Scene-aware and Social-aware Motion Prediction for Autonomous Driving

Baris Sözüdogru, Alfred Nguyen

Technical University of Munich

January 06, 2024

Agenda

Introduction

Autonomous Driving Promise

- Efficiency and Safety

Challenges in Motion Prediction

- Multimodality
- Scene Dependence
- Social Acceptability

Crucial Understanding

- Human-Driven Behavior Key

Limitations of Current AI Tools

- Control Perspective Absent
- Intent Interpretation Challenge

Autonomous Driving Promise

- Efficiency and Safety

Challenges in Motion Prediction

- Efficiency and Safety

Multimodality

- Scene Dependence
- Social Acceptability

Crucial Understanding

- Human-Driven Behavior Key

Limitations of Current AI Tools

- Control Perspective Absent
- Intent Interpretation Challenge

Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

$$s(k+1) = s(k) + dt \cdot v(k) + \frac{dt^2}{2} a(k)$$
$$v(k+1) = v(k) + dt \cdot a(k)$$

Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

$$s(k+1) = s(k) + dt \cdot v(k) + \frac{dt^2}{2} a(k)$$
$$v(k+1) = v(k) + dt \cdot a(k)$$

Acceleration Equations (Rearranged):

$$a(k) = \frac{2}{dt^2}\Big(s(k+1) - s(k) - dt \cdot v(k)\Big)$$
$$a(k) = \frac{1}{dt}\Big(v(k+1) - v(k)\Big)$$

Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

$$s(k+1) = s(k) + dt \cdot v(k) + \frac{dt^2}{2} a(k)$$
$$v(k+1) = v(k) + dt \cdot a(k)$$

Acceleration Equations (Rearranged):

$$a(k) = \frac{2}{dt^2}\Big(s(k+1) - s(k) - dt \cdot v(k)\Big)$$
$$a(k) = \frac{1}{dt}\Big(v(k+1) - v(k)\Big)$$

Problem: Accelerations are not equal!

Agenda

Agenda

Agenda

Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

$$s(k+1) = s(k) + dt \cdot v(k) + \frac{dt^2}{2} a(k)$$
$$v(k+1) = v(k) + dt \cdot a(k)$$

Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

$$s(k+1) = s(k) + dt \cdot v(k) + \frac{dt^2}{2} a(k)$$
$$v(k+1) = v(k) + dt \cdot a(k)$$

Acceleration Equations (Rearranged):

$$a(k) = \frac{2}{dt^2} \Big(s(k+1) - s(k) - dt \cdot v(k) \Big)$$
$$a(k) = \frac{1}{dt} \Big(v(k+1) - v(k) \Big)$$

Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

$$s(k+1) = s(k) + dt \cdot v(k) + \frac{dt^2}{2} a(k)$$
$$v(k+1) = v(k) + dt \cdot a(k)$$

Acceleration Equations (Rearranged):

$$a(k) = \frac{2}{dt^2}\Big(s(k+1) - s(k) - dt \cdot v(k)\Big)$$
$$a(k) = \frac{1}{dt}\Big(v(k+1) - v(k)\Big)$$

Problem: Accelerations are not equal!

Our Integration Model

Distance and Velocity Equations:

Our Integration Model

Distance and Velocity Equations:

$$
\begin{aligned}
s(k+1) &= s(k) + dt \cdot v(k) + c_1 a(k) + c_2 a(k-1) \\
v(k+1) &= v(k) + c_3 a(k) + c_4 a(k-1)
\end{aligned}
$$

Our Integration Model

Distance and Velocity Equations:

$$
\begin{aligned}
s(k+1) &= s(k) + dt \cdot v(k) + c_1 a(k) + c_2 a(k-1) \\
v(k+1) &= v(k) + c_3 a(k) + c_4 a(k-1)
\end{aligned}
$$

Acceleration Equations:

$$
\begin{aligned}
a(k) &= -\overline{c_1} a(k-1) + \overline{c_2} \big(s(k+1) - s(k) - dt \cdot v(k) \big) \\
a(k) &= -\overline{c_3} a(k-1) + \overline{c_4} \big(v(k+1) - v(k) \big)
\end{aligned}
$$

