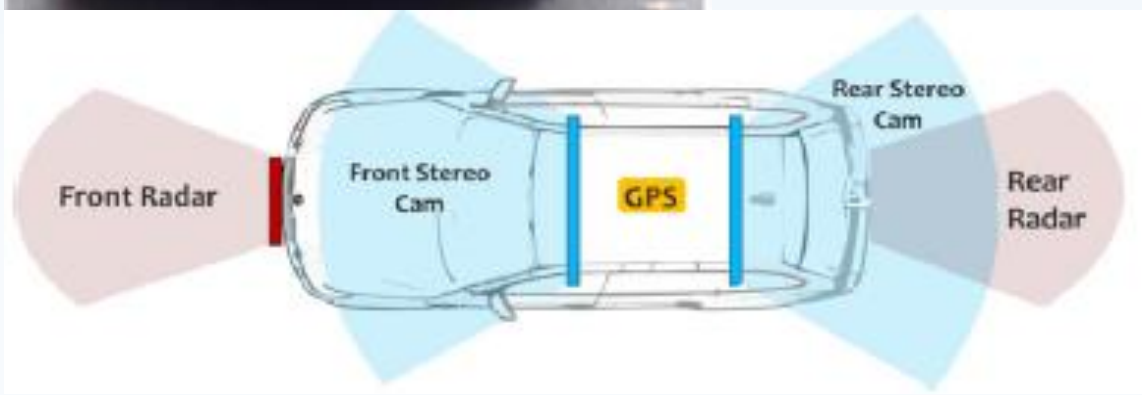
Maneuver Identification in Highway Traffic Using Elastic Template Matching

JOHANNES KEPLER UNIVERSITÄT LINZ

Bachelor Thesis
Christina Schmid

Final Presentation - 24. 10. 2018

Supervision by
Dr. Pavlo Tkachenko



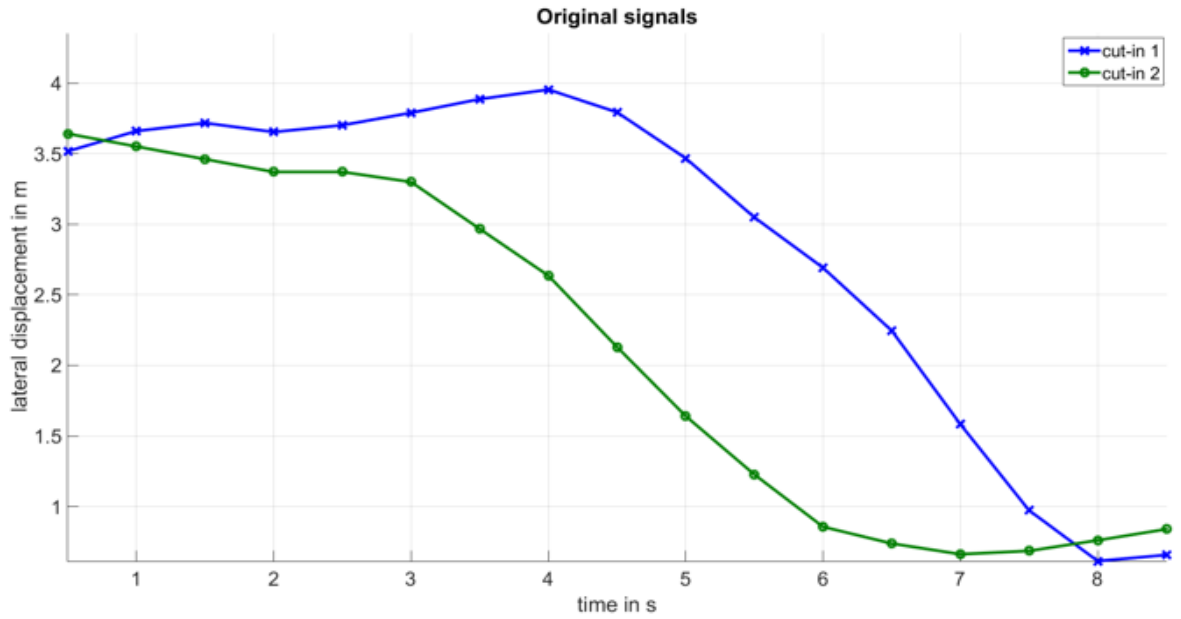
Institute for Design and Control of Mechatronical Systems

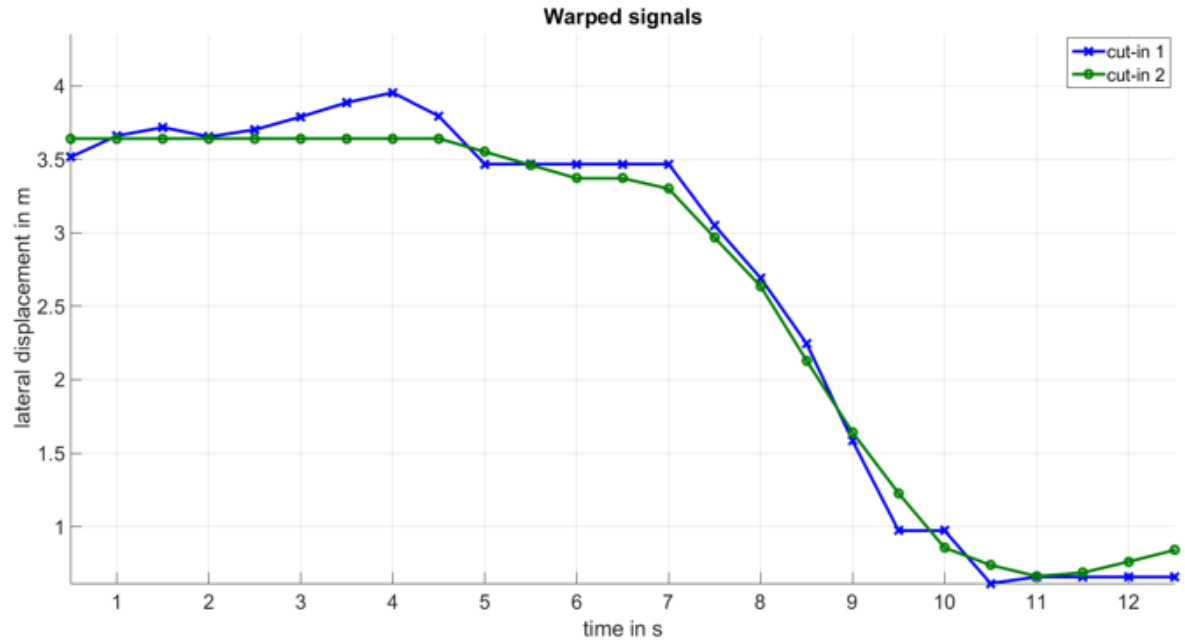
Topic Overview - Goal

Development of a method for a **robust identification** of different maneuvers in highway traffic

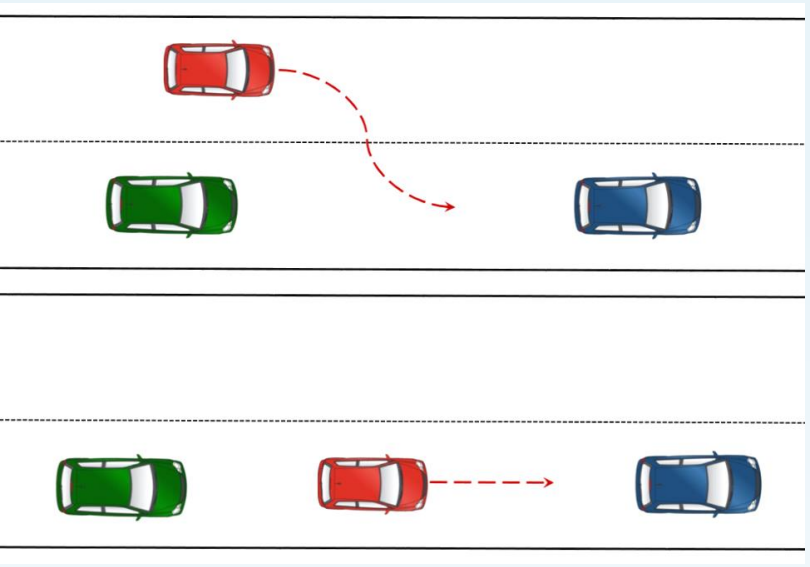
- ➢Based on a special elastic **template matching algorithm**
- ➢**Dynamic time warping (DTW)** method for an optimal non-linear alignment.
- ➢Enables a correct assignment of the same maneuver although :
  - different dynamics (speed, aggressiveness, etc.)
  - data with missing values.

