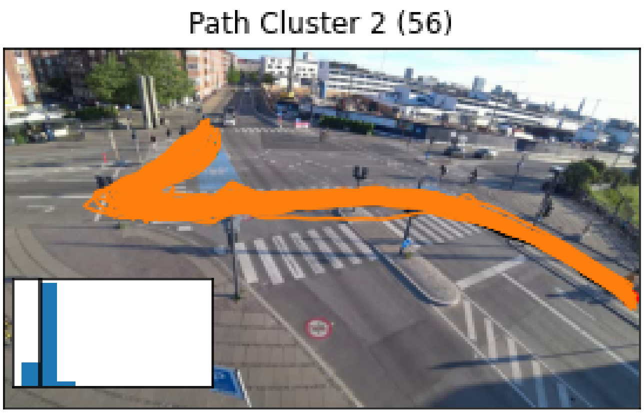


The deeper clustering using DTW indicated 6 path clusters with a total of 8 samples that, from their path choices, look like they assumed that the bicycle lane was going to continue on the south side of the intersection, only to find out that it did not:

- Path clusters 11, 17 and 19 (5 samples total) all show paths that crossed the median strip between the car lanes. This is no place to be for a bicycle and is a traffic violation.
- Path clusters 12 and 16 (2 samples total) did not cross the median strip after driving too far. Instead, they seemingly chose to drive on the pedestrian crossing[11], which is a traffic violation.
- Path cluster 15 (1 sample) had the most correct reaction to the confusion; they notice that bicycles are not allowed further, then stops, turns around, and proceeds to cross in the location they should originally have crossed.

Out of the 8 confused cyclists, 7 (87.5%) committed a traffic violation following their confusion.

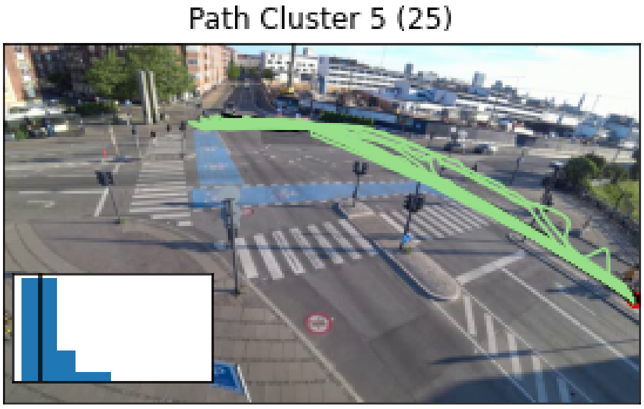
6.3.2 Driving on the Pedestrian Crossing for Alignment



Instead of getting off their bicycles to adjust their direction by turning the back wheel so their bicycle faces in the correct direction, cyclists in path cluster 2 choose to stay on their bicycles while adjusting the direction of it by committing a traffic violation. This also blocks the passing pedestrians since cyclists are now occupying the pedestrian crossing. Sometimes, this happens when there are too many cyclists to properly wait without blocking either cyclists or cars, making it hard to not commit a violation, however, many cyclists still choose this path when there are only a few other cyclists nearby.

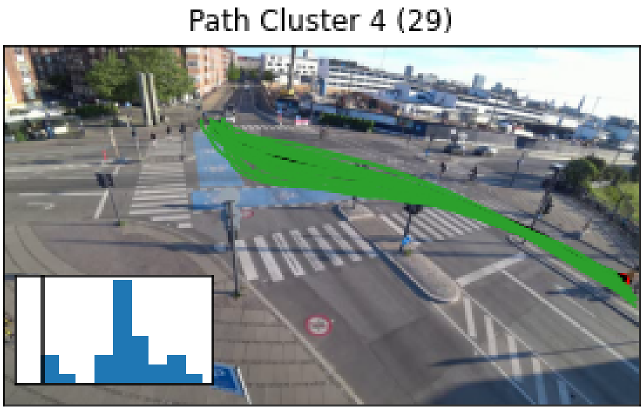
[11]None of the samples are included in the sample video. However, the paths suggest that this was done while mounted on the bike. Supervisor confirms this.