Our Integration Model

Distance and Velocity Equations:

$$
\begin{aligned}
s(k+1) &= s(k) + dt \cdot v(k) + c_1 a(k) + c_2 a(k-1) \\
v(k+1) &= v(k) + c_3 a(k) + c_4 a(k-1)
\end{aligned}
$$

Acceleration Equations:

$$
\begin{aligned}
a(k) &= -\overline{c_1} a(k-1) + \overline{c_2}\big(s(k+1) - s(k) - dt \cdot v(k)\big) \\
a(k) &= -\overline{c_3} a(k-1) + \overline{c_4}\big(v(k+1) - v(k)\big)
\end{aligned}
$$

$\Rightarrow$ This can be solved using linear regression.

Agenda

Scenario filtering

Scenarios we filtered the dataset with:

- Lane merging

Scenario filtering

Scenarios we filtered the dataset with:

- Lane merging
- Lane exiting

Scenario filtering

Scenarios we filtered the dataset with:

- Lane merging
- Lane exiting
- Entering behaviour

Scenario filtering

Scenarios we filtered the dataset with:

- Lane merging
- Lane exiting
- Entering behaviour
- Exiting behaviour

Video demo of the scenarios

Agenda

Reminder: Our Model - Matrix Form

Acceleration from Distance formula:

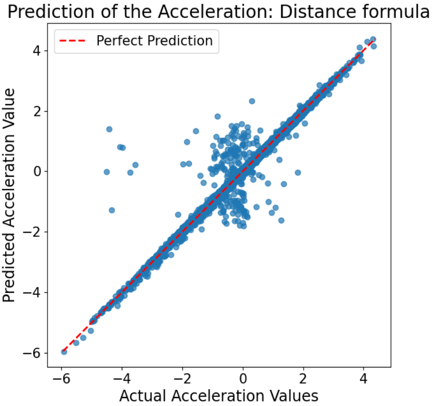
$$\begin{bmatrix} a(k) \end{bmatrix} = \begin{bmatrix} -a(k-1) & v(k+1) - v(k) \end{bmatrix} \begin{bmatrix} \overline{c_1} \\ \overline{c_2} \end{bmatrix}$$

Acceleration from Velocity formula:

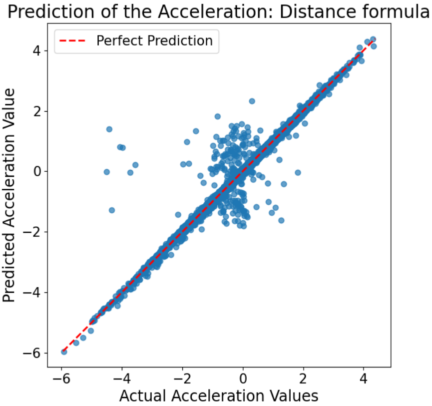
$$\begin{bmatrix} a(k) \end{bmatrix} = \begin{bmatrix} -a(k-1) & s(k+1) - s(k) - dt \cdot v(k) \end{bmatrix} \begin{bmatrix} \overline{c_3} \\ \overline{c_4} \end{bmatrix}$$

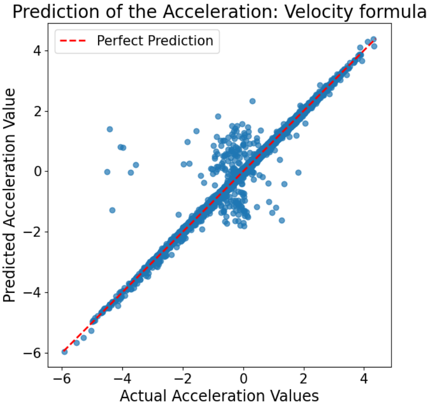
$\Rightarrow$ This can be solved using linear regression.

