In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (IROS 2022), Kyoto, Japan, October 2022

VI-IKD: High-Speed Accurate Off-Road Navigation using Learned Visual-Inertial Inverse Kinodynamics

Haresh Karnan[1*], Kavan Singh Sikand[2*], Pranav Atreya[2], Sadegh Rabiee[2], Xuesu Xiao[2], Garrett Warnell[2,4], Peter Stone[2,3], and Joydeep Biswas[2]

***Abstract*— One of the key challenges in high-speed off-road navigation on ground vehicles is that the kinodynamics of the vehicle-terrain interaction can differ dramatically depending on the terrain. Previous approaches to addressing this challenge have considered learning an inverse kinodynamics (IKD) model, conditioned on inertial information of the vehicle to sense the kinodynamic interactions. In this paper, we hypothesize that to enable accurate high-speed off-road navigation using a learned IKD model, in addition to inertial information from the *past*, one must also anticipate the kinodynamic interactions of the vehicle with the terrain in the *future*. To this end, we introduce Visual-Inertial Inverse Kinodynamics (VI-IKD), a novel learning based IKD model that is conditioned on visual information from a terrain patch ahead of the robot in addition to past inertial information, enabling it to anticipate kinodynamic interactions in the future. We validate the effectiveness of VI-IKD in accurate high-speed off-road navigation experimentally on a scale 1/5 UT-AlphaTruck off-road autonomous vehicle in both indoor and outdoor environments and show that compared to other state-of-the-art approaches, VI-IKD enables more accurate and robust off-road navigation on a variety of different terrains at speeds of up to** $3.5m/s$**.**

I. INTRODUCTION

Constraining wheeled mobile robot navigation to structured environments and low speeds allows roboticists to use simplified assumptions about the robot's dynamics. Most state-of-the-art classical autonomous navigation systems [1], [2] incorporate motion planners that model a complex kinodynamic system such as a wheeled mobile robot using simplified kinematic models, often ignoring dynamic effects like slippage and wheel suspension. In addition to kinodynamic effects, delays caused by actuation latency inherent in the vehicle's hardware are often ignored. While ignoring such effects at low speeds may be acceptable, the combination of actuation latency coupled with kinodynamic responses due to vehicle-terrain interaction can have a magnified effect on the state of a vehicle when travelling at high speeds, and can be catastrophic (e.g., cause collisions) if not accounted for by the controller.

While accurate mathematical modelling of such effects is difficult [3]–[5], recent learning-based approaches to robot navigation have shown promising results in modelling the kinodynamic effects utilizing information from the Inertial Measurement Unit (IMU) to sense the vehicle-terrain interaction. Xiao et al. [6] introduce a learned inverse kinodynamics model (IKD) that enables a ground vehicle to sense the terrain and adaptively navigate at high speeds. This learned IKD model (henceforth called IMU-IKD) utilizes inertial sensors on a vehicle to sense the vehicle-terrain interactions and takes a data-driven approach to model the kinodynamic effects experienced by the vehicle on different terrains. However, an inertial sensor is limited in its capability: it can only sense interactions with terrain *after* the vehicle has driven over it. During high speed navigation, latency inherent in the hardware of a vehicle causes actuation commands to be executed at a future world position. Thus, when traversing between terrain types, it is important for the vehicle to *proactively* adjust its controls based on the terrain it is about to encounter in the future, not just the terrain it is currently driving over. A model relying on inertial information alone cannot foresee the kinodynamic response at this future position. Unlike an inertial sensor, a visual sensor from an egocentric viewpoint enables perception of the world ahead, providing information about the terrain the vehicle will interact with in the future. We therefore hypothesize that in addition to inertial information from the past, conditioning a learned IKD model on the visual information of the terrain ahead will improve the vehicle's capability to accurately navigate at high speeds.

Towards this end, in this paper, we present Visual-Inertial Inverse Kinodynamics (VI-IKD), a novel, computationally tractable learning-based approach for incorporating visual information into an inverse kinodynamic model. VI-IKD conditions the IKD model on—in addition to inertial information—a visual patch of terrain in the future, by sub-sampling an image captured from a forward-facing camera and extracting only the region where the next actuation command will be executed, considering actuation delays. Specifically, VI-IKD learns a viewpoint-invariant representation of visual terrain patches combined with inertial information captured by an on-board IMU to learn a terrain cognizant IKD model. The resultant IKD model is capable of anticipating the effect of terrain on the robot's dynamics and proactively adapts controls to accurately track planned trajectories on varying types of terrain.

We evaluate the performance of VI-IKD on a scale 1/5 Ackermann-drive vehicle in challenging indoor and outdoor real-world environments with varying types of terrain and

[1] The University of Texas at Austin, Department of Mechanical Engineering haresh.miriyala@utexas.edu

[2] The University of Texas at Austin, Department of Computer Science, {kvsikand, pranavatreya}@utexas.edu, {srabiee, xiao, joydeepb, pstone}@cs.utexas.edu

[3] Sony, AI

[4] Computational and Information Sciences Directorate, Army Research Laboratory garrett.a.warnell.civ@army.mil

*Equal Contribution

demonstrate that it can accurately navigate the robot at high speeds of up to $3.5m/s$, resulting in improved success rates on the task of reference trajectory following, compared to state-of-the-art approaches.

II. RELATED WORK

In this section, we first review related literature on classical methods for wheeled robot navigation in the presence of wheel slippage. We then survey related learning-based approaches for off-road robot navigation.