6.3.3 Clockwise Crossing



Crossing the intersection clockwise seems to be done in two ways; a few choose to cross using the pedestrian crossing (presumably dragging their bicycle accordingly), but most cross on bicycle in the opposite direction of the intended (violating the traffic regulations). Even though some might not know that crossing an intersection clockwise on bicycle is a violation of the traffic regulations, it is a peculiar choice, as there is a clear lane marking, a feature that is otherwise highly desirable for many cyclists (Monsere et al., 2019).

6.3.4 Diagonally Crossing



Crossing the intersection diagonally is a clear violation of the traffic regulations. However, many seem to be doing it anyways.

Path cluster 4 has both the highest average speed and the lowest time on screen (see figures 10 and 11). This suggests that the diagonal crossing of the intersection might to be due to the cyclist wanting to spend as little time in the intersection as possible, given that this is the most direct way of crossing in terms of distance covered. However, whether this preference is caused by being in a rush or something else cannot be determined from the data at hand.

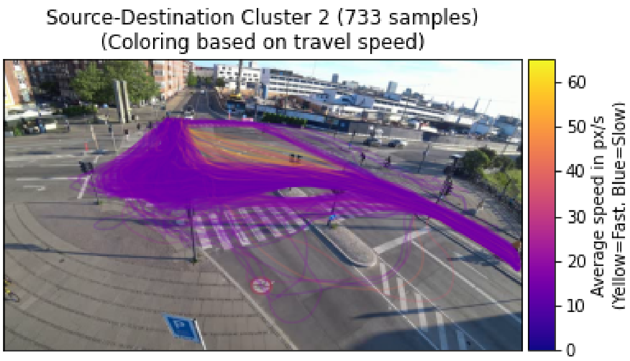


Figure 10: Source-destination cluster 2 colored based on average speed of travel. Yellow indicates a high speed and blue indicates a low speed. Note that the diagonally crossing lines (path cluster 2) are the highest speed, by a large margin.

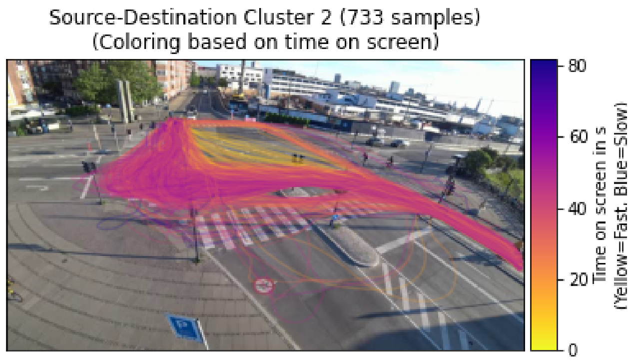


Figure 11: Source-destination cluster 2 colored based on total time on screen. Yellow indicates a short amount of time on screen and purple indicates a long time on screen. Note that the diagonally crossing lines (path cluster 2) are on screen for the shortest amount of time.

7 Proposed Solution by the Municipality of Copenhagen

The Dybbølsbro intersection has proven to not be "tilstrækkelig sikker og tryg [...]" [adequately safe and secure] (Therkildsen, 2021). Especially during morning traffic that "har givet anledning til [...] usikre situationer [og] ulovlig trafikantadfærd [...]" [has caused unsafe situations and illegal behaviour from road users] (Therkildsen, 2021). Because of this, the Municipality of Copenhagen has proposed and approved a new solution (Vesterbro Lokaludvalg, 2021); to construct a diagonal bi-direcitonal bicycle lane diagonally through the intersection (see figure 12):