- ➢ Focus only on the most important and basic ones:
  - **cut-in** maneuver
  - **pull-out** maneuver
  - **following** case

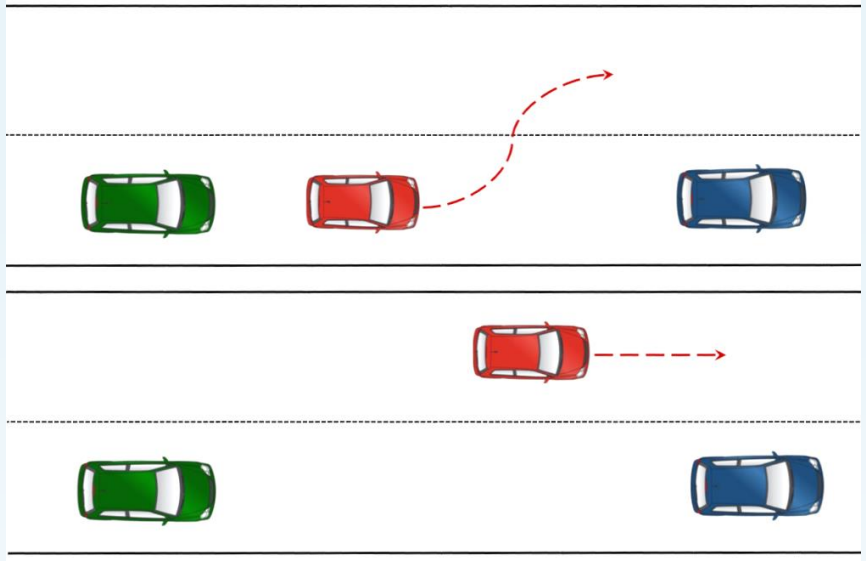




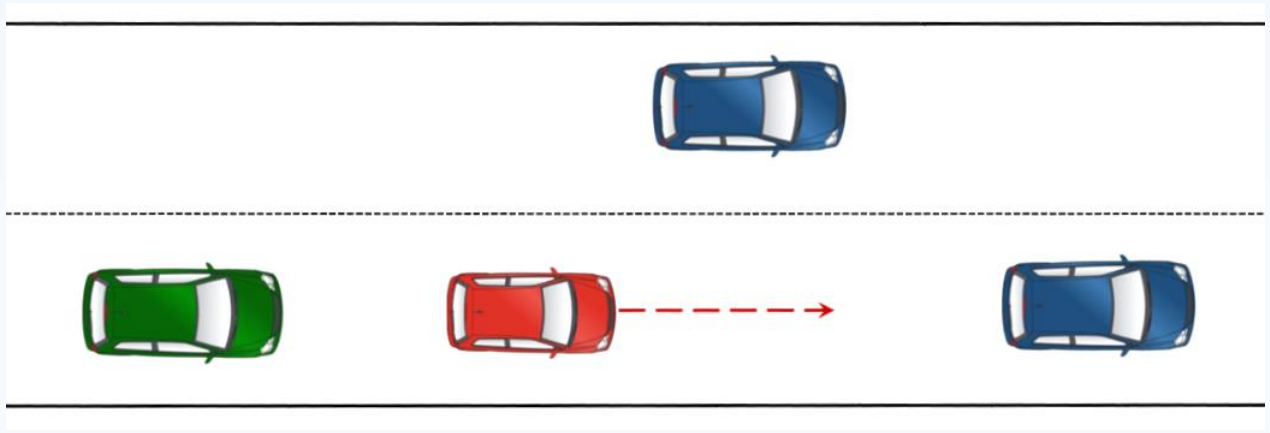
Maneuvers



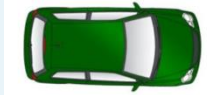
Cut In



Pull Out



Following



Ego car

Vehicle which executes maneuver

Other traffic participant

Template Matching Algorithm

1. Off-line initialization
Select maneuver **template** dX_T

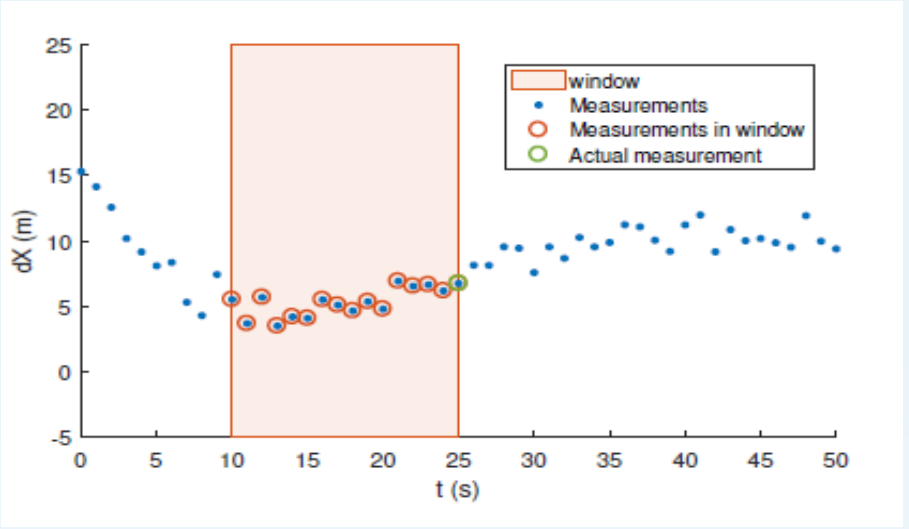
2. On-line data
Collect lateral displacement $dX_i\ (i = 1, \dots, N)$ between ego vehicle and all N surrounding vehicles.

3. DTW alignment
Use DTW method to align dX_T and dX_i

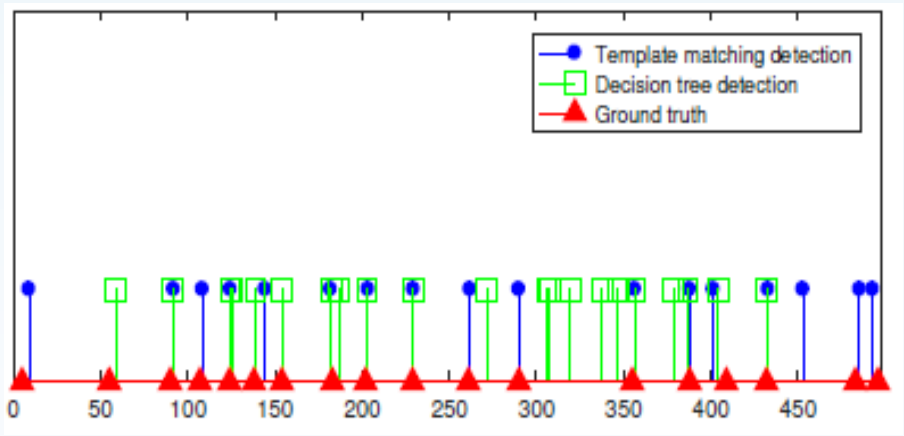
4. Maneuver recognition
Calculate normalized Euclidian distance $D(dX_i^a, dX_T^a)\ /|dX_i^a|$

If there exist $i = i^*$ such that this value satisfys certain conditions related to certain **constant treshold value** (d_1) go to step 5.

5. Identification related to ego vehicle
If the vehicle with index i^* has a **predefined position** around the ego car the maneuver is identified.



