



端到端横向规划

📅 2021 年 3 月 29 日 · ⌚ 阅读时间：8 分钟

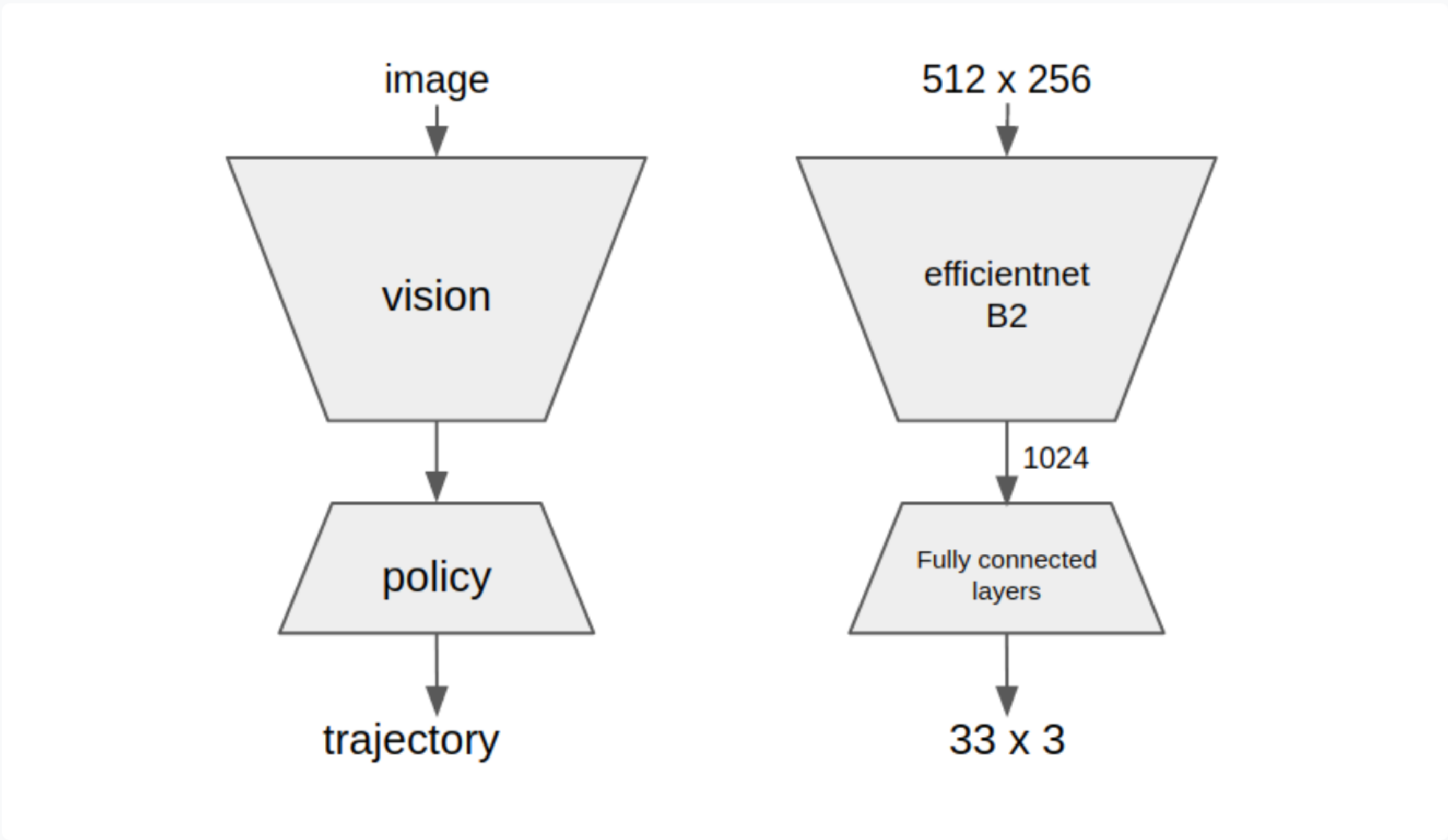
comma 研究团队的目标是打造一个超人驾驶代理。[我们之前讨论过](#)，我们认为打造超人系统的最有效方法是构建一个模仿人类驾驶的端到端规划器。到目前为止，openpilot 严重依赖经典感知来检测车道线并引导汽车做出驾驶决策。最终，我们希望横向和纵向规划能够完全端到端地完成，但我们最近的重点是横向规划。

[openpilot](#)的最新版本0.8.3 包含一个可以进行高质量横向规划的新模型，并且经过了完全端到端的训练。这篇文章将概述这些模型的工作原理以及实现这一点所涉及的挑战。