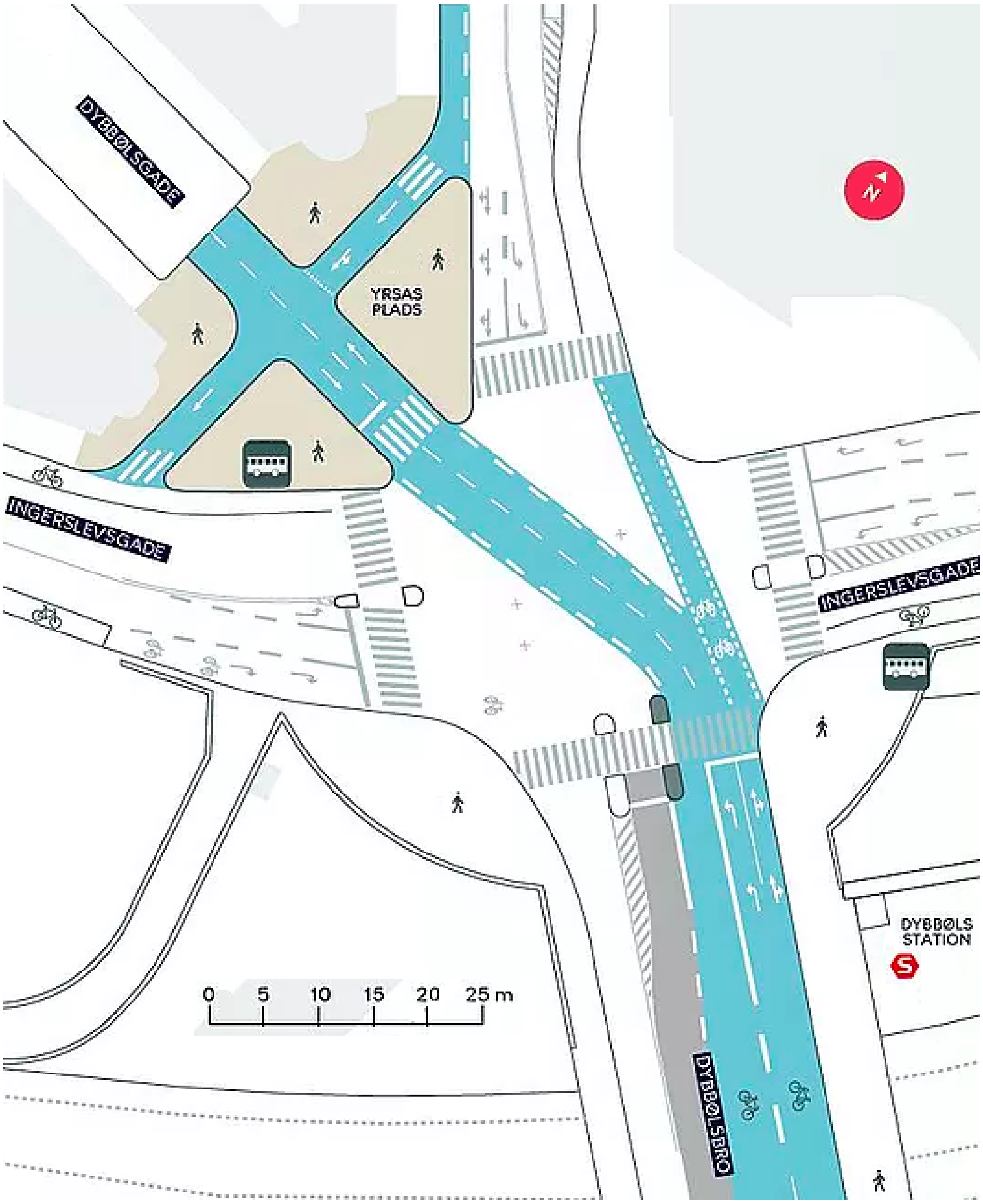


Figure 12: Diagram of the new diagonal bicycle lane proposed by the Municipality of Copenhagen.

Edited; original image source: Municipality of Copenhagen; https://www.kk.dk

This new solution is hoped to improve traffic safety by allowing cyclists to travel from Dybbølsgade (top left) to Dybbølsbro directly and vice versa, decreasing the number of unsafe situations and/or traffic violations in the intersection. The diagonal bicycle lane is especially a good idea given that many violations in the intersection cross this way already (although it is currently a violation; see 6.3 Deeper Analysis of Source-Destination Cluster 2).

