DLAV Project – End-to-End Planner

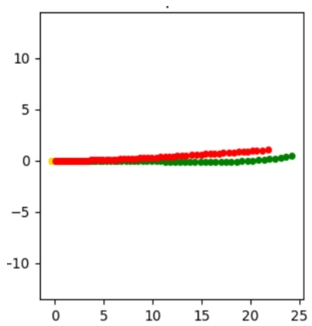
Nael Darwiche, Mohamed Abd El-Razak Bouchouata

Problem Statement:

end-to-end planning model that predicts the future trajectory of an ego vehicle based on:

•**Trajectory history**

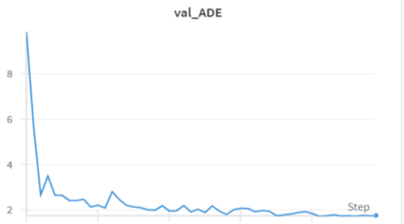
•**RGB camera image** of the current scene



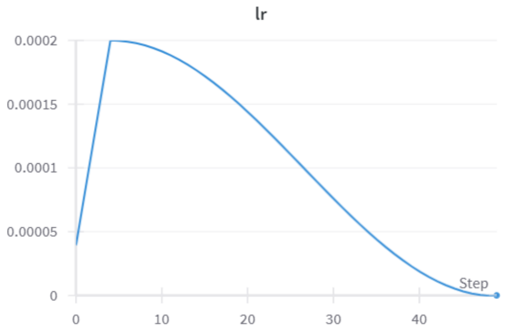
Car departs on green

Methods:

- **Training**:
  - ADE loss
  - (linear) warmup + cosine LR decay
  - Wandb for logging
  - Early checkpointing
  - AdamW optimizer (with regularization), Dropout
  - Data mixing
- **Post-processing**:
  - Butterworth, Elliptic, and Average pooling
- **Augmentations**:
  - **Spatial** are rotation, horizontal flip and (vertical) translation
  - Note: Vertical transformation is "harmless" to the ground truth trajectories, since it only affects the third dimension, which we don't train
  - **Photometric** are Color jitter, gaussian blur, ...
- **Architectures**:
  - **Milestone 1**: (CNN + ViT) **encoder** + Concat.History + MLP **decoder**
    - Variations: Pre-trained CNN (Resnet50), LSTM decoder, MLP encoder for history
  - **Milestone 2**: **Encoder** is Resnet-18 + Transformer decoder; **Decoder** is an MLP
    - Variations: Google's ViT base patch, Swin Tiny, Upsampling network to predict depth map, Additional Transformer decoder (in between CNN encoder, and history transformer decoder



Stable training, mildly unstable validation

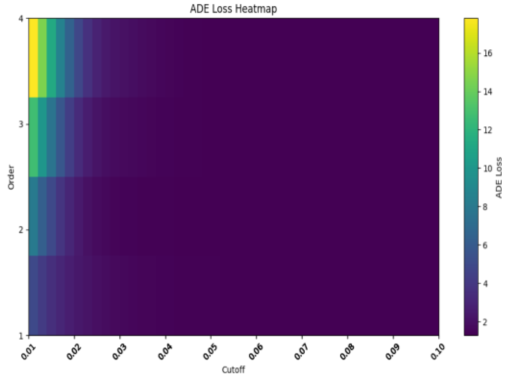


Analysis:

- Affine transformations, namely horizontal flip and rotation are centered around the image's center, while the GT history/future trajectories are centered (the origin) around the ego vehicle's position, which may partially explain the incoherences since we can only apply the transforms on GT trajectories around the ego's position
- ("Manual") Post-processing isn't needed, the model designs its own low-pass filter. Edge effects (e.g. due to padding) are more important for low cutoff frequencies and higher orders, since more coefficients are needed for the filter. Also, a cutoff frequency that is too small may change the trajectory's shape.
- Error accumulation with LSTM
- Extra MLP for history, Resnet-50 (in M1) backbone, or additional transformer decoder (in M2) complexified the optimization problem, thus lowering the ADE; for instance, different components may need different LRs

Results:

- Milestone 1 design, ADE 2.2-2.3
- M1, Resnet-50, LSTM, MLP history encoder worsened ADE
- M2 Google's ViT, ADE ~2-2.1
- M2 Swin Tiny, ADE ~1.75
- M2 Resnet-18, ADE ~1.67
- M2, Additional depth map decoder worsened ADE
- Spatial augmentations incoherences
- Post-processing didn't help (In the graph below, a high cutoff frequency results in minimal filtering)
- Key improvements included adding augmentations, and a pre-trained backbone (especially resnet-18)



ADE loss with respect to (butterworth) filter's cutoff and order, does NOT improve on ADE without filtering