Sliding window strategy for extracting dX sequences in the on-line mode



Comparison of decision tree and elastic template matching algorithms with ground truth
- Example Cut-In

DTW Barycenter Averaging (DBA) & Adaptive Scaling (AS)

Pattern = Average of the corresponding training set

- NLAAF average can grow very long (N*T)
  - N (number of sequences), T (length of sequences)
  - DTW complexity increases enormously
- PSA tries to improve NLAAF
  - Possibly information gets lost due to short average

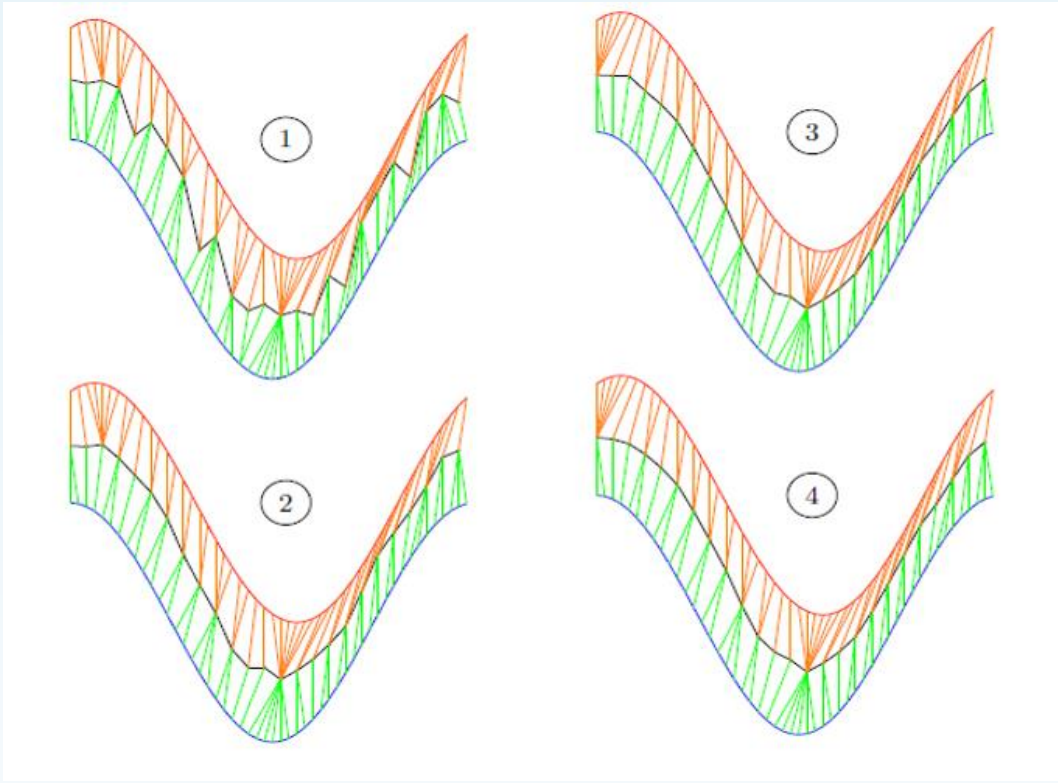
Both are sensitive to the order in which elements are processed

- **DBA** avoids using iterative pairwise averaging
  - Sets a random element of the training set as initial average and gradually refines it
  - Balance : Information/DTW comlexity

Assumption: Window size = length pattern

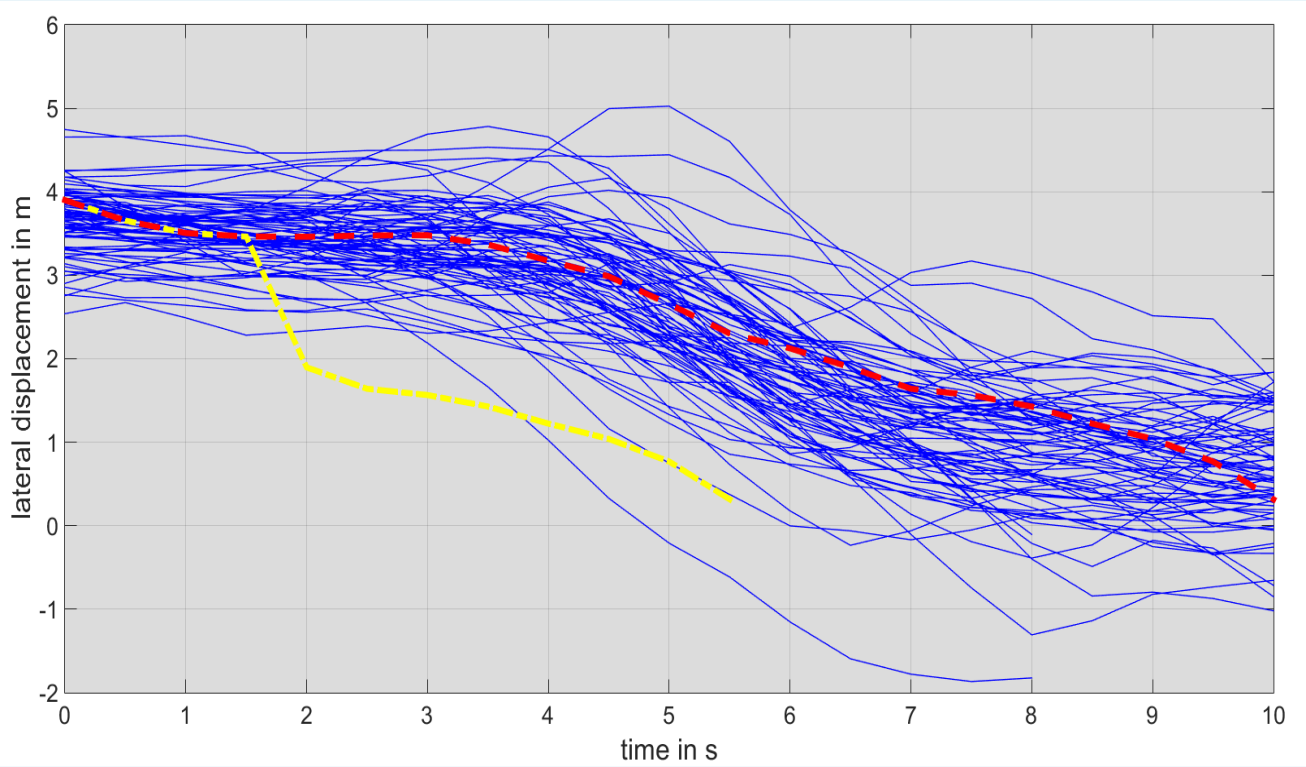
AS : most accurate method to shorten DBA results

- Iterativly combining the two closest elements
- Again Balance : Information/DTW comlexity

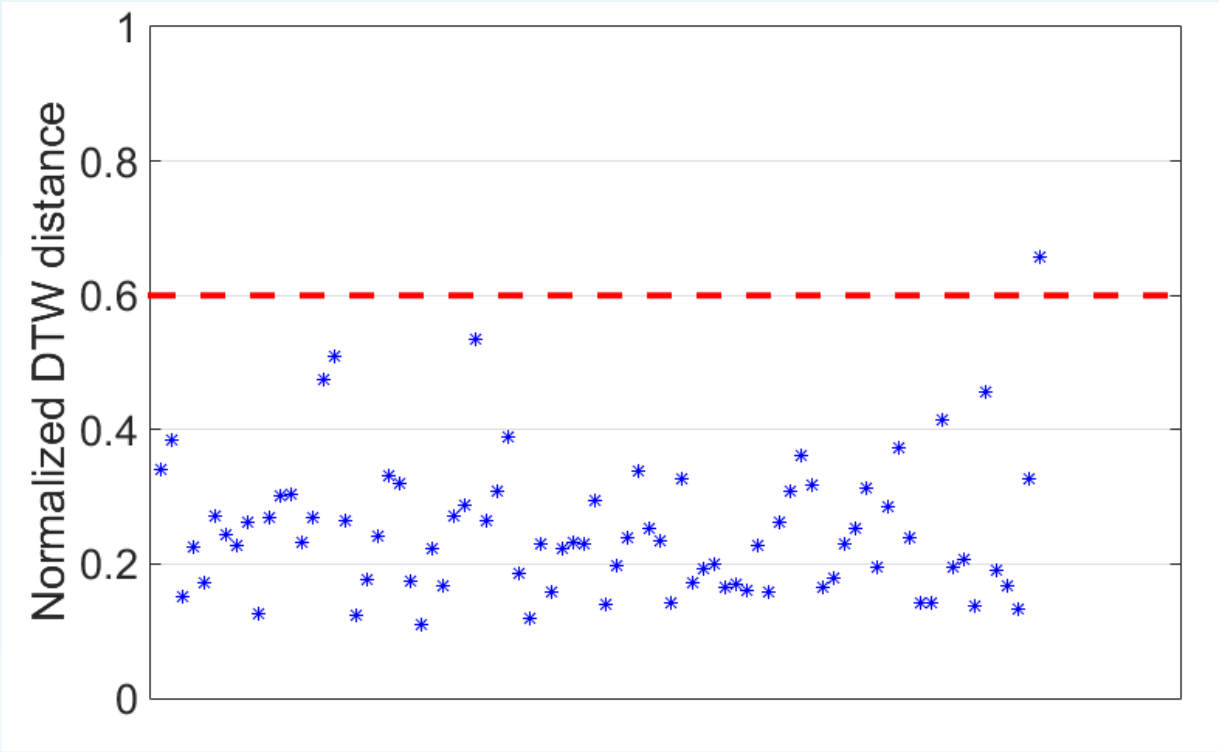


DBA: refining the average of two sequences

Cut In – Training Set, Template, Treshold



Training set (blue) and template using DBA (red), DBA + AS (yellow)