什么是 e2e 以及它为什么很难？

一个简单的方法

我们称模型为 e2e（端到端），当它接收图像/视频并输出您应该如何驾驶时，在我们的例子中，模型会输出在 3D 空间中行驶的轨迹。该轨迹由 x、y 和 z 坐标（以米为单位）表示，我们将其留给控制器来执行该轨迹。



简单的离线策略 e2e 规划模型图

我们可以想到的最简单的方法是收集大量人类驾驶数据，然后训练一个模型，该模型接收视频并预测人类在每种情况下的驾驶轨迹。我们使用一种MHP 损失，以确保模型可以预测多种可能的轨迹。我们从一个全卷积高效网络 B2块开始，该块减少到 1024 维向量，我们将该块称为视觉模型。然后，该特征向量进入一个包含几个完全连接层的块，我们将其称为策略模型。



简单方法模型的轨迹预测示例

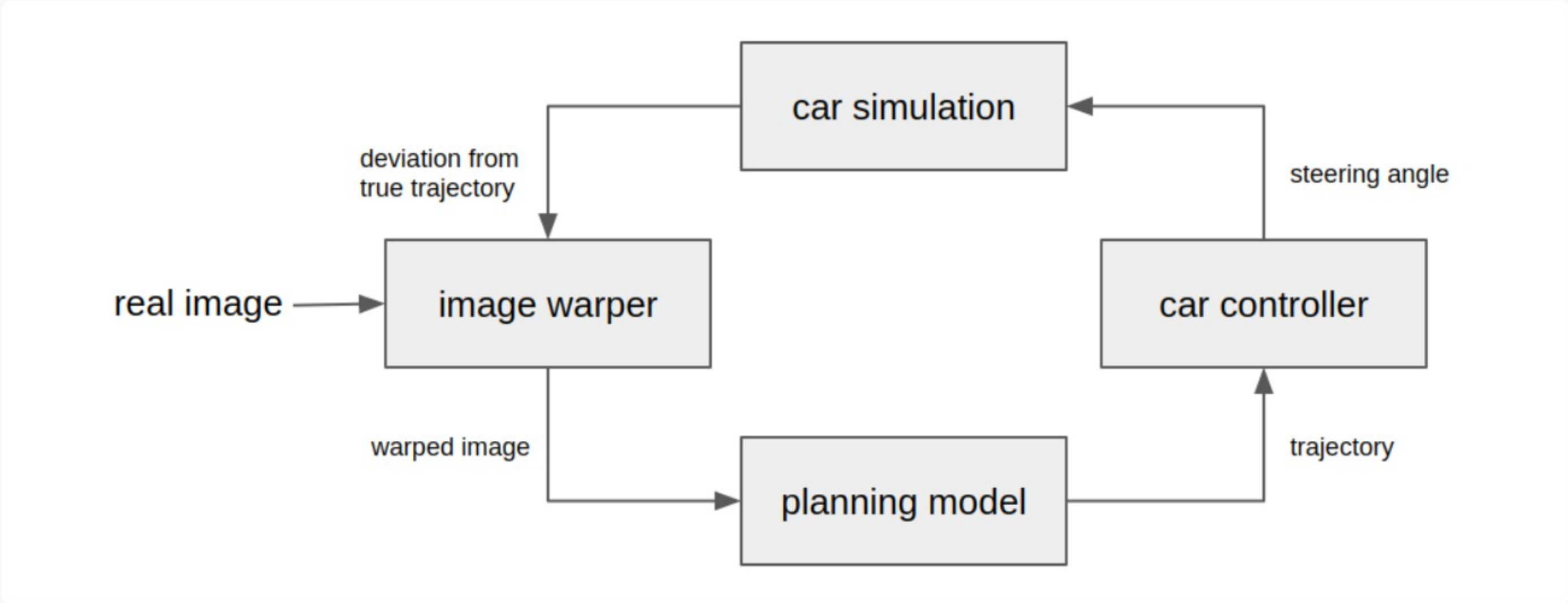
模拟器

在现实世界中测试事物是否有效效率相当低，因此我们使用模拟器。我们使用模拟器接收现实世界的视频并计算 openpilot 在特定模型下的行为方式。该模型预测您应该驾驶的位置，控制堆栈计算实现该轨迹的理想转向角度，然后一个简单的物理模型计算汽车在该转向角度下将如何移动。



在模拟器中移动

经过一次这样的循环迭代后，现在需要修改输入视频以反映汽车的运动。这是最困难的部分。你可以在 CARLA 这样的模拟环境中运行它，引擎允许你随意移动车辆并为你提供新的摄像机视角。然而，这会限制测试的多样性，使之仅限于 CARLA 可以模拟的范围，我们将无法发现许多潜在的真实世界故障。相反，我们选择使用真实世界的视频并对其进行增强，使其看起来好像汽车在按照我们想要的方式移动。有很多方法可以做到这一点，但我们只是对图像应用一个简单的扭曲，假设整个场景都在道路平面上。对于任何大幅度的运动，这都会严重扭曲图像，但对于小偏差，这种方法效果很好。