Results: Integration Method

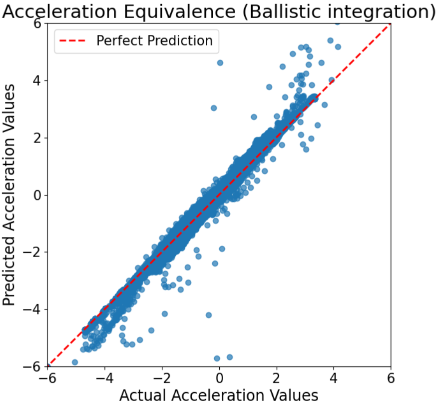


Results: Integration Method



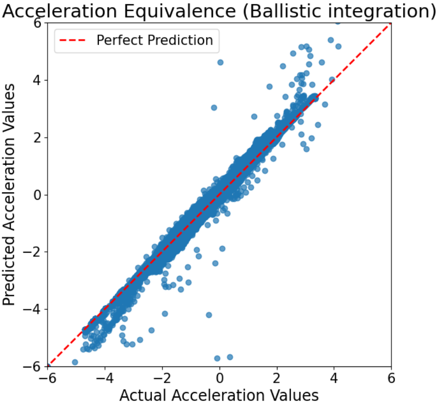


Results: Comparison to the old acceleration model

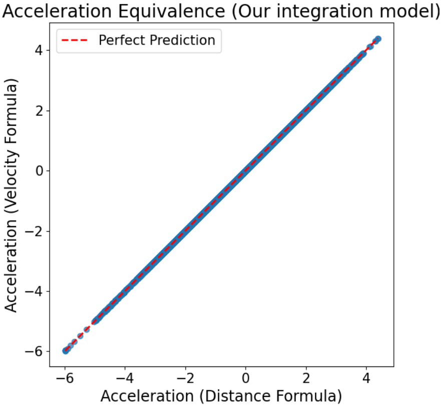


MSE: 3.0786e+02

Results: Comparison to the old acceleration model



MSE: 3.0786e+02



MSE: 1.9220e-09

Results: Integration Method

Rearranging the formula to the distance and velocity gives us these results:

Video demo of predicted car

Results

Summary:

- Successfully implemented the filtering mechanism

Results

Summary:

- Successfully implemented the filtering mechanism
- Able to filter out X different scenarios in Y datasets

Results

Summary:

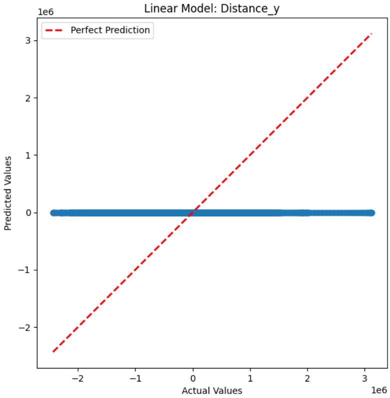
- Successfully implemented the filtering mechanism
- Able to filter out X different scenarios in Y datasets
- Found a better integration method where the accelerations match

Results

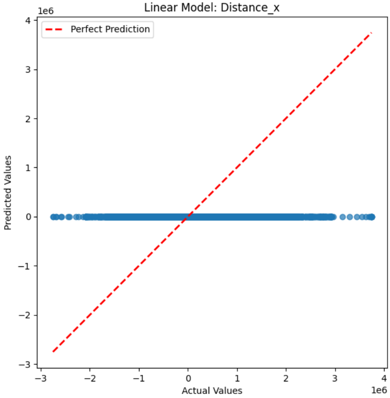
Summary:

- Successfully implemented the filtering mechanism
- Able to filter out X different scenarios in Y datasets
- Found a better integration method where the accelerations match
- Able to visualize the integration method and modulate the movement of a car