Normalized DTW distance between template and training set $\rightarrow d = \mathbf{0.6}$

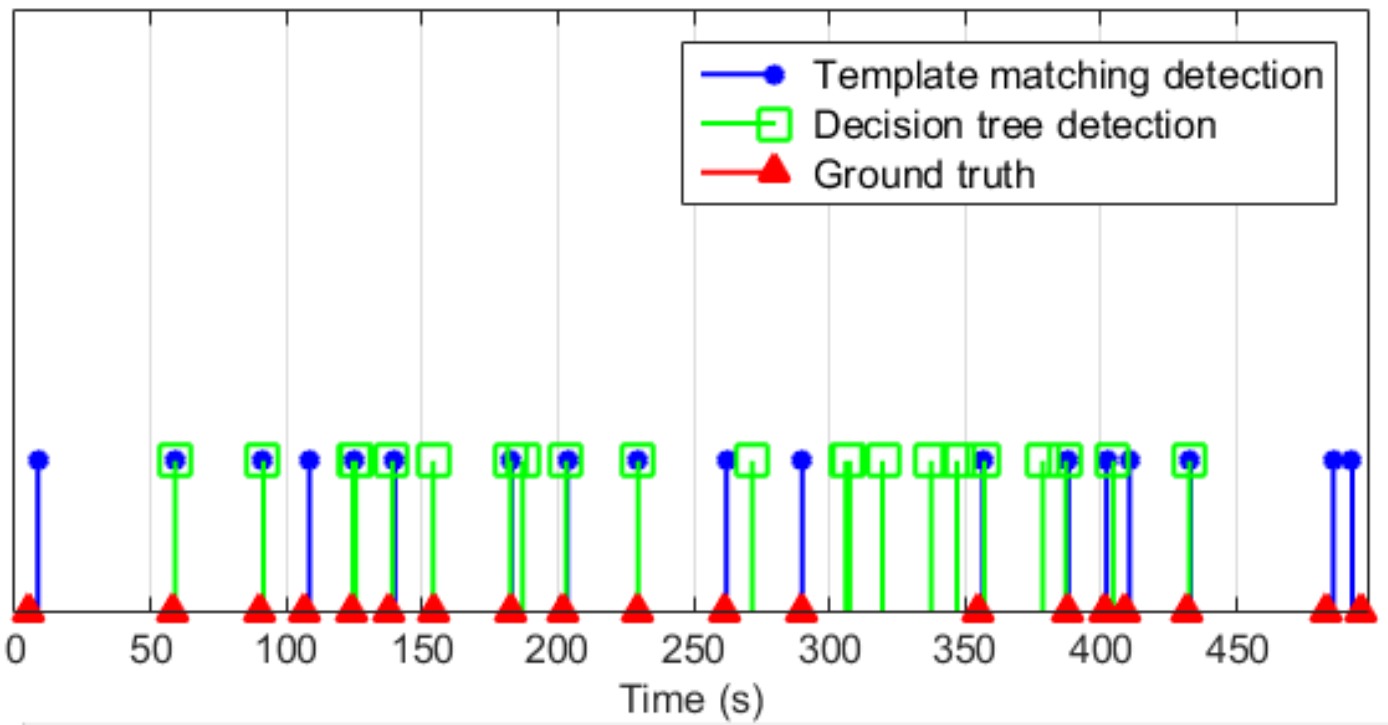
- Collect a training set from the historical data
- Get a *template* dX_T using DTW averaging (DBA) & AS (12 samples)

- Same training set as for the template
- Estimation of treshold d by calculating normalized DTW distances

Cut In – Results

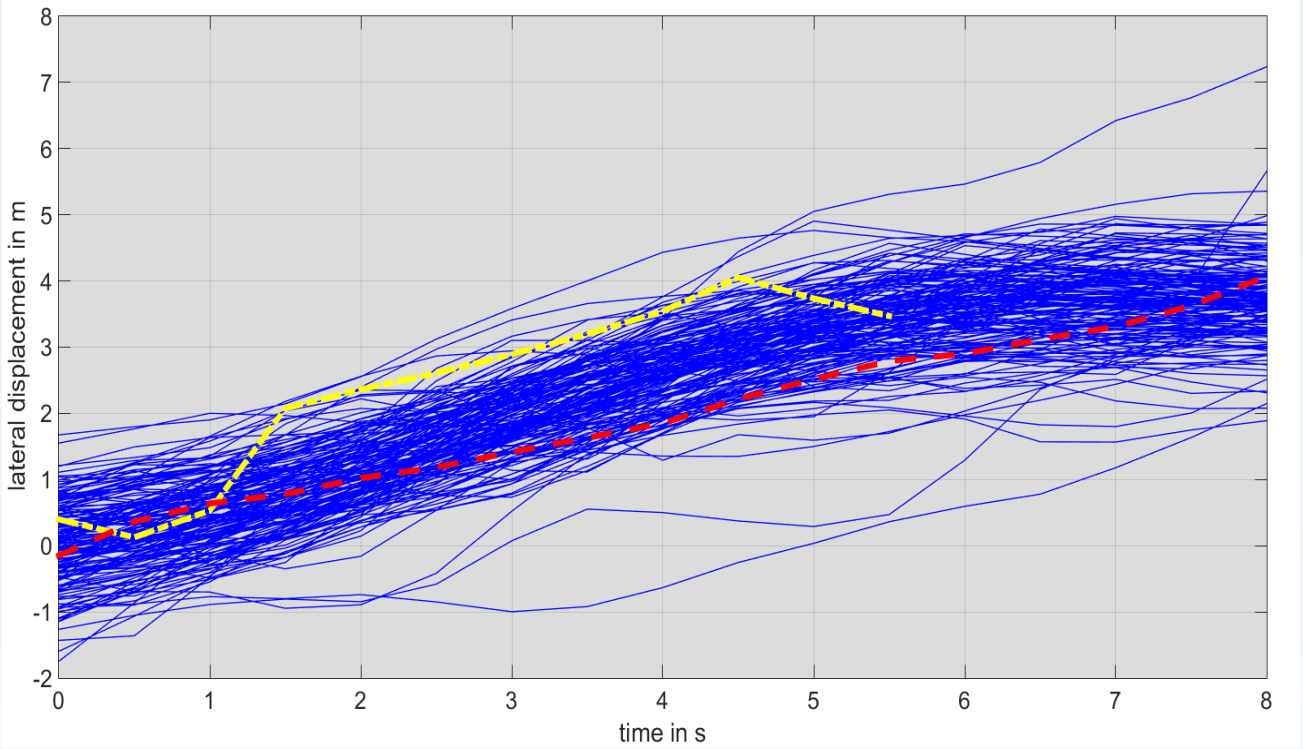
1. Off-line initialization
   - template dX_T
2. On-line data
3. DTW alignment
4. Maneuver recognition
   - constant treshold value (d_1)
5. Identification related to ego vehicle
   - predefined position
     - check if 2 cut-ins within short time

Parameter set	w_s	*block*	*threshold*	TP	FP	FN	F_1-score
b_1	12	3	0,6	15	1	4	0,857
b_2	12	2	0,6	15	1	4	0,857
b_3	14	2	0,6	18	3	1	0,9
b_4	14	2	0,63	18	3	1	0,9
b_5	10	2	0,63	17	1	2	0,918
b_6	10	2	0,62	17	0	2	0,944
b_7	10	1	0,62	18	0	1	0,972