我们的扭曲模拟器图

问题

现在我们有了一个模拟器，我们可以测试这个简单方法的效果。正如您在下面的视频中看到的，它基本正确地预测了人类驾驶的位置，但它仍然不能很好地保持在正确的轨道上。当我们引入一些轻微的现实世界噪音（转向滞后、风等）时，情况会变得更糟。



使用简单模型驾驶且不添加噪音（左）和注入真实噪音（右）

这里究竟发生了什么值得一谈，但我们将尝试通过一个简单的思维实验来阐明这一点。上面描述的简单模型可以根据一些驾驶视频预测人类最有可能在哪里驾驶。想象一下一条长而直的高速公路车道，其中一个人完全位于车道的中心。人类最有可能保持完全直线行驶，这就是我们的模型将预测的。但是，假设人类出于某种原因距离中心 10 厘米，那么人类现在最有可能的轨迹是什么？它仍然是完全笔直的。

如果你接受这个说法，你很快就会发现，一个只能预测人类最可能轨迹的模型并不能预测如何从错误中恢复。如果汽车因任何原因偏离理想轨迹，模型不会预测恢复到人类轨迹的轨迹。因此，当有任何噪音导致汽车偏离计划轨迹时，就没有恢复的压力，汽车就会继续偏离理想轨迹。在实践中总是有噪音的，即使在我们不引入噪音的模拟中，也存在线性化误差、模型预测误差、舍入误差等.....

我们的方法

在模拟器中学习

为了解决这个问题，我们需要找到一种方法来训练模型，使其能够预测如何从偏离良好轨迹的任何偏差中恢复。一种方法是使用前面描述的模拟器。我们可以引入噪音，让汽车偏离理想轨迹，然后我们可以训练一个模型来从这个新角度预测人类轨迹的位置。就我们的目的而言，我们始终认为人类轨迹是理想的。



使用简单模型的预测轨迹（白色）和人体轨迹（黑色）在嘈杂的模拟中漂移

您可以在视频中看到实际效果，我们让汽车四处漂移，但向模型展示理想轨迹，无论其自身位置如何。让我们将之前的思想实验应用到这个模型上。如果汽车距离车道中心 10 厘米，模型现在将预测更接近车道中心的轨迹。如果我们让控制器在该轨迹上执行，它就会恢复。



使用在嘈杂模拟器中训练的模型在嘈杂的模拟中驾驶，它驾驶得很好

作弊者，一个新问题

上面的解决方案似乎有效，但它有一个隐藏的新问题。它在模拟中有效，但仅限于模拟。我们的模拟方法只是扭曲图像，使汽车看起来像是在横向移动。这种方法引入了很多伪影。



明显的模拟器瑕疵

我们认为理想轨迹就是人类实际驾驶的轨迹。这意味着在模拟过程中，当你走在理想路径上时，图像就不会扭曲。因此，要找出理想轨迹，模型只需学习图像扭曲的程度和方向。对于模型来说，学习扭曲伪影并查看图像如何扭曲要比学习人类在给定场景中会如何驾驶容易得多。因此，最终结果是，如果视频看起来没有扭曲，模型就会预测你走在理想轨迹上。当你在现实生活中实际使用这个模型时，它最终总是预测你走在理想轨迹上，并且永远不会从任何错误中恢复。这些模型只是学会了在测试中表现出色，但从未学会我们希望它们完成的真正任务，所以我们称它们为作弊者。

KL 解决方案

我们需要确保我们的模型能够学会在哪里驾驶，而不仅仅是如何利用模拟器的伪影。为了解决这个问题，我们可以尝试制作一个没有明显伪影的更好的模拟器，但这非常困难。相反，我们试图让模型对模拟器的伪影视而不见。

将羽衣甘蓝拉到模特的眼睛上

想象一下，一个特征向量包含与轨迹规划相关的特定道路场景的所有信息，但不包含任何其他信息，例如任何潜在的模拟器伪影。如果我们仅使用上述简单方法训练视觉模型并采用其输出的特征向量，则该向量应该包含与轨迹规划相关的大部分信息。这个特征向量对我们来说是无法解释的，但我们知道它一定包含这些信息，因为策略模型可以从中生成良好的轨迹。