7.1 Benefits of the Proposed Solution

The new proposal targets most of the issues identified in this paper and (theoretically) mostly succeeds in handling them:

- **Less Confusion (6.3.1)** The risk of cyclists getting confused and performing illegal maneuvers is much less likely since there is no sudden stop of a bicycle lane. Instead, the new diagonal bicycle lane would likely function similar to the paths taken in the source-destination clusters 7, 8 and 12; following the clearly marked bicycle lane.
- **Less Driving on the Pedestrian Crossing for Alignment (6.3.2)** Cyclists coming from the west and going south will no longer be in a situation where they might feel the need to drive on the pedestrian crossing trying to align their bikes.
- **Less Clockwise Crossing (6.3.3)** Since the proposal creates a clearly marked, direct path through the intersection, it is illogical to take a longer route that is both unmarked and a traffic violation.
- **Diagonally Crossing (6.3.3) is no longer an issue** Since diagonal crossing becomes the intended method of crossing the intersection, it is no longer a traffic violation and the dangers of doing so are significantly reduced.

7.2 Potential new Unaddressed Issues from the Proposed Solution

Despite its many benefits, the proposed solution also comes with potential new unaddressed issues:

- Cyclists coming from the west and going south will no longer have a clearly marked bicycle lane. This creates new potential for confusion. A cyclist coming from the west no longer has any indication that turning right immediately is not the correct maneuver, so they might perform the mistakes described in 6.3.1 Confusion. In fact, some cyclists may even notice that the intersection has been reworked and that the previously clearly marked bicycle lane to 'go straight instead of turning right' has been removed. This would suggest that the intended method to use the intersection has been changed and that you now need to turn as one would expect normally; turn immediately (which is still not intended).
  - It should be noted that the fix for this issue is perhaps quite simple; keep the original (clearly marked) bicycle lane marking that goes from west to south. To reduce the risk of confusion even more, they could even add symbols to the existing marking, similar to the lane markings for buses going straight through a right-turn lane.
- Cyclists coming from the north and going east will have to cross over the lanes of cyclists that are coming from the south (see figure 13). If many cyclists are crossing the intersection at the same time (as mentioned earlier that there is), this might create a bottleneck where cyclists coming from the north and going east must wait for cyclists from the south to pass before they can cross the lane. While they wait, they will block the passage for cyclists coming from the north and going south.
  - This issue could be mitigated with two separate green signal phases for bicycles, but that would take up quite a bit of time in the intersection, where other signals have to be red.

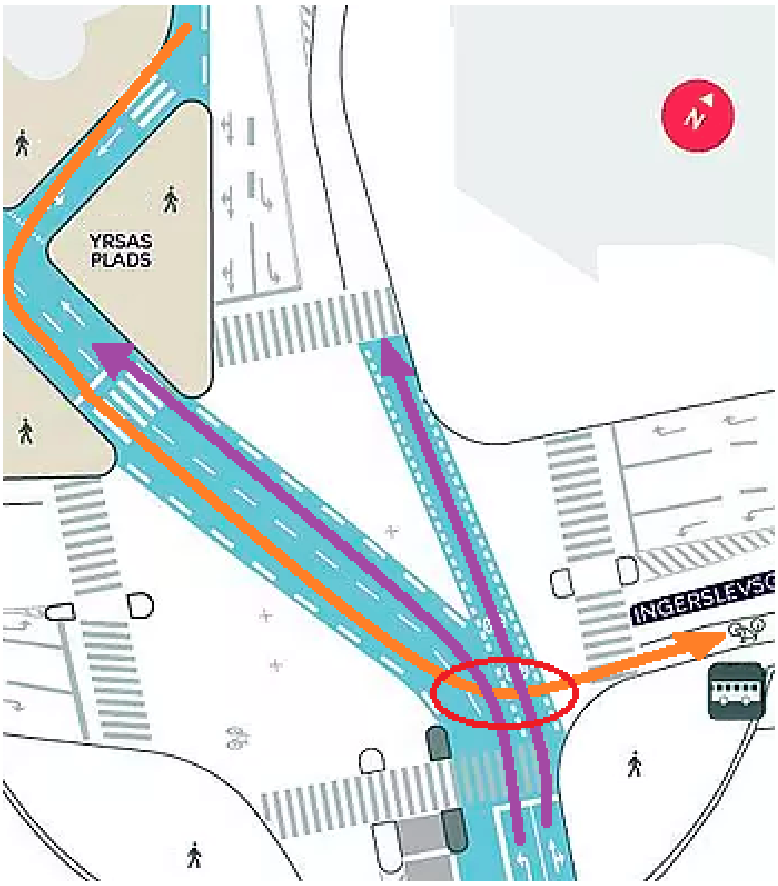


Figure 13: Illustration of the potential new issue originating from the proposed changes; note that in the red circle, the orange line crosses both purple lines. In the case that orange and purple both get green light at the same time, any orange that needs to turn left (following the arrow) would have to wait for purple before crossing, blocking any cyclist that wants to continue straight (onto the bridge).