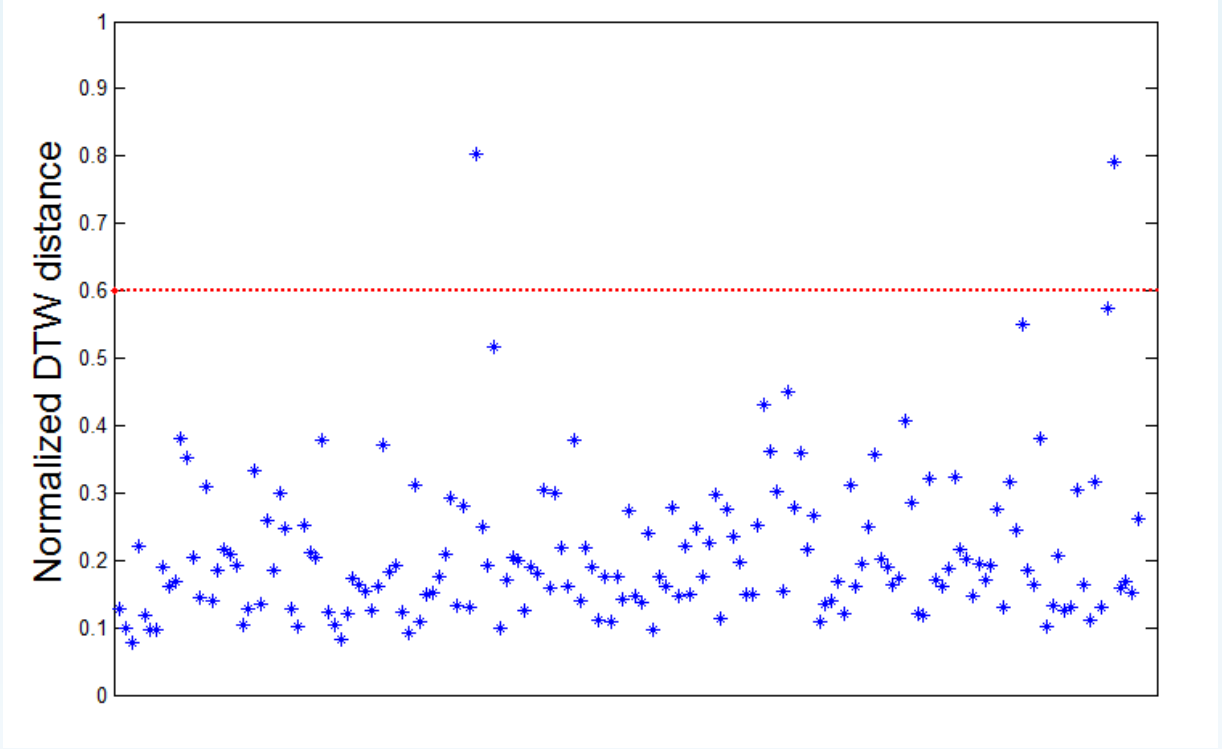


Identification method	TP	FP	FN	F_1-score
DT	12	6	6	0,67
TMA	18	0	1	0,97

Pull Out – Training Set, Template, Treshold



Training set (blue) and template using DBA (red), DBA + AS (yellow)



Normalized DTW distance between template and training set $\rightarrow d = \mathbf{0.6}$

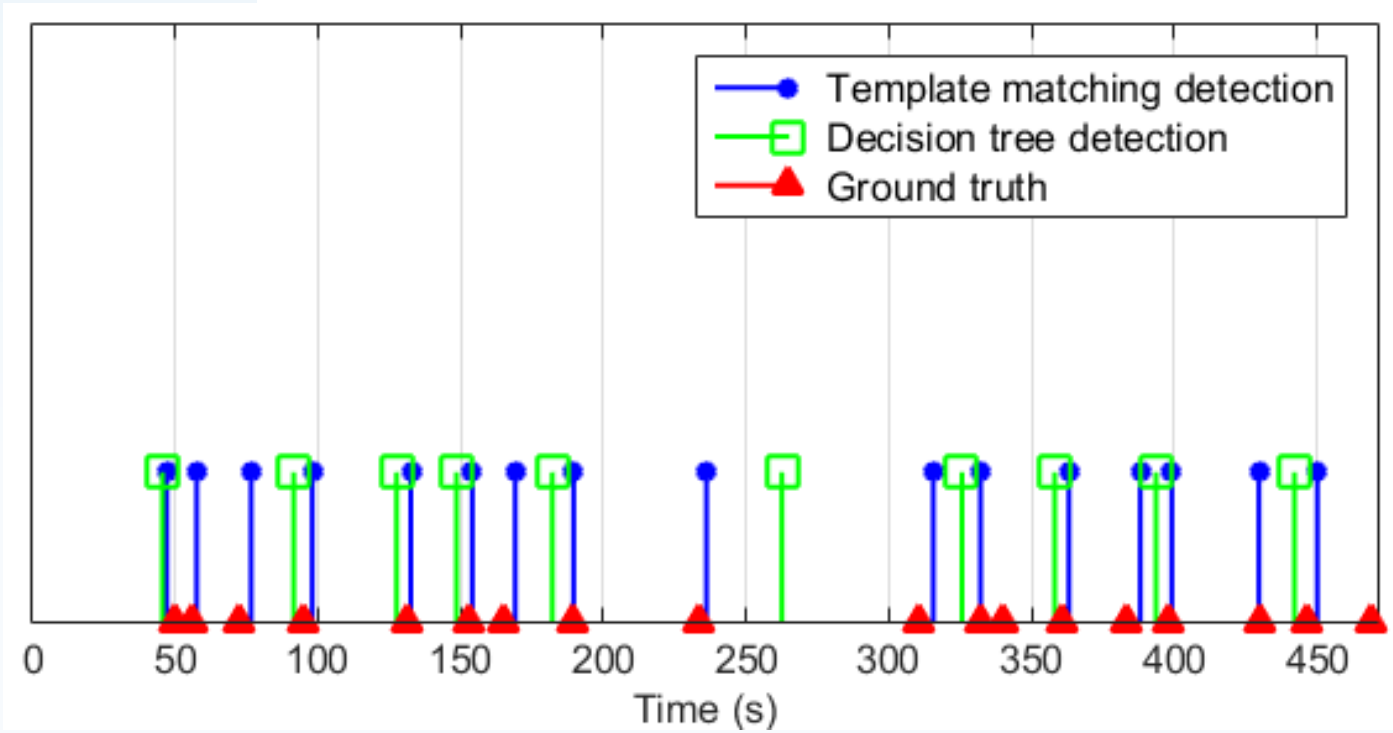
- Collect a training set from the historical data
- Get a *template* dX_T using DTW averaging (DBA) & AS (12 samples)

- Same training set as for the template
- Estimation of treshold d by calculating normalized DTW distances