A. Physics-Based Kinodynamic Models

There exists a plethora of research on empirically derived physics-based dynamic and kinodynamic models for wheeled mobile robots that predict the effects of wheel slippage [7]–[9]. Seegmiller et al. [7] propose a parametric kinodynamic model to predict the residual velocity of the robot with respect to the output of a pure kinematic model, given the velocity of the robot and the estimated centrifugal forces. Rabiee et al. [8] incorporate an empirical wheel-terrain interaction model into the forward kinematic model of skid-steer robots. All of these approaches include a calibration phase that is performed separately for each discrete type of terrain. During inference, these methods rely on perception modules to classify the terrain into pre-specified classes using IMU and camera data [10], [11] in order to switch between different terrain-dependent parameter sets.

B. Error Modelling and Reactive Control

In off-road unstructured environments, the terrain traversed by the robot cannot be easily delineated into large uniform regions. Instead, there exist frequent transitions between terrain types, e.g. small patches of grass or loose leaves on dirt, such that different robot wheels can be in contact with patches of terrain with significantly different characteristics. Xiao et al. [6] treat terrain characteristics in a continuous manner and learn an inverse-kinodynamic model that uses a history of IMU data along with the robot's current and desired state to issue control commands. They demonstrate that this approach enables the robot to accurately navigate at high speeds on unstructured terrain without an explicit enumeration of terrain types. Another line of work that does not require enumeration of terrain types is closed-loop motion control for trajectory following in the presence of slip [12]–[14]. Koppel et al. [15] learn a statistical model for terrain disturbance using control and visual information. Ostafew et al. [14] learn a non-parametric disturbance model online to compensate for slippage that is estimated using visual odometry. These methods are inherently reactive to the sensed changes in terrain characteristics, and therefore only target low-speed navigation applications such as planetary exploration rovers. In high-speed navigation, however, the effect of motion control loop delay on trajectory tracking accuracy is significant, as the robot displacement during the period of a control loop is considerable. Sensory information from cameras and LiDAR reveals a great deal about the characteristics of terrain, and can be leveraged to anticipate its effects on the robot's dynamics. While researchers have recently started to incorporate visual information into gait planners for legged-robots [16], wheeled mobile robot motion planners that use visual information have been mostly limited to end-to-end learning solutions.

C. Learning for Off-Road Navigation

With the initial success of applying machine learning techniques to mobile robot navigation instead of explicitly modeling the environment and designing complex navigation systems [17]–[29], roboticists have also applied learning for off-road navigation. Pan et al. [30] propose an end-to-end learning solution that uses camera and odometry data to navigate a high-speed robot on a race track. While such learning-based solutions are appealing for their ability address perception, planning, and control together in a single model, they require large amounts of training data and struggle to generalize to new environments. Siva et al. [31] enhance ground maneuverability consistency on complex off-road terrain by learning offset behaviors in a self-supervised fashion to compensate for the inconsistency between the actual and expected behaviors without requiring the explicit modeling of various confounding factors. Other prior works in the literature have taken a hybrid approach, e.g., learning from visual information for slip-aware robot navigation to estimate the traversal cost of different regions of terrain [29], [32], [33]. Angelova et al. [33] propose a non-parametric method for learning to predict slip on patches of terrain given the appearance and geometric properties perceived by stereo-vision. The resultant information is used to inform the robot to avoid challenging terrain types. Our work, however, seeks to learn to navigate the robot on such challenging terrain as it is unavoidable in unstructured off-road environments.

Our approach is similar to the approach by Xiao et al. [6] in that we learn an inverse kinodynamic model for motion planning without enumerating discrete types of terrain, but we incorporate visual information as well as IMU data in a computationally tractable manner to anticipate the effects of future terrain on the robot's dynamics, making our approach significantly more responsive to variations in terrain characteristics and robust to the effects of actuation latency during high-speed maneuvers.

III. METHOD

In this section we discuss the formulation of the navigation problem and our novel Visual-Inertial Inverse Kinodynamic (VI-IKD) approach.

A. Problem Formulation

The goal of a navigation planner is to incorporate both global and local information to identify a sequence of actions to take a robot from its current state x_0 to a target state x_n which it attempts to reach as efficiently and safely as possible. For simplicity of notation, we will treat the robot's traversal through the environment as a sequence of timesteps, which can be arbitrarily small. The planned sequence of states $\{x_0, x_1, ..., x_n\}$ is referred to as the navigation plan.

At a given timestep $t \in [0, n)$, the navigation planner is responsible for producing navigation command u_t with the goal of taking the robot from state x_t to x_{t+1}.

Given a vehicle state x_t, a control input u_t, and a world state w, the robot's true response upon executing u_t is given by its forward kinodynamic function f

$$x_{t+1} = f(x_t, u_t, w). \tag{1}$$

The navigation planner is therefore attempting to find u_t such that:

$$u_t = f^{-1}(x_t, x_{t+1}, w) \tag{2}$$

In practice, existing navigation planners struggle to accurately model f^{-1}, and therefore after executing u_t, the resultant robot state $\hat{x}_{t+1}$ does not match the navigation planner's intended subsequent robot state x_{t+1}. There are two primary reasons for this inconsistency: the navigation planner uses a simplified model of the robot's motion response (often considering only the kinematic response), and the world state w is not directly observable, and therefore the planner does not have sufficient information to correctly estimate the effects of f.