- A Diagonal crossing has never been seen before in Denmark (Therkildsen, 2021). This might cause cyclists to become confused about if they interpret the intersection correctly.
  - Because the markings are so clear and cannot be interpreted as much else than a bicycle lane, this should not be a problem. However, it should still be kept in mind that anything introduced for the first time should be *extremely* well indicated to avoid confusion.

8 Future Work

As a suggestion to other authors, future works could entail one or more of the following improvements:

- Using the same data set, one could try to repair the broken trajectories and perform the same analyses as this paper. This improvement in data set quality might have an influence on the results, although I find it unlikely that the conclusion changes. I do, however, believe that it would improve the accuracy of the results, given that this would enlarge the data set.
- My results for cyclists running red lights seem to be overinflated. Future works could try to optimize the signal line adjustment, or perhaps even the Conputer-Vision-obtained paths themselves. This would surely create a more accurate representation of cyclists running red lights.
- The main focus of this paper is cyclists, and thus uses primarily cyclist data when performing analyses. For the purposes of extending the findings on this paper, future works could choose to focus on the interactions between bicycles and other types of road users.
- This paper evaluates the solution proposed and approved by the Municipality of Copenhagen. Once the change is done in (expected) December 2022 (Therkildsen, 2021), papers can focus on the actual effects of the solution, provided that they obtain a new data set.

9 Conclusion

This paper used a data science approach to find patterns in behavior amongst cyclists in the Dybbølsbro intersection, focusing on traffic violations and otherwise dangerous situations. This analysis was done in stages, initially using DBSCAN to cluster samples into source-destination clusters such that samples going from and to the same places are grouped together. After initial clustering, each of the source-destination clusters were individually sub-clustered hierarchically using Dynamic Time Warping distance. Furthermore, one source-destination cluster was analyzed in further depth, providing analysis on selected path clusters.
The results of the analysis were used to evaluate the solution to issues at the Dybbølsbro intersection that was proposed and approved by the Municipality of Copenhagen.
Lastly, the proposal was evaluated for potential new unaddressed issues that might be created. These potential new issues were also given proposed solutions.

9.1 Main Finding

The analysis conducted in this paper indicates that the solution to issues at the Dybbølsbro intersection that was proposed and approved by the Municipality of Copenhagen does target the key issues.
The analysis found that the diagonal bicycle lane is especially a good idea given the number of violations this corrects. However, there is potential new unaddressed issues currently not covered by the proposed solution, including added confusion and/or bottle-necking.

9.2 Other Findings

When a cyclist got confused and accidentally went too far, few to none opted to dismount their bicycle and cross as a pedestrian while dragging their bicycle. In the North-to-south direction, 7 out of 8 tried to 'correct their mistake', either by cycling on the pedestrian crossing or by crossing through a median strip.

10 Acknowledgements

The works in this paper were supervised and supported by Bojan Kostic and Michael Szell, who also provided the data. I thank them for their insight and expertise during the process.

References

Bach, P. S. (2018, September). *Københavnsk bro udvides: Får byens bredeste cykelsti [copenhagen bridge gets expanded: Gets the widest bicycle lane in the city].* Retrieved from https://www.tv2lorry.dk/kobenhavn/kobenhavnsk-bro-udvides-far-byens-bredeste-cykelsti

Copenhagenize. (2019). *The 2019 copenhagenize index of bicycle-friendly cities - 01. copenhagen.* Retrieved from https://copenhagenizeindex.eu/about/the-index and https://copenhagenizeindex.eu/cities/copenhagen

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd.* Retrieved from https://www.aaai.org/Papers/KDD/1996/KDD96-037.pdf

Green, K. (2020, March). *5 health and environmental benefits of riding your bike.* Retrieved from https://biofriendlyplanet.com/green-alternatives/5-health-and-environmental-benefits-of-riding-your-bike