Pull Out – Results

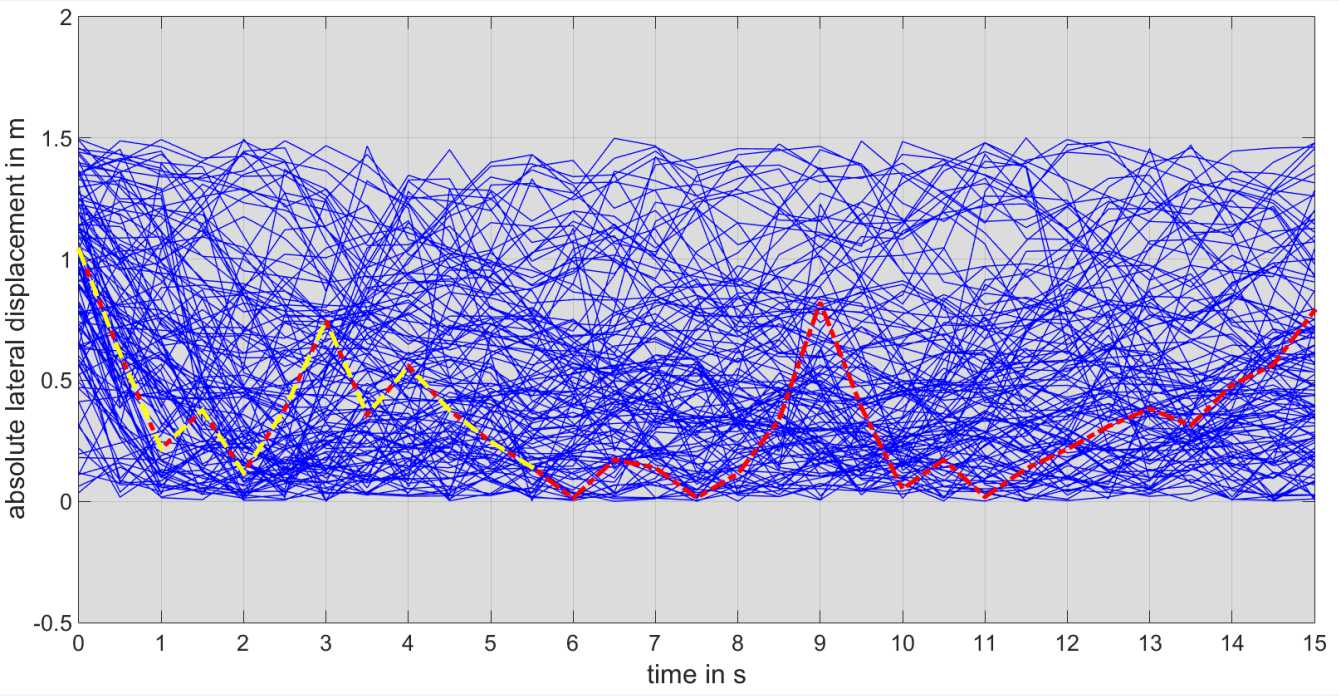
1. Off-line initialization
   - template dX_T
2. On-line data
3. DTW alignment
4. Maneuver recognition
   - constant treshold value (d_1)
5. Identification related to ego vehicle
   - predefined position
     - check if 2 pull-outs within short time

Parameter set	w_s	*block*	*threshold*	TP	FP	FN	F_1-score
c_1	12	3	0,6	12	0	6	0,8
c_2	12	3	0,65	13	0	5	0,827
c_3	12	2	0,65	14	0	4	0,875
c_4	14	2	0,65	14	0	4	0,875
c_5	14	2	0,75	15	0	3	0,909
c_6	14	2	1	16	1	2	0,914
c_7	14	2	0,95	16	0	2	0,941

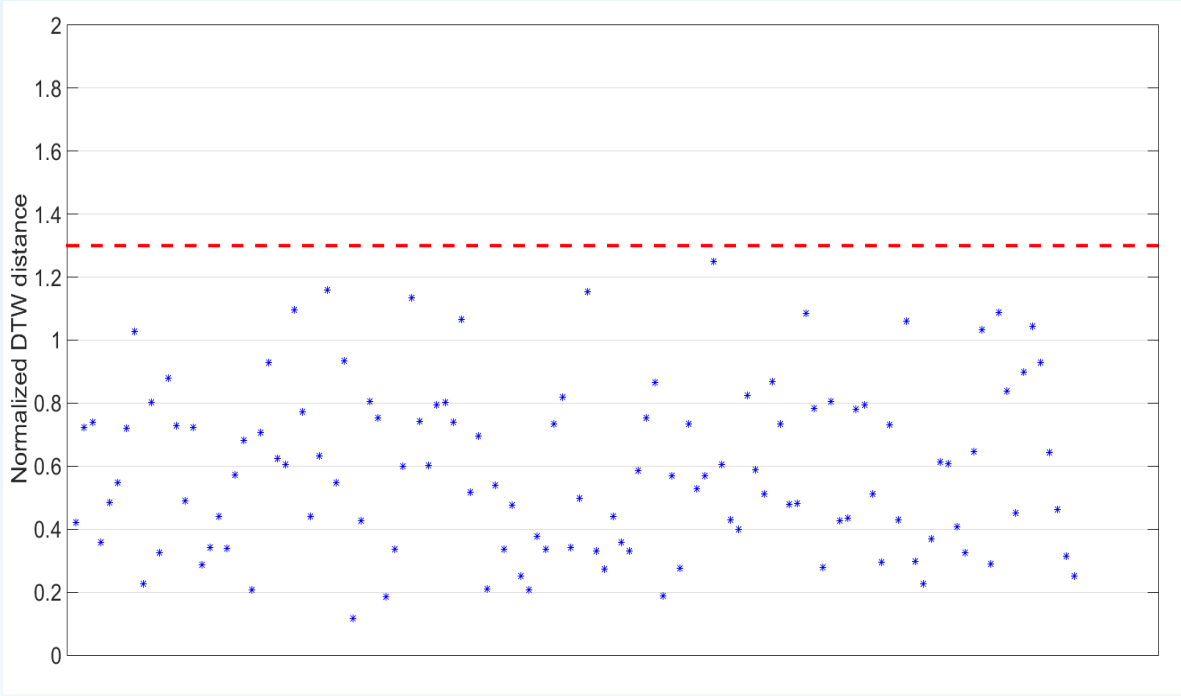


Identification method	TP	FP	FN	F_1-score
DT	12	6	6	0,6
TMA	18	0	1	0,94

Following – Training Set, Template, Treshold



Training set (blue) and template using DBA (red), DBA + cut (yellow)



Normalized DTW distance between template and training set $\rightarrow d = \mathbf{1.3}$

- Collect a training set from the historical data
- Get a *template* dX_T using DTW averaging (DBA) & cut to size 12 (samples)

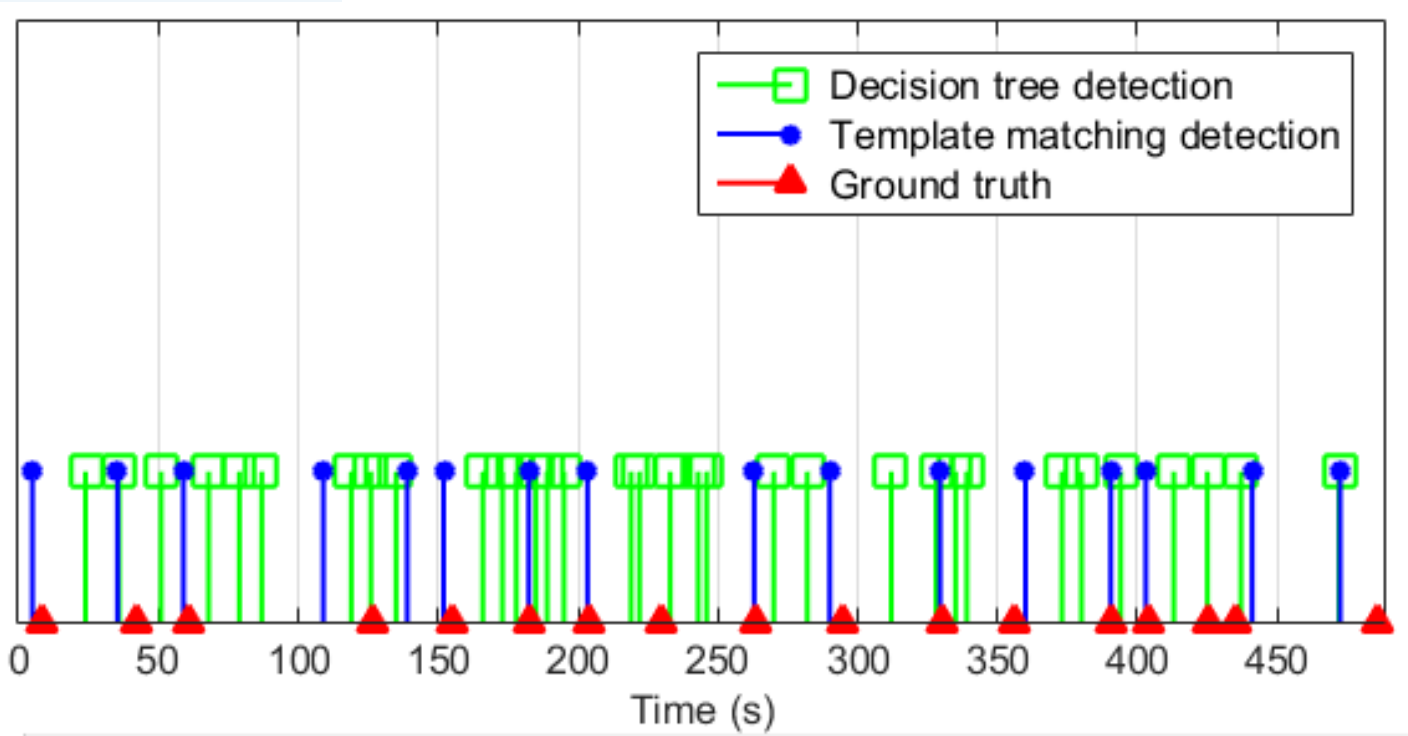
- Same training set as for the template
- Estimation of treshold d by calculating normalized DTW distances