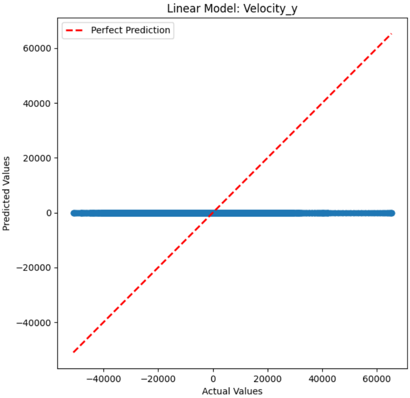
Acceleration Modification in the Y-axis



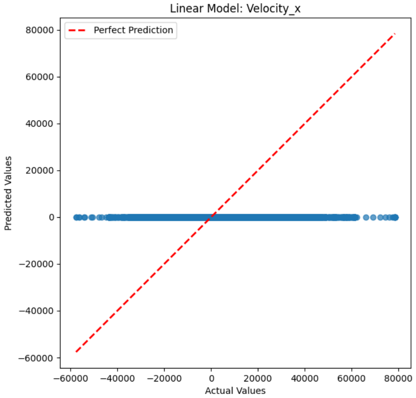
Acceleration Modification in the Y-axis



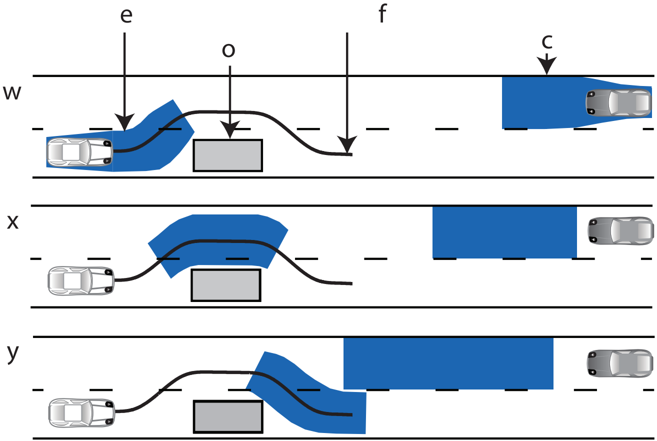
Acceleration Modification in the Y-axis



Acceleration Modification in the Y-axis



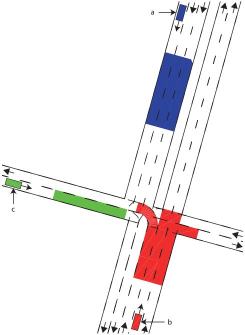
Motivation for Set-Based Prediction [1]



[1] M. Althoff and S. Magdici, "Set-based prediction of traffic participants on arbitrary road networks," IEEE Transactions on Intelligent Vehicles, vol. 1, no. 2, pp. 187–202, 2016.

SPOT

SPOT: A tool for set-based prediction of traffic participants [2]



Initial configuration and $\mathcal{O}(t)$ for $t \in [1.5\,\mathrm{s}, 2.0\,\mathrm{s}]$

[2] M. Koschi and M. Althoff, "SPOT: A tool for set-based prediction of traffic participants," in Proc. of the IEEE Intelligent Vehicles Symposium, pp. 1679–1686, 2017.

Conclusions

- Item
- Item
- Item

beginframe

Distance and Velocity Equations:

$$s(k+1) = s(k) + dt \cdot v(k) + \frac{dt^2}{2} a(k)$$

$$v(k+1) = v(k) + dt \cdot a(k)$$

onslide¡2¿

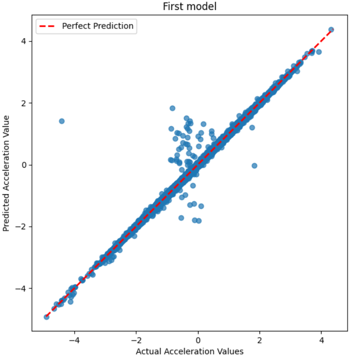
Acceleration Equations:

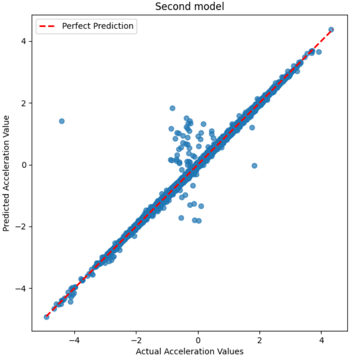
$$a(k) = \frac{2}{dt^2} \Big(s(k+1) - s(k) - dt \cdot v(k) \Big)$$

$$a(k) = \frac{1}{dt} \Big(v(k+1) - v(k) \Big)$$

endframe

Results: Integration Method





Results: Integration Method

Video demo of predicted car

Our Integration Model

Our Distance and Velocity Equations:

Our Integration Model

Our Distance and Velocity Equations:

$$s(t+1) = s(t) + dt \cdot v(t) + c_1 a(t) + c_2 a(t-1)$$
$$v(t+1) = v(t) + c_3 a(t) + c_4 a(t-1)$$

Our Integration Model

Our Distance and Velocity Equations:

$$
\begin{aligned}
s(t+1) &= s(t) + dt \cdot v(t) + c_1 a(t) + c_2 a(t-1) \\
v(t+1) &= v(t) + c_3 a(t) + c_4 a(t-1)
\end{aligned}
$$

Our Acceleration Equations:

$$
\begin{aligned}
a(k) &= -\bar{c}_1 a(k-1) + \bar{c}_2 \big(s(k+1) - s(k) - dt \cdot v(k) \big) \\
a(k) &= -\bar{c}_3 a(k-1) + \bar{c}_4 \big(v(k+1) - v(k) \big)
\end{aligned}
$$

Our Integration Model

Model in matrix form:

$$\begin{bmatrix} a(k) \\ a(k) \end{bmatrix} = \begin{bmatrix} -a(k-1) & s(k+1) - s(k) - dt \cdot v(k) & 0 & 0 \\ 0 & 0 & -a(k-1) & v(k+1) - v(k) \end{bmatrix} \begin{bmatrix} \overline{c_1} \\ \overline{c_2} \\ \overline{c_3} \\ \overline{c_4} \end{bmatrix}$$

Our Integration Model

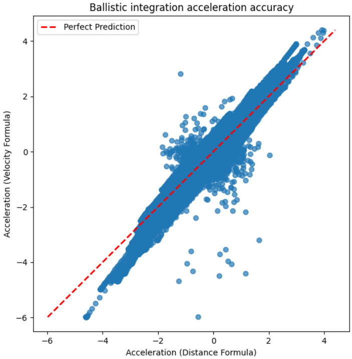
Model in matrix form:

$$\begin{bmatrix} a(k) \\ a(k) \end{bmatrix} = \begin{bmatrix} -a(k-1) & s(k+1) - s(k) - dt \cdot v(k) & 0 & 0 \\ 0 & 0 & -a(k-1) & v(k+1) - v(k) \end{bmatrix} \begin{bmatrix} \overline{c_1} \\ \overline{c_2} \\ \overline{c_3} \\ \overline{c_4} \end{bmatrix}$$

$\Rightarrow$ This can be solved using linear regression.

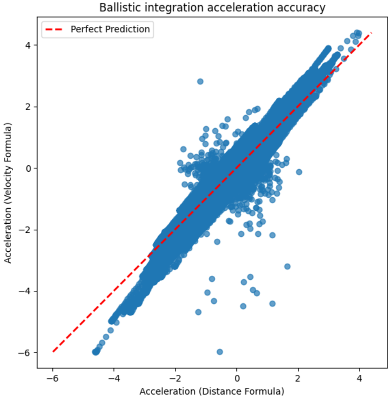
Results: Integration Method

Accuracy of the prediction for the acceleration using the Ballistic Integration method (MSE): 4.3249e-02



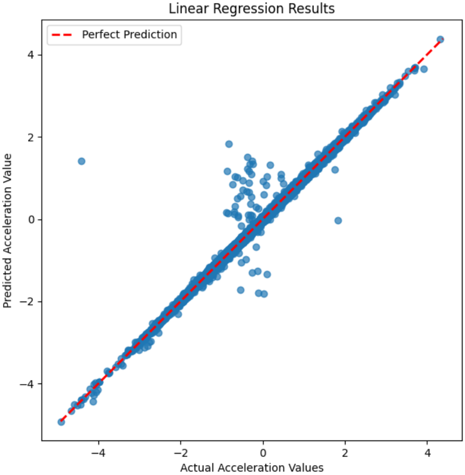
Previous Integration Model - Accuracy

Accuracy of the prediction for the acceleration using the Ballistic Integration method (MSE): 4.3249e-02



Results: Integration Method

Accuracy of the prediction for the acceleration (MSE): 3.0955e-03



Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

$$s(k+1) = s(k) + dt \cdot v(k) + \frac{dt^2}{2} a(k)$$
$$v(k+1) = v(k) + dt \cdot a(k)$$

Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

$$s(k+1) = s(k) + dt \cdot v(k) + \frac{dt^2}{2} a(k)$$
$$v(k+1) = v(k) + dt \cdot a(k)$$

Acceleration Equations (Rearranged):

$$a(k) = \frac{2}{dt^2}\Big(s(k+1) - s(k) - dt \cdot v(k)\Big)$$
$$a(k) = \frac{1}{dt}\Big(v(k+1) - v(k)\Big)$$

Previous Integration Model

Distance and Velocity Equations (Ballistic Integration):

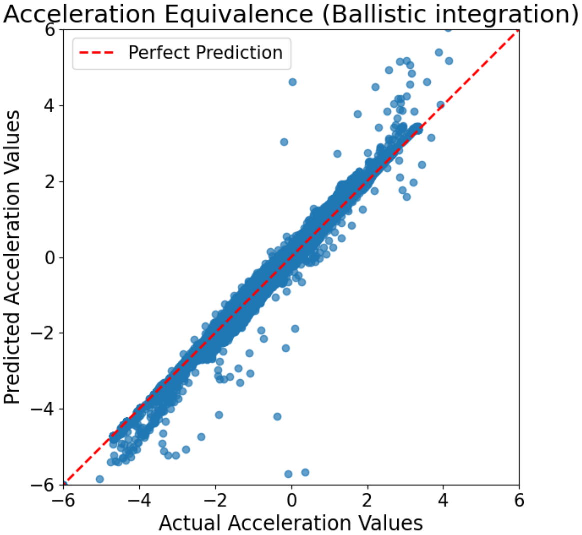
$$s(k+1) = s(k) + dt \cdot v(k) + \frac{dt^2}{2} a(k)$$
$$v(k+1) = v(k) + dt \cdot a(k)$$

Acceleration Equations (Rearranged):

$$a(k) = \frac{2}{dt^2}\Big(s(k+1) - s(k) - dt \cdot v(k)\Big)$$
$$a(k) = \frac{1}{dt}\Big(v(k+1) - v(k)\Big)$$

Problem: Accelerations are not equal!

Previous Integration Model - Accuracy



Our Integration Model - Matrix Form

Acceleration from Distance formula:

$$\begin{bmatrix} a(k) \end{bmatrix} = \begin{bmatrix} -a(k-1) & v(k+1) - v(k) \end{bmatrix} \begin{bmatrix} \overline{c_1} \\ \overline{c_2} \end{bmatrix}$$

Acceleration from Velocity formula:

$$\begin{bmatrix} a(k) \end{bmatrix} = \begin{bmatrix} -a(k-1) & s(k+1) - s(k) - dt \ v(k) \end{bmatrix} \begin{bmatrix} \overline{c_3} \\ \overline{c_4} \end{bmatrix}$$

Our Integration Model - Matrix Form

Acceleration from Distance formula:

$$\begin{bmatrix} a(k) \end{bmatrix} = \begin{bmatrix} -a(k-1) & v(k+1) - v(k) \end{bmatrix} \begin{bmatrix} \overline{c_1} \\ \overline{c_2} \end{bmatrix}$$

Acceleration from Velocity formula:

$$\begin{bmatrix} a(k) \end{bmatrix} = \begin{bmatrix} -a(k-1) & s(k+1) - s(k) - dt\ v(k) \end{bmatrix} \begin{bmatrix} \overline{c_3} \\ \overline{c_4} \end{bmatrix}$$

$\Rightarrow$ This can be solved using linear regression.

Agenda

Future Work

Scenario Filtering:

- Specify even more scenarios for a broader range of use cases.

Future Work

Scenario Filtering:

- Specify even more scenarios for a broader range of use cases.
- Explore other datasets

Future Work

Scenario Filtering:

- Specify even more scenarios for a broader range of use cases.
- Explore other datasets

Integration Model:

- Finetune the integration model (adding other parameteres)

Future Work

Scenario Filtering:

- Specify even more scenarios for a broader range of use cases.
- Explore other datasets

Integration Model:

- Finetune the integration model (adding other parameteres)
- Test the integration model with the neural network for performance (task for the next team)

Questions?