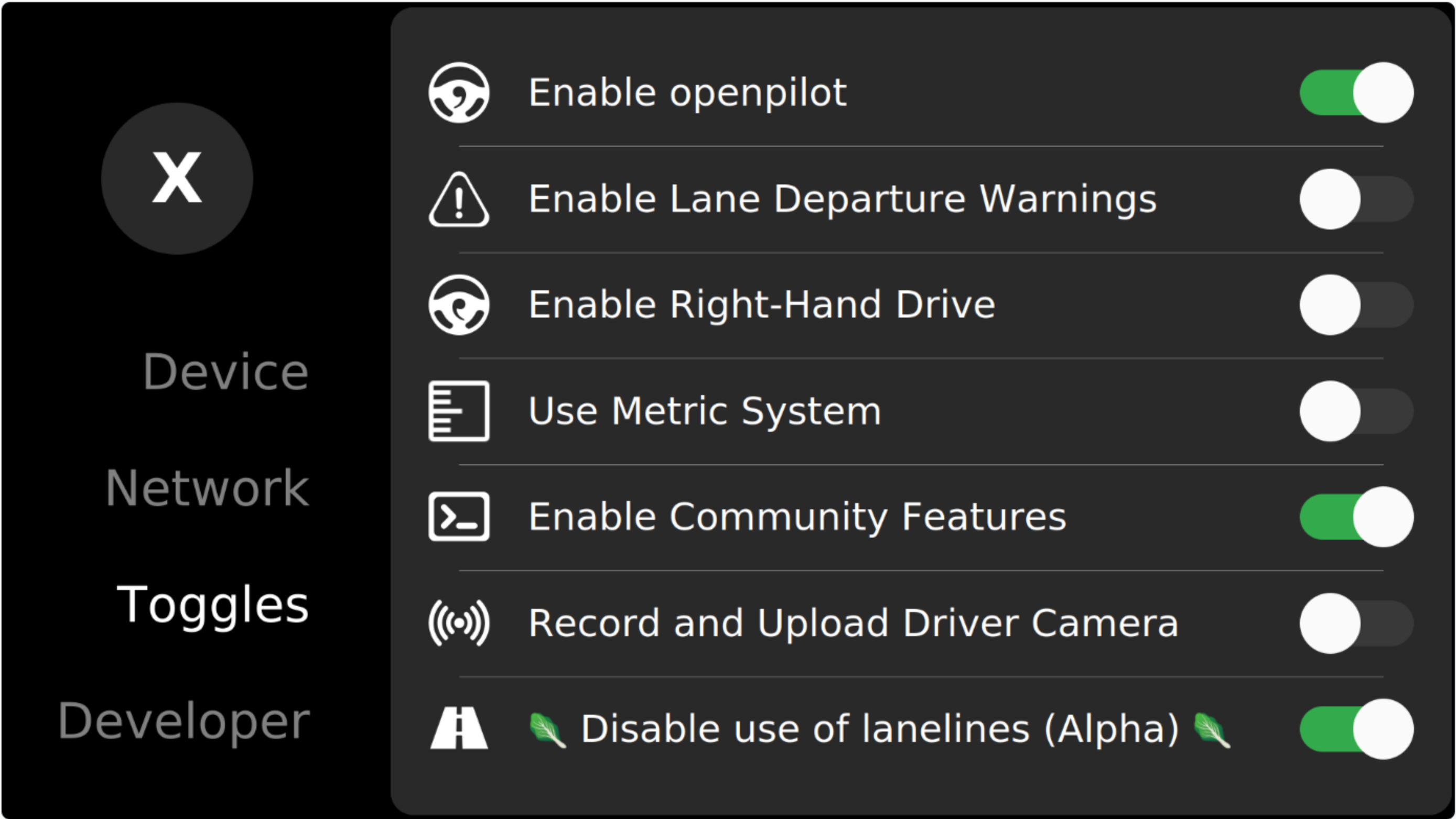


不幸的是，我们的测试表明，该向量仍然包含有关图像如何扭曲的信息。为了删除该信息，我们在训练时对该特征向量应用[KL 散度损失](#)。此损失使视觉模型最小化特征向量中的信息内容。当对此损失进行适当加权时，视觉模型将保留与轨迹规划相关的信息，因为这些信息是轨迹预测损失函数良好运行所必需的，并拒绝其他所有内容。生成的特征向量

仅包含与其训练问题相关的信息，即对未扭曲图像进行轨迹规划。

我们现在可以在模拟器中训练一个新模型，我们重用视觉模型中将其压缩为最小向量的部分并保持该部分冻结。我们只是训练一个新的策略模型。我们希望模型具有一些时间背景，因此我们还在策略模型中添加了一个 GRU 块。现在它将学习应该在哪里行驶而不利用扭曲。该模型现在在模拟和现实生活中都有效。

尝试一下！



openpilot 0.8.3 版本包含一个禁用车道线的开关。在此模式下，openpilot 将仅使用此 e2e 学习轨迹进行横向控制。亲自体验神奇之处！

加入我们

您是否有兴趣加入逗号研究团队？ 尝试[招聘挑战吧](#)！

Harald Schäfer, [comma.ai](#)
首席技术官