Recent work has made great strides towards enabling a motion planner to encode complex system dynamics by leveraging deep neural networks, adding a learned inverse kinodynamic module which indirectly captures world state w [6]. For example, Xiao et al. [6] introduce the IMU-IKD algorithm in which a recent history of the robot's inertial state $S_t^h = \{s_{t-k}, ...s_{t-1}\}$, where k is the length of the history, is used to estimate the world state w for timestep t. Specifically, the IMU-IKD algorithm [6] estimates f^{-1} by learning a function f_θ^{IMU} such that:

$$f^{-1}(x_t, x_{t+1}, w) \approx f_\theta^{\text{IMU}}(x_t, x_{t+1}, S_t^h) \tag{3}$$

Using a history of recent sensor observations relies on the assumption that the current world state w can be predicted from a recent history of inertial observations. However, for an inertial sensor, this may not always be true. For example, a robot driving on bumpy terrain may subsequently encounter smooth terrain, where the inertial response is much different; Even though the smoothness of the terrain ahead where the next actuation command will be executed is a part of the world state that significantly affects the state of the vehicle, an inertial sensor cannot detect this change unless the vehicle drives over the smooth terrain. To address this limitation, in this work, we propose using extereoceptive sensors, specifically RGB images, to help inform the motion planner of the world state w before the vehicle physically interacts with the terrain ahead. A front-facing camera can see parts of the terrain that the robot has not yet encountered, which enables the use of image observations from previous timesteps to help estimate the current world state. We define λ_t as the visual terrain information obtained for timestep t. Our visual-inertial inverse kinodynamic module therefore attempts to find a function f_θ^{VI} which estimates f^{-1} such that:

$$f^{-1}(x_t, x_{t+1}, w) \approx f_\theta^{\text{VI}}(x_t, x_{t+1}, S_t^h, \lambda_t) \tag{4}$$

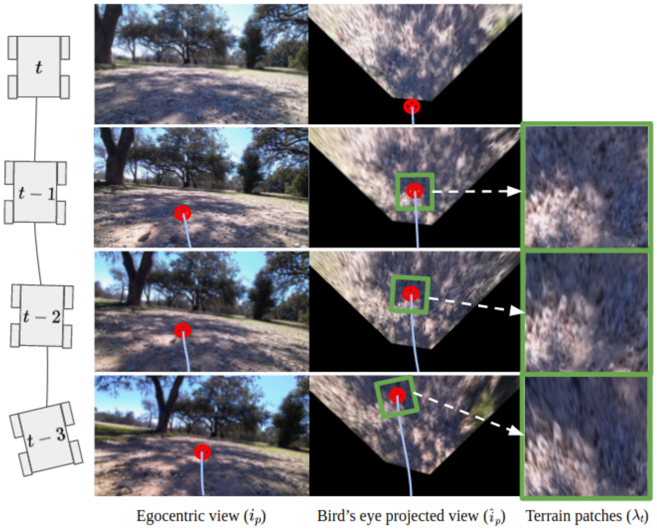
t
t − 1
t − 2
t − 3
Egocentric view (i_p)
Bird's eye projected view ($\hat{i}_p$)
Terrain patches (λ_t)

Fig. 1: Overview of the Visual Patch Extraction process at time t. The robot's next location $\tilde{x}_t$ is estimated (red circle) based on current velocity, and a visible image patch of terrain at the same consistent location $\tilde{x}_t$ is extracted from bird's eye view images from previous timesteps of different viewpoints.

The process for obtaining λ_t is given in Sec. III-B and shown in Fig. 1. The process for training f_θ^{VI} is given in Sec. III-C. The resulting navigation system is summarized in Fig. 2.

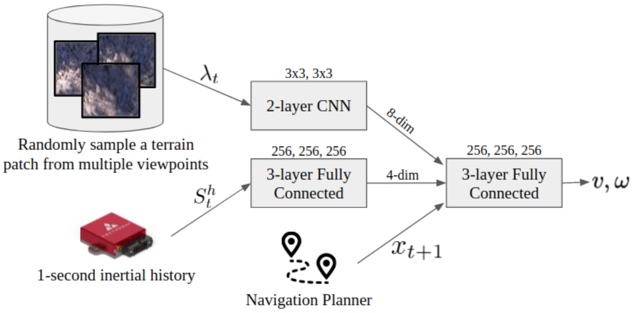

Fig. 2: Training setup for the Visual-Inertial Inverse Kinodynamic model. VI-IKD samples one image at random from a set of terrain patches of the same location in ground as viewed from different viewpoints, in every epoch of training. This viewpoint-invariant visual representation combined with inertial information and desired next state is used as input to the IKD model that produces the action commands.

B. Visual Patch Extraction

When incorporating visual information into the inverse kinodynamic model, it is important to ensure that the visual information used is relevant to the prediction task at hand. Specifically, this should be visual information corresponding to the terrain under the robot at the time a given command is executed, as this is the part of the world state relevant to the kinodynamic response of the robot. As the robot moves through the environment, the front-facing camera captures an egocentric view of the terrain the robot is approaching. The patch of terrain under the robot at any point in time can be extracted from previous camera images with knowledge of the pose information of the robot between frames.

For a particular timestep t, a set of captured camera images I, a set of IMU measurements S, and recent odometry measurements O, we seek to find λ_t, the visual information relevant to the robot's current navigation command. To this end, we define a patch extraction operator $P : \{I, S, O\} \rightarrow \Lambda$ which extracts patches of visual information $\lambda_t \in \Lambda$ of a terrain ahead, from recorded history of observations where the next actuation command will be executed. This operator takes as input a camera image $i_p \in I$ from some timestep $p < t$. This camera image is projected to a birds-eye view (BEV) $\hat{i}_p$ using a homography transform H derived from the static extrinsic camera calibration and s_p, the inertial data of the robot at time p. We compute the homography transform H in real-time considering the inertial data from the robot due to significant roll-pitch motion experienced during high-speed maneuvers. After this transformation, a fixed distance in BEV projected pixel-space corresponds to a fixed distance in the real world along the ground plane. Once this is done, the robot's real-time recent history of odometry estimates O is used to determine the robot's location relative to the location from which the image was captured. Finally, the robot's current odometry information $o_t \in O$ is used to predict the future location, $\tilde{x}_t$, of the robot in the bird's eye view image plane, where the robot will be at the time when its next issued command will be executed. Note that for a command issued at time t and robot state x_t, the command will be executed on the robotic platform at a slightly later state $\tilde{x}_t$ due to actuation latency on a real robot platform. The patch λ is then defined as the region of the image around location $\tilde{x}_t$, and is extracted from $\hat{i}_p$. This patch extraction process is shown in Fig. 1. The patches extracted from this process are significantly smaller than a full camera image, enabling VI-IKD to run in real-time.