Following – Results

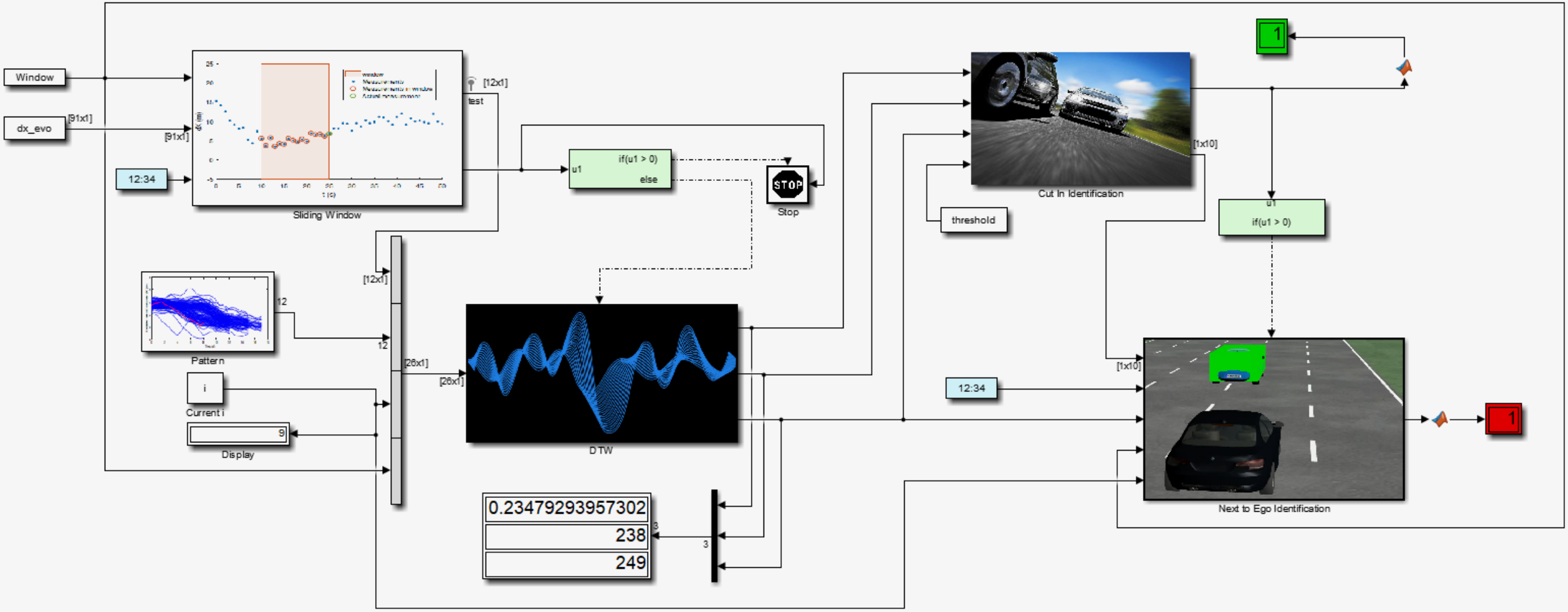
1. Off-line initialization
   - template dX_T
2. On-line data
3. DTW alignment
4. Maneuver recognition
   - constant treshold value (d_1)
5. Identification related to ego vehicle
   - predefined position
     - Hold following state

Parameter set	w_s	*block*	*threshold*	TP	FP	FN	$F_1 - Score$
d_1	12	5	1,3	12	3	5	0,75
d_2	12	5	1,7	13	3	4	0,787
d_3	14	5	1,7	12	3	5	0,75
d_4	14	4	1,7	12	4	5	0,727
d_5	14	3	1,7	13	5	4	0,742
d_6	16	3	1,7	10	6	7	0,606
d_7	18	3	1,7	10	6	7	0,606

Identification method	TP	FP	FN	$F_1 - Score$
DT	8	5	20	**0,39**
TMA	13	3	4	**0,79**



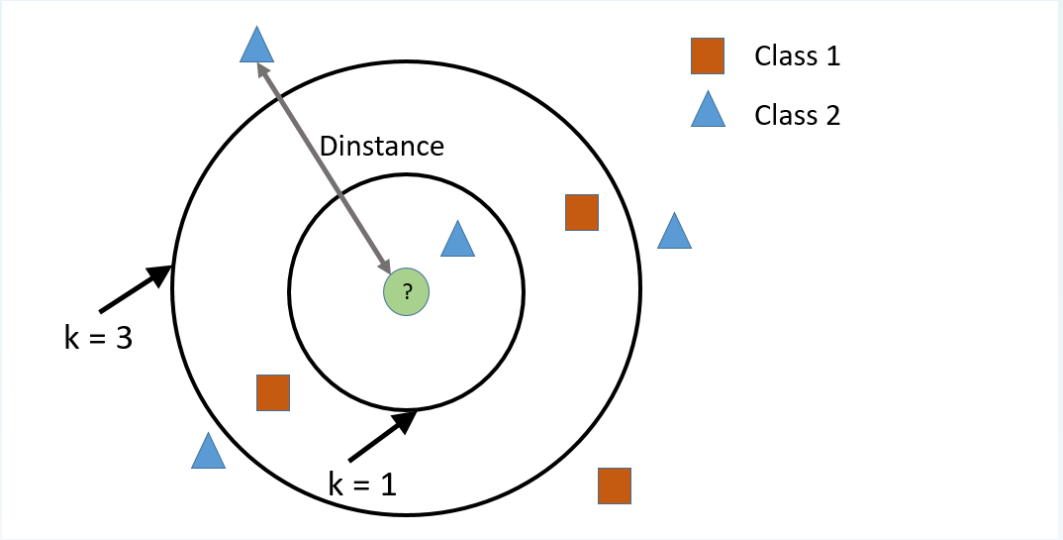
Simulink Implementation



Continuous Regognition - Method

- 1-NN Algorithm
  - Compute distances to training set members
  - Assign sequence to class of nearest element
  - Requires a lot of data -> computationally expensive

- Nearest Centroid Classification
  - Represent training set through centroid
  - Compute distances to centroids
  - Assign sequence to nearest centroid
  - Distance measure: DTW