Monsere, C., McNeil, N., Wang, Y., Sanders, R., Burchfield, R., & Schultheiss, W. (2019, November). Contextual guidance at intersections for protected bicycle lanes. final report. *NITC-RR-987.*

Müller, M. (2007, January). Dynamic time warping. *Information Retrieval for Music and Motion*, *2*, 69-84. doi: 10.1007/978-3-540-74048-3_4

Nørd News [Nord News]. (2021, April). *Local round-up: City planners reconsider the problematic fisketorvet intersection – again!* Retrieved from https://nord.news/2021/04/07/local-round-up-city-planners-reconsider-the-problematic-fisketorvet-intersection-again

Pettinger, T. (2022). *Stats on cycling through red lights.* Retrieved from https://cyclinguphill.com/reviews/stats-cycling-red-lights/

Pucher, J., & Buehler, R. (2007, December). At the frontiers of cycling: Policy innovations in the netherlands, denmark, and germany. *World Transport Policy and Practice*, *13*, 8-57.

Salvador, S., & Chan, P. (2004, January). Toward accurate dynamic time warping in linear time and space. In (Vol. 11, p. 70-80).

Schubert, E., Sander, J., Ester, M., Kriegel, H. P., & Xu, X. (2017, July). DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. *ACM Trans. Database Syst.*, *42*(3). Retrieved from https://doi.org/10.1145/3068335 doi: 10.1145/3068335

SIGKDD. (2014, August). *2014 SIGKDD test of time award.* Retrieved from http://www.kdd.org/News/view/2014-sigkdd-test-of-time-award

The Danish Ministry of Transportation. (2021, August). *Bekendtgørelse af færdselsloven [announcement of the road traffic act].* Retrieved from https://www.retsinformation.dk/eli/lta/2021/1710

Therkildsen, C. S. N. (2021, March). *Omstridt og berygtet lyskryds får løsning, der aldrig før er set i danmark [controversial and infamous intersection gets a solution that has never been seen before in denmark].* Retrieved from https://www.tv2lorry.dk/koebenhavn/omstridt-og-berygtet-lyskryds-faar-loesning-der-aldrig-foer-er-set-i-danmark

Vesterbro Lokaludvalg. (2021, April). *Orientering om dybbølsbro [proceedings about dybbølsbro].* Retrieved from https://www.kk.dk/dagsordener-og-referater/Vesterbro%20Lokaludvalg/m%C3%B8de-21042021/referat/punkt-13

"Wannesm", "Khendrickx", Yurtman, A., Robberechts, P., Vohl, D., Ma, E., ... Wu, S. (2022, January). *wannesm/dtaidistance: v2.3.5.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.5901139 doi: 10.5281/zenodo.5901139

11 Appendix

Accompanying this paper are the following folders and files. Note that the video and location data will not be publicly available due to privacy concerns, as per request of supervisors:

Folder/filename	Description
• data	← Data folder
– interim	← Processed data folder
∗ df_bike_clustered.pickle	← Cyclist samples clustered using DBSCAN and DTW
∗ df_bike_clustered_w_violations.pickle	← df_bike_clustered.pickle with added columns for violations
∗ signal_line_nonred_crosses.pickle	← Column for signal_lines.pickle with count for times the line was crossed while not red.
∗ signal_line_violations.pickle	← Column for signal_lines.pickle with count for times the line was crossed while red.
– raw	← Raw data folder
∗ signal_lines.pickle	← Adjusted signal lines
∗ signal_lines_true.pickle	← Unadjusted signal lines
∗ signals.pickle	← Signal types (mapping)
∗ signals_dense.csv	← Signal colors
∗ traj_01_elab.pkl	← Original data
∗ traj_01_elab_latest.pkl	← Data with corrected frame numbers. Only frame numbers are used from this.
∗ traj_01_elab_new.pkl	← Data with some corrected frame numbers (currently unused)
• img/clusterplots	← Folder for images not included in paper
– full_src_dst_cluster{N}.png	← All path clusters of source-destination cluster N in the same plot
– src_dst_cluster{N}.png	← All path clusters of source-destination cluster N in different plots plot
• vid/09-06-2021-mpeg4-360p-1h.m4v	← Sample video
• analysis.ipynb	← Analysis code - Jupyter Notebook