During the training step, VI-IKD uses all observations from different viewpoints of the same consistent location to learn a viewpoint invariant visual representation of that location. By repeating this procedure for different locations in the world, we ensure that VI-IKD is *viewpoint-invariant* – that is, it is invariant to observations of the same location irrespective of image variations due to differing observing poses. This procedure also provides robustness to image aberrations and distortion due to artifacts such as motion blur.

C. Learning Visual-Inertial Inverse Kinodynamics

To train the VI-IKD module, we collect a set of human demonstrations D in an open environment by teleoperating the vehicle with a joystick. For each demonstration $d \in D$, we track joystick commands U, inertial data S, odometry data O, and image data I, and we record the observed sequence of robot states X_{obs}. We then generate training samples of the form $\langle x_{t+1}, x_t, O_t^h, S_t^h, i_t, u_t \rangle$, where $x_{t+1} \in X_{obs}$ is the desired robot state, $x_t \in X_{obs}$ is the preceding state, $S_t^h \subset S$ is the recent inertial history of the robot, $O_t^h \subset O$ is the recent history of odometry measurements, $i_p \in I : p < t$ is a recent camera image, and $u_t \in U$ is the command which transitions the robot from state x_t to x_{t+1}. Because we are recording actual observations, these samples encode the true kinodynamic response of the robot, and we know $f^{-1}(x_t, x_{t+1}, w) = u_t$. Given our patch extraction operator P, our training loss then seeks to find parameters θ which minimize

$$\arg\min_{\theta} \sum ||u_t - f_\theta^{\text{VI}}(x_t, x_{t+1}, S_t^h, P(i_p, S_t^h, O_t^h))||. \quad (5)$$

This learning objective enforces that the VI-IKD-generated control for reaching state x_{t+1} from x_t matches the controls that were actually executed to effect that change. Note that in this formulation, for each x_t, we frequently have multiple different preceding states from which visual information i_p can be extracted, as each traversed patch of terrain may appear in multiple preceding image frames. In these situations, we replicate this for each available choice of $i_p \in I$ from which a patch can be extracted, which helps ensure that regardless of the viewpoint, we learn the same mapping of visual information to predicted command. Regularizing the training process with terrain patches from a consistent location on the ground, but as seen from different viewpoints at different times provides viewpoint invariance in the learned visual representations. In the event where there is no patch information available for a sample, we provide a vector of zeros as the visual representation to the IKD model.

D. Implementation Details

The Visual Inverse Kinodynamic Module f_θ^{VI} consists of a visual encoder (2-layer convolutional neural network with a kernel size of 3 and stride of 2), an IMU encoder (3-layer Multi-Layer Perceptron (MLP) with skip connections and hidden layers of size 256), and a final shared 3-layer MLP with skip connections and hidden layers of size 256. To ensure fair comparison, the baseline IMU-IKD algorithm uses the same network architecture for the IMU encoder and the IKD network. The network architecture along with the inputs during training time are shown in Fig. 2. The visual encoder was run off-board at inference time using a GPU-enabled laptop (Nvidia RTX 2060). We regularize the training by randomly sampling a visual terrain patch for a data sample $\langle x_{t+1}, x_t, O_t^h, S_t^h, i_t, u_t \rangle$ from a set of visual terrain patches of the same unique location sub-sampled from observations recorded at previous timesteps. We maintain a buffer of 30 past images to perform patch extraction. The terrain patches are RGB images of fixed size 64-by-64. This patch size was chosen to maximize visual information while ensuring the VI-IKD model can run at 40hz on the GPU with PyTorch and CUDA acceleration.

IV. Experimental Results

To evaluate the effectiveness of the Visual-Inertial Inverse Kinodynamic (VI-IKD) model in accurately tracking a trajectory at high speeds, we performed a series of experiments in a controlled indoor environment and an unstructured outdoor environment with different terrains. In this section,

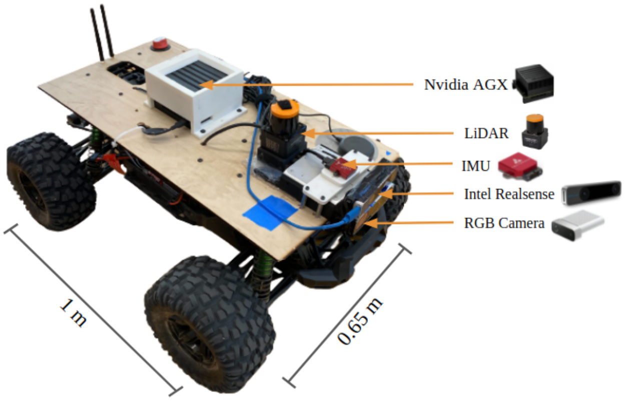

Fig. 3: The UT-AlphaTruck scale 1/5th autonomous vehicle and various attached sensors utilized in this work.

we describe the experimental setup followed by the indoor and outdoor experiments.