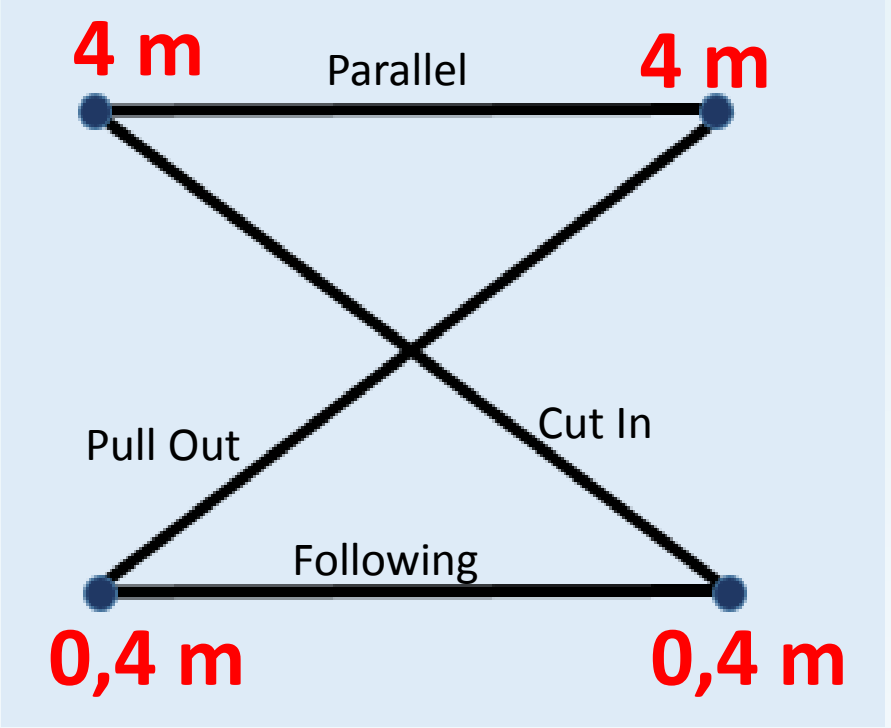


3-NN and 1-NN classification

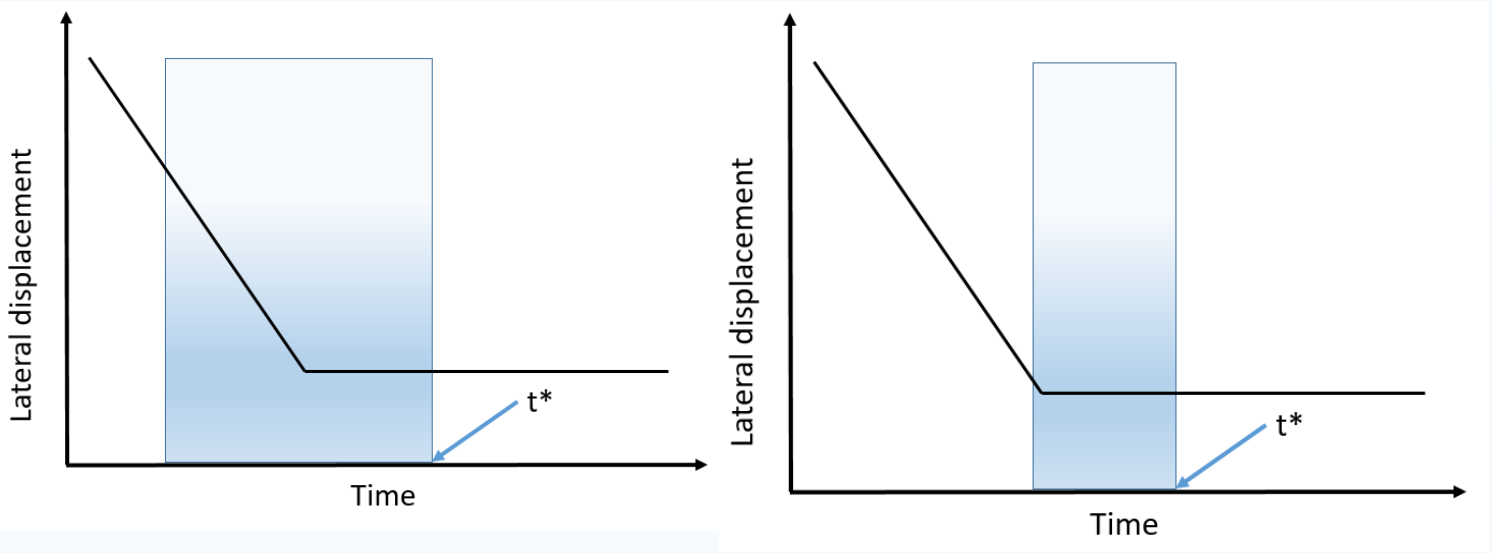
Nearest Neighbors algorithm – Settings, New Maneuver

Principle

- Continuous detection
- Calculate DTW distances between patterns and test sequence
- Select maneuver with minimal distance
- No thresholds
- Synthetic patterns
- New Maneuver: Parallel

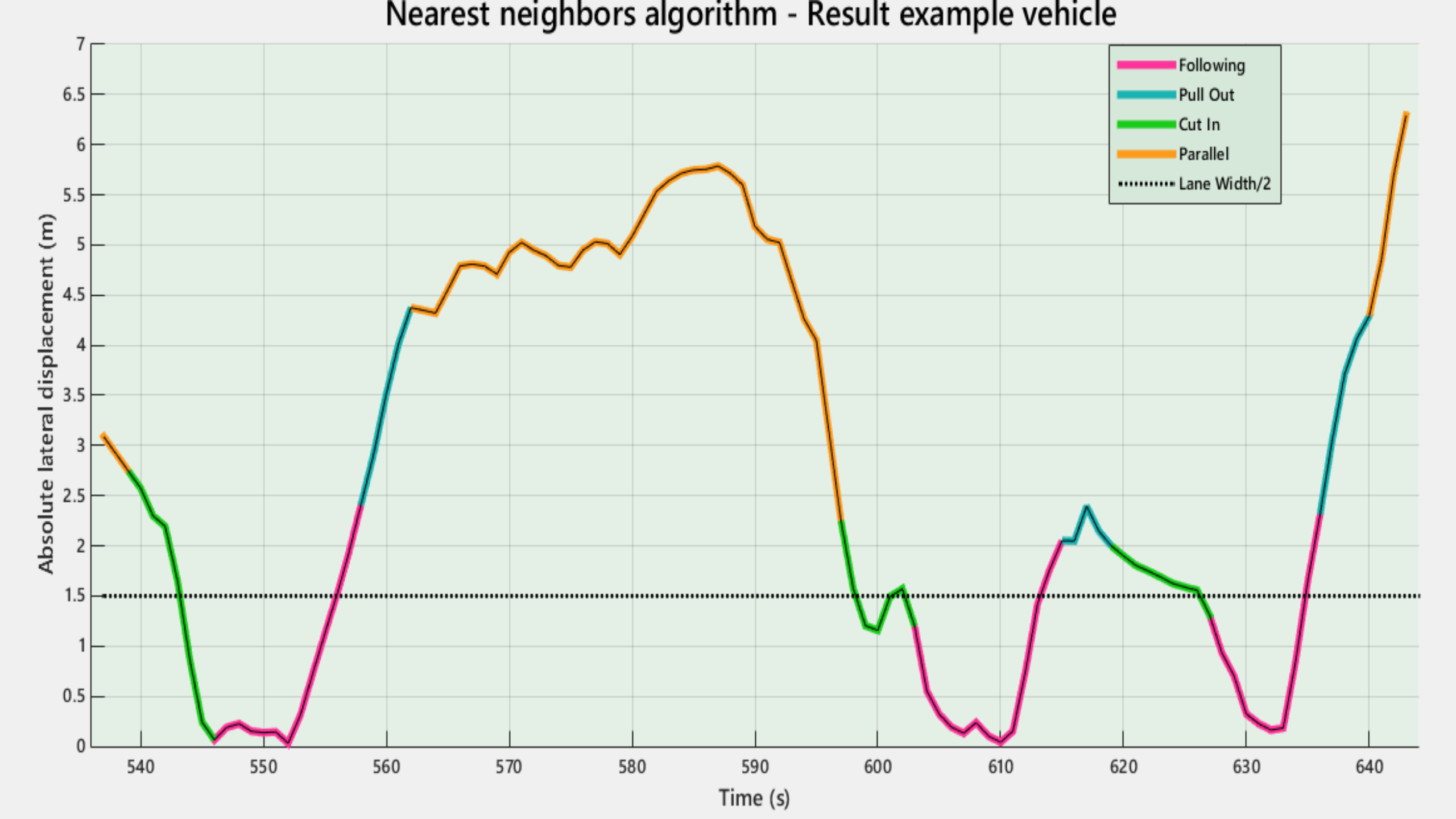


Patterns



Sliding window size	12 -> 6


IPG


Nearest neighbors algorithm - Result example vehicle
Following
Pull Out
Cut In
Parallel
Lane Width/2
Absolute lateral displacement (m)
Time (s)
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
5.5
6
6.5
7
540
550
560
570
580
590
600
610
620
630
640

Conclusion

- Achieved the goal to show that modified TMA is powerful identification method
- DBA & AS are advantegous regarding the pattern extraction
- Centroid classifier enables continous identification

- Promising Approach: SPRING
  - Superfast subsequence search using DTW
  - Increase accuracy and speed