A. Experimental Setup

We used the same robotic platform for all experiments, pictured in Fig. 3: the UT-AlphaTruck, a 1/5th scale Ackermann-steer vehicle. The robotic platform is equipped with a Hokuyo planar LiDAR (for obstacle detection and localization), an Intel RealSense T265 tracking camera (for obtaining odometry at 200Hz), a VectorNav VN200 Inertial Measurement unit (for obtaining 6-axis accelerometer and gyroscope measurements at 200Hz), an Azure Kinect camera (for obtaining RGB images at 30Hz), and a Nvidia Xavier AGX (for on-board compute). For the patch extraction procedure, we compute the actuation latency of this hardware to be approximately 0.25 seconds. We do so by subtracting the sensing latency of the RealSense from the sense-act latency (between an issued joystick command and its result as measured by the Intel RealSense). In our experiments, all methods use a graph-based global planner [1] which provides the desired next state x_{t+1} towards a navigation goal. For the indoor experiments, we use Episodic non-Markov Localization (EnML) [34] to track the vehicle's state by fusing LiDAR observations and Intel RealSense's visual-odometry estimates. To collect demonstration data for the training the IKD models, we teleoperate the vehicle using a joystick with $v \in [0, 4]m/s$ and $\omega \in [-1.8, 1.8]rad/s$ for 60 minutes, randomly varying the linear and angular velocities every trajectory. In total, we collect about 32 trajectories containing 73,238 data samples and split them equally into train and test sets. Training the VI-IKD model takes less than 10 minutes on a Nvidia RTX 2060 laptop GPU. We use the same data to train both the IMU-IKD and the VI-IKD models.

We compare our method to two alternate approaches:

- **Baseline**: The base navigation stack of the Autonomous Mobile Robotics laboratory, which includes a trajectory-rollout based receding horizon local planner that uses a basic kinematic motion model for an Ackermann-drive vehicle [1], [8].

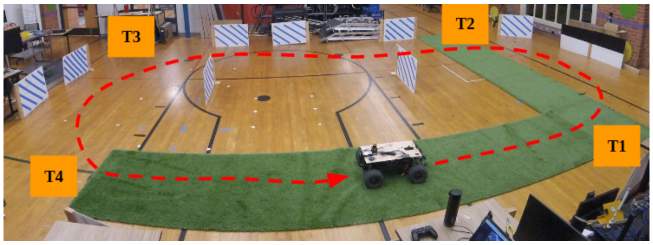

Fig. 4: Indoor evaluation environment. Evaluation trajectory is illustrated in red. T1, T2, T3, T4 indicate the four distinct turns in the trajectory. Striped blue rectangular posts are the virtual fixtures used to aid indoor localization using EnML [34].

TABLE I: Navigation Success Rates in Indoor Environment at $3.2m/s$

	Turn 1	**Turn 2**	**Turn 3**	**Turn 4**
Navigation Controller	Success Count	Success Count	Success Count	Success Count
Baseline	0	7	8	5
IMU-IKD	9	**10**	**10**	**10**
VI-IKD (Ours)	**10**	**10**	**10**	**10**

- **IMU-IKD**: The IMU based IKD model (IMU-IKD), introduced by Xiao et al. [6]. The IMU-IKD model takes as its inputs inertial history S_t^h of the vehicle and a desired next state x_{t+1} to predict a low-level actuation command (forward velocity v and angular velocity ω).

The Visual-Inertial Inverse Kinodynamic (**VI-IKD**) model utilizes both inertial and visual information from on-board sensors and produces low-level actuation commands based on the desired next state x_{t+1} provided by the global planner.

B. Indoor Experiments

To evaluate the effectiveness of VI-IKD in accurately and successfully tracking a desired trajectory at high speeds, we set up an indoor course (30 meters long, 15 meters wide) containing two distinct terrain types with different kinodynamic responses at high speeds—wooden floor and green turf—shown in Fig. 4. The scale 1/5 UT-AlphaTruck vehicle used in these experiments experiences significantly more slip on the wooden floor than on the green turf at high speeds. To aid localization in providing accurate state estimates, we set up virtual fixtures (shown as striped blue posts in Fig. 4). To obtain a reference trajectory for navigating this environment, we allowed the robot to autonomously navigate (counter-clockwise) between manually-defined waypoints using the baseline navigation implementation at a slow speed ($0.5m/s$). At this speed, the impact of dynamics is minimal, and the baseline kinematic motion planner is sufficient for accurate trajectory following. We performed 10 trials for all three navigation systems (baseline, IMU-IKD and VI-IKD) at a nominal speed of $2.0m/s$ and at high speeds ranging from $2.5m/s$ - $3.2m/s$ in increments of $0.1m/s$. In total, we perform 270 laps across this loop to evaluate the three approaches. Due to the limited size of the indoor track, the

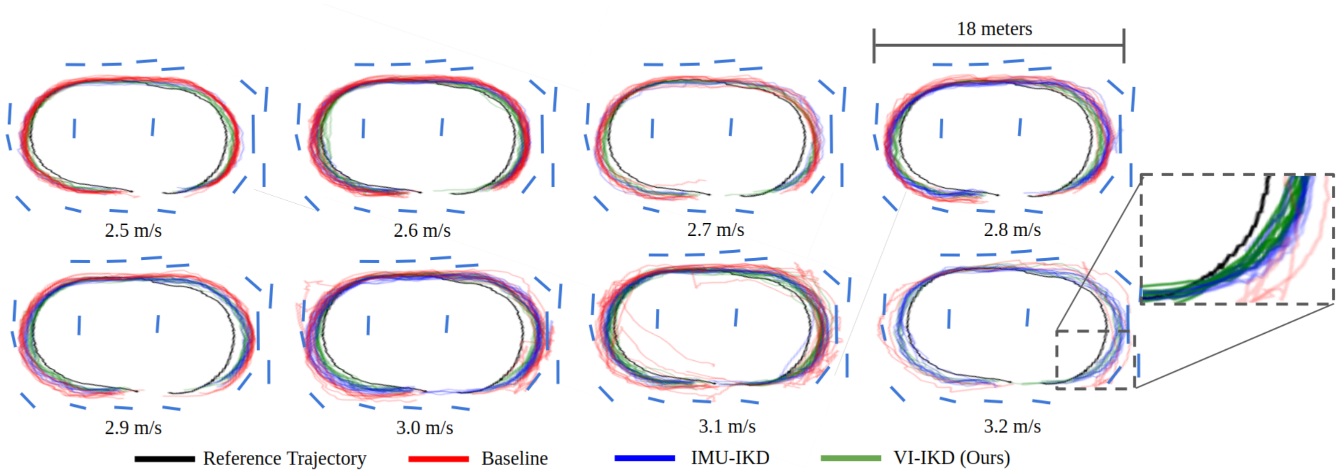

Fig. 5: Trajectory traces for the indoor experiments where the vehicle tracks a reference trajectory counter-clockwise at different speeds. The inset shows Turn 1 as executed by the vehicle at $3.2m/s$ using the three approaches. We see that VI-IKD is able to track the reference trajectory more accurately than IMU-IKD [6], confirming our hypothesis. Blue lines along the track show the virtual fixtures used as a map for vehicle state-estimation using EnML [34].

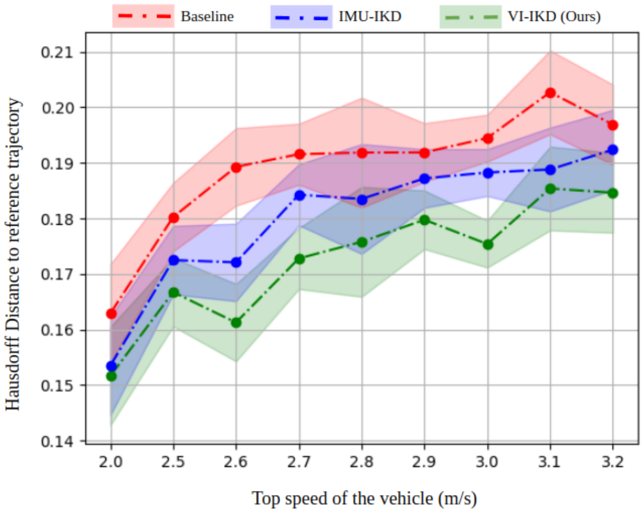

Fig. 6: Hausdorff distance (lower is better) between the reference trajectory and trajectories traced by different algorithms at different top speeds of the vehicle in the indoor evaluation environment. We see that VI-IKD is the most accurate compared to the receding horizon controller with no IKD model (baseline) and IMU-IKD [6].

baseline navigation model caused frequent unsafe collisions, preventing us from running experiments at speeds greater than $3.2m/s$. However, the outdoor experiments presented in Section IV-C show the potential of VI-IKD to successfully navigate at high speeds of up to $3.5m/s$.

The resultant trajectory traces for each of the navigation systems at various speeds, as well as the reference trajectory, are presented in Fig. 5. We see that the VI-IKD model introduced in this work is more accurate than the baseline sampling-based local planner and the state-of-the-art IMU-IKD model [6]. Additionally, we tracked each system's success rate at navigating turns in the environment, where success is any turn that did not result in a collision. We present these success rates when travelling at a maximum speed of $3.2m/s$ in Table I. To obtain a quantitative measurement of the accuracy of each navigation system, we use an undirected Hausdorff distance, which measures the distance from each point in the trajectory Γ to the closest point in the reference trajectory:

$$H(\Gamma_a, \Gamma_b) = \max(d(\Gamma_a, \Gamma_b), d(\Gamma_b, \Gamma_a)), \qquad (6)$$
$$d(\Gamma_a, \Gamma_b) = \max_{a \in \Gamma_a} \min_{b \in \Gamma_b} ||a - b||.$$

The results of this numerical evaluation for each navigation system at different navigation speeds is presented in Fig. 6. We see that VI-IKD is the most accurate compared to the receding horizon controller with no IKD model (baseline) and IMU-IKD [6].

C. Outdoor Experiments

In addition to the controlled indoor environment, we evaluate VI-IKD in a heterogeneous outdoor environment. We run each navigation system through a fixed set of target waypoints in the environment pictured in Fig. 7 at a speed of $3.5m/s$. We provide the reference trajectory to track by manually teleoperating the vehicle around the off-road track. For this trajectory following task outdoors, all algorithms in this experiment use the Intel RealSense's visual-odometry estimates for localization because unlike the controlled indoor experiments, the outdoor track is in off-road, open-ground conditions, unsuitable for accurate LiDAR based localization [34]. The outdoor track (50 meters long, 30 meters wide) contains three major turns during which the robot had the potential to slip and deviate from the desired trajectory at high speeds of $3.5m/s$. Specifically, in Turn 1, the robot makes a u-turn while transitioning from slippery fine sand into grass with increased friction. In Turn 2, the

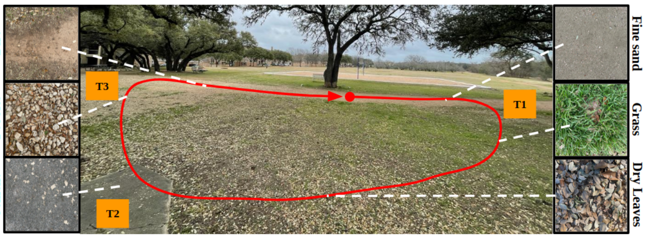

Fig. 7: Outdoor Evaluation Environment. Various traversed terrain types are highlighted, and the evaluation trajectory is illustrated in red. T1, T2, and T3 indicate the distinct turns in the trajectory.

TABLE II: Navigation Results in Outdoor Environment at $3.5m/s$.

	Turn 1	Turn 2	Turn 3
Navigation Controller	Success Count	Success Count	Success Count
Baseline	6	**10**	3
IMU-IKD	8	7	8
VI-IKD (Ours)	**10**	**10**	**10**

robot transitions between grass, dry leaves, cement and onto pebbles, each producing different kinodynamic responses at high speeds. Finally in Turn 3, the vehicle makes a nearly 180 degree turn on pebbles and enters into a dirt track, which can cause significant slippage at high speeds. Refer to the supplementary video for visual comparisons of the laps performed by the vehicle in this off-road track. The three turns contain significant variance in terrain, requiring an IKD model to anticipate the kinodynamic responses to navigate successfully at high speeds.

In our evaluation, each model performs ten laps across this outdoor loop. We mark a turn as unsuccessful if the robot deviates from the desired trajectory beyond the point at which the navigation stack is able to get the robot back on track. At such a failure, we resume trajectory tracking after re-initializing the vehicle in the track at a position after the unsuccessful turn. In this experiment, we measured the rate at which each navigation system was able to successfully navigate each turn, and present the results of 10 repetitions of the course in Table II. We see that unlike the baseline and IMU-IKD model, VI-IKD is able to successfully complete all turns at a high speed of $3.5m/s$. Although IMU-IKD performs better than baseline, the different terrain types present in these turns make it challenging for IMU-IKD model to track the reference trajectory without anticipating kinodynamic interactions with the terrain ahead. By anticipating the kinodynamic effects, the VI-IKD model is able to proactively control the vehicle and complete the loops successfully in all 10 trials without any failures.

V. Conclusion

In this work, we introduce Visual-Inertial Inverse Kinodynamics (VI-IKD), a novel approach for leveraging visual terrain information ahead in addition to inertial information of the past to enhance accuracy in high-speed navigation using a learned IKD model. We hypothesized that utilizing visual information of the terrain helps an IKD model to anticipate kinodynamic effects of the vehicle-terrain interaction and proactively control the vehicle to navigate accurately at high speeds while accounting for actuation delays. Towards this end, the proposed VI-IKD model leverages visual information by learning a viewpoint-invariant representation of the terrain patch ahead, which is used to anticipate kinodynamic responses for the next actuation command executed in the terrain ahead. We validate our hypothesis by comparing VI-IKD to state-of-the-art approaches on the task of trajectory following in both indoor and outdoor real-world environments on a scale 1/5 Ackermann-drive vehicle and observe that VI-IKD is able to navigate successfully around turns at high speeds of up to $3.5m/s$ outdoors, and that VI-IKD is able to accurately track a reference trajectory at speeds of up to $3.2m/s$ indoors.

VI. Future Work

There are a few avenues one could pursue to further improve the performance of VI-IKD in future work. First, one could consider a longer control horizon [35], rather than the one-step horizon of control we currently use. This would allow the robot to pursue short-term sub-optimal actions to improve long-term utility. Additionally, one could investigate and improve the performance of VI-IKD in unseen terrains, which is essential in off-road conditions where a high-speed vehicle may encounter novel terrains. Finally, one could incorporate additional sensors such as microphones and ground-facing range sensors to further improve the learned terrain representations.

Acknowledgement

This work has taken place in the Learning Agents Research Group (LARG) and Autonomous Mobile Robotics Laboratory (AMRL) at UT Austin. LARG research is supported in part by NSF (CPS-1739964, IIS-1724157, NRI-1925082), ONR (N00014-18-2243), FLI (RFP2-000), ARO (W911NF19-2-0333), DARPA, Lockheed Martin, GM, and Bosch. AMRL research is supported in part by NSF (CAREER2046955, IIS-1954778, SHF-2006404), ARO (W911NF-19-2- 0333,W911NF-21-20217), DARPA (HR001120C0031), Amazon, JP Morgan, and Northrop Grumman Mission Systems. Peter Stone serves as the Executive Director of Sony AI America and receives financial compensation for this work. The terms of this arrangement have been reviewed and approved by the University of Texas at Austin in accordance with its policy on objectivity in research. We would like to thank Corrie Van Sice and Keith Fritz for their assistance with the physical hardware experiments.

References

[1] J. Biswas and M. M. Veloso, "Localization and navigation of the cobots over long-term deployments," *The International Journal of Robotics Research*, vol. 32, no. 14, pp. 1679–1694, 2013. [Online]. Available: https://doi.org/10.1177/0278364913503892

[2] "ROS movebase navigation stack," http://wiki.ros.org/move_base, accessed: 2021-09-9.

[3] J. P.Hanna, S. Desai, H. Karnan, G. Warnell, and P. Stone, "Grounded action transformation for sim-to-real reinforcement learning," *Special Issue on Reinforcement Learning for Real Life, Machine Learning, 2021*, May 2021.

[4] S. Desai, I. Durugkar, H. Karnan, G. Warnell, J. Hanna, and P. Stone, “An imitation from observation approach to transfer learning with dynamics mismatch,” 2020. [Online]. Available: https://arxiv.org/abs/2008.01594

[5] S. Desai, H. Karnan, J. P. Hanna, G. Warnell, and P. Stone, “Stochastic grounded action transformation for robot learning in simulation,” 2020. [Online]. Available: https://arxiv.org/abs/2008.01281

[6] X. Xiao, J. Biswas, and P. Stone, “Learning inverse kinodynamics for accurate high-speed off-road navigation on unstructured terrain,” *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 6054–6060, 2021.

[7] N. Seegmiller, F. Rogers-Marcovitz, G. Miller, and A. Kelly, “Vehicle model identification by integrated prediction error minimization,” *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 912–931, 2013.

[8] S. Rabiee and J. Biswas, “A friction-based kinematic model for skid-steer wheeled mobile robots,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8563–8569.

[9] M. Tarokh and G. J. McDermott, “Kinematics modeling and analyses of articulated rovers,” *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 539–553, 2005.

[10] E. G. Collins and E. J. Coyle, “Vibration-based terrain classification using surface profile input frequency responses,” in *2008 IEEE International Conference on Robotics and Automation*. IEEE, 2008, pp. 3276–3283.

[11] M. Wigness, S. Eum, J. G. Rogers, D. Han, and H. Kwon, “A rugd dataset for autonomous navigation and visual perception in unstructured outdoor environments,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 5000–5007.

[12] R. Gonzalez, F. Rodriguez, J. L. Guzman, C. Pradalier, and R. Siegwart, “Control of off-road mobile robots using visual odometry and slip compensation,” *Advanced Robotics*, vol. 27, no. 11, pp. 893–906, 2013.

[13] D. M. Helmick, A. Angelova, M. Livianu, and L. H. Matthies, “Terrain adaptive navigation for mars rovers,” in *2007 IEEE Aerospace Conference*. IEEE, 2007, pp. 1–11.

[14] C. J. Ostafew, A. P. Schoellig, T. D. Barfoot, and J. Collier, “Learning-based nonlinear model predictive control to improve vision-based mobile robot path tracking,” *Journal of Field Robotics*, vol. 33, no. 1, pp. 133–152, 2016.

[15] A. Koppel, J. Fink, G. Warnell, E. Stump, and A. Ribeiro, “Online learning for characterizing unknown environments in ground robotic vehicle models,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 626–633.

[16] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, “Learning robust perceptive locomotion for quadrupedal robots in the wild,” *Science Robotics*, vol. 7, no. 62, p. eabk2822, 2022.

[17] X. Xiao, B. Liu, G. Warnell, and P. Stone, “Motion control for mobile robot navigation using machine learning: a survey,” *Autonomous Robots*, 2022.

[18] M. Bojarski, D. D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba, “End to end learning for self-driving cars,” 2016.

[19] L. Tai, S. Li, and M. Liu, “A deep-network solution towards modelless obstacle avoidance,” in *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2016, pp. 2759–2764.

[23] Z. Wang, X. Xiao, A. J. Nettekoven, K. Umasankar, A. Singh, S. Bommakanti, U. Topcu, and P. Stone, “From agile ground to aerial navigation: Learning from learned hallucination,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 148–153.

[20] M. Pfeiffer, M. Schaeuble, J. Nieto, R. Siegwart, and C. Cadena, “From perception to decision: A data-driven approach to end-to-end motion planning for autonomous ground robots,” in *2017 ieee international conference on robotics and automation (icra)*. IEEE, 2017, pp. 1527–1533.

[21] X. Xiao, B. Liu, G. Warnell, and P. Stone, “Toward agile maneuvers in highly constrained spaces: Learning from hallucination,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1503–1510, 2021.

[22] X. Xiao, B. Liu, and P. Stone, “Agile robot navigation through hallucinated learning and sober deployment,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 7316–7322.

[24] L. Tai, G. Paolo, and M. Liu, “Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 31–36.

[25] H.-T. L. Chiang, A. Faust, M. Fiser, and A. Francis, “Learning navigation behaviors end-to-end with autorl,” *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 2007–2014, 2019.

[26] Y. F. Chen, M. Liu, M. Everett, and J. P. How, “Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning,” in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 285–292.

[27] H. Karnan, G. Warnell, X. Xiao, and P. Stone, “Voila: Visual-observation-only imitation learning for autonomous navigation,” in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022.

[28] H. Karnan, A. Nair, X. Xiao, G. Warnell, S. Pirk, A. Toshev, J. Hart, J. Biswas, and P. Stone, “Socially compliant navigation dataset (scand): A large-scale dataset of demonstrations for social navigation,” 2022. [Online]. Available: https://arxiv.org/abs/2203.15041

[29] T. Manderson, S. Wapnick, D. Meger, and G. Dudek, “Learning to drive off road on smooth terrain in unstructured environments using an on-board camera and sparse aerial images,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1263–1269.

[30] Y. Pan, C.-A. Cheng, K. Saigol, K. Lee, X. Yan, E. A. Theodorou, and B. Boots, “Imitation learning for agile autonomous driving,” *The International Journal of Robotics Research*, vol. 39, no. 2-3, pp. 286–302, 2020.

[31] S. Siva, M. Wigness, J. Rogers, and H. Zhang, “Enhancing consistent ground maneuverability by robot adaptation to complex off-road terrains,” in *5th Annual Conference on Robot Learning*, 2021.

[32] P. Drews, G. Williams, B. Goldfain, E. A. Theodorou, and J. M. Rehg, “Vision-based high-speed driving with a deep dynamic observer,” *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1564–1571, 2019.

[33] A. Angelova, L. Matthies, D. Helmick, and P. Perona, “Learning and prediction of slip from visual information,” *Journal of Field Robotics*, vol. 24, no. 3, pp. 205–231, 2007.

[34] J. Biswas and M. M. Veloso, “Episodic non-markov localization,” *Robotics and Autonomous Systems*, vol. 87, pp. 162–176, 2017.

[35] P. Atreya, H. Karnan, K. S. Sikand, X. Xiao, G. Warnell, S. Rabiee, P. Stone, and J. Biswas, “High-speed accurate robot control using learned forward kinodynamics and non-linear least squares optimization,” 2022. [Online]. Available: https://arxiv.org/abs/2206.08487