2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Velocity estimation for motorcycles using image-to-road mapping

Martin Pryde[1], Lamri Nehaoua[1], Hicham Hadj-Abdelkader[1], Hichem Arioui[1]

***Abstract*— The authors propose a visual-inertial approach to estimate the body-fixed lateral velocity of motorcycles traveling along extra-urban roads. The approach comprises the following steps: First, a monocular camera takes video of the road ahead. Key features from sequential images of the road surface are extracted using the Harris corner detector and matching features are identified using the Fast retina keypoint descriptor. The locations of these features on the road surface are determined using a mapping based on an intuitive ray-casting approach. Next, the feature locations on the road, the angular velocity measurements and the optical flow of the feature projection locations on the image plane are used to formulate the ego-motion of the motorcycle as a system of linear equations from which a velocity estimate is solved for using the least-squares method. Finally, this estimate is fused with readings from an inertial navigation system using a Kalman filter to produce a filtered estimate and correct integrator drift. The approach is validated against simulation data generated using BikeSim and the results are compared against state observer approaches and previously published visual-inertial approaches from the authors.**

I. INTRODUCTION

Riders of motorcycles and other Powered Two-Wheeled Vehicles (P2WVs) continue to experience rates of injury and mortality far higher than other motorists [1]. Besides the obvious differences in levels of protection between cars and motorcycles, a contributing factor to this disparity has been the widespread introduction of *Advanced Driver Assistance Systems* (ADAS) into modern cars since the 1990s [2]. With this in mind, researchers at the IBISC laboratory at the University of Paris-Saclay and elsewhere have been developing the technologies required to conceive of equivalent aids for riders, known in industry as *Advanced Rider Assistance Systems* (ARAS).

We are particularly interested in developing an *Electronic Stability Control* (ESC) system for motorcycles. However, doing so is not as trivial as applying existing ESC for cars to motorcycles. In his seminal work *Motorcycle Dynamics* [3], Vitorre Cossalter proves the importance of knowing the front and rear wheel *lateral slip angles* to evaluate the steering behavior of motorcycles in bends. While these angles may be sufficiently estimated in cars using inertial measurements and model-based state observers, they are not so easily estimated in motorcycles. The factors contributing to this challenge are numerous: A motorcycle is inherently unstable due to having only two wheels. The learning curve for riders safely navigating bends is much steeper as they must not only turn the handlebars but also lean into the bends to balance the overturning moment generated by the road-tire interaction which can rapidly destabilize the motorcycle. Finally, whereas a car occupies most of the width of a standard lane, motorcycles permit a far larger envelope of lateral motion leading to much faster possible transients in lateral slip, thus more possibilities for rapid dangerous under and oversteering events.

It is this last challenge which motivates our works to date. In order to accurately estimate the slip angles of a motorcycle, we must first estimate the body-fixed lateral velocity. Previous attempts to estimate this velocity have predominantly focussed on designing state observers based on variations of Robin Sharp's dynamical model first introduced in 1971 [4]. However, while these observers can estimate important states such as lean angle and lean rate, lateral velocity continues to elude them [5][6]. To this end, we have been investigating alternative approaches based on *Visual-Inertial Odometry* (VIO), a technique that utilizes machine vision to identify the locations of features in the surrounding environment as well as measurements from inertial instruments to reconstruct the *ego motion* of a vehicle. VIO is already widely used in the domain of Uncrewed Arial Vehicles (UAVs) [7] and is increasingly finding applications for Uncrewed Ground Vehicles (UGVs) [8][9].

A. Motivation

Previous works from IBISC [10] [11], including some of our own [12], have focussed on using the *Inverse Perspective Mapping* (IPM) to reconstruct bird's eye view images of the road surface from which to identify features for motion estimation. However, with IPM there can be ambiguity in the depth of the *virtual camera* from the scene leading to uncertainty of the feature locations in the real world. Moreover, our implementations of IPM-based ego-motion estimation [13] suffer from nontrivial deviation from the ground truth at transient peaks. Thus, we propose a new approach based on directly mapping the locations of features in the image plane of the motorcycle camera to the road surface. The paper is organized as follows: We begin with a description of the apparatus involved and briefly summarize the feature recognition and matching algorithms used. Next, we derive our point mapping method. We derive the equations of ego-motion for a motorcycle from first principles thus completing the visual component of our VIO. Subsequently, we revisit the Kalman filter-based sensor fusion algorithm used in our previous works thus completing the inertial component. Finally, we present the results of our simulations in BikeSim and compare them to our previous IPM-based approaches as well as other motorcycle velocity estimation approaches.

[1]Authors are affiliated with Laboratoire IBISC, University of Paris-Saclay, Evry-Courcouronnes, France martin.pryde@univ-evry.fr

II. SYSTEM DESCRIPTION

Consider a motorcycle traveling at high speed along an extra-urban road. We propose to attach a monocular camera in front of the steering head fixed to the rear body as shown in Figure 1. Concerning environmental conditions, assume the road surface is smooth and well-illuminated. Furthermore, assume good weather so that no raindrops and little jitter are present in the camera images.

We define two important frames of reference on the motorcycle body. The first is the frame fixed to the center of mass G_r of the rear body of the motorcycle $\mathfrak{R}_{G_r}$. The second is the frame $\mathfrak{R}_c$ aligned with the optical axis of the camera and whose origin is the camera's focal center. For this work, we consider that there is an *Inertial Measurement Unit* (IMU) located at the origin of $\mathfrak{R}_{G_r}$. The orientation of the axes of $\mathfrak{R}_c$ with respect to $\mathfrak{R}_{G_r}$ is chosen so that the x and y directions of $\mathfrak{R}_c$ align with the positive directions of the image plane. This will facilitate derivation later on.

The IMU measures the body-fixed angular velocity $\boldsymbol{\omega}^{G_r}(t) = \begin{bmatrix}\omega_x & \omega_y & \omega_z\end{bmatrix}^{\mathrm{T}}$ as well as the body-fixed acceleration $\boldsymbol{a}^{G_r}(t) = \begin{bmatrix}a_x & a_y & a_z\end{bmatrix}^{\mathrm{T}}$ in frame $\mathfrak{R}_{G_r}$. We assume that the IMU module is programmed with an *Attitude Heading and Reference System* (AHRS) which provides us with the *roll*, *pitch* and *yaw* angles $\boldsymbol{q}(t) = \begin{bmatrix}\phi & \theta & \psi\end{bmatrix}^{\mathrm{T}}$ of $\mathfrak{R}_{G_r}$ with respect to the road.

We choose the *Tait-Bryan* convention for Euler angles so that $\mathfrak{R}_{G_r}$ is oriented in the world frame firstly by a rotation about the local z-axis by yaw, then by pitch about the local y-axis and finally by roll about the local x-axis. For motorcycles, note that it is important not to confuse *roll* with *lean* where the latter is the rotation about the local x-axis of the frame rotated first by yaw. When not leaning, the camera is located at a height h_c above the road. In $\mathfrak{R}_{G_r}$, it is located l_f meters in front and $(h_c - h_r)$ meters above G_r. The camera is also angled downwards towards the road by a *tilt* denoted by μ with respect to $\mathfrak{R}_{G_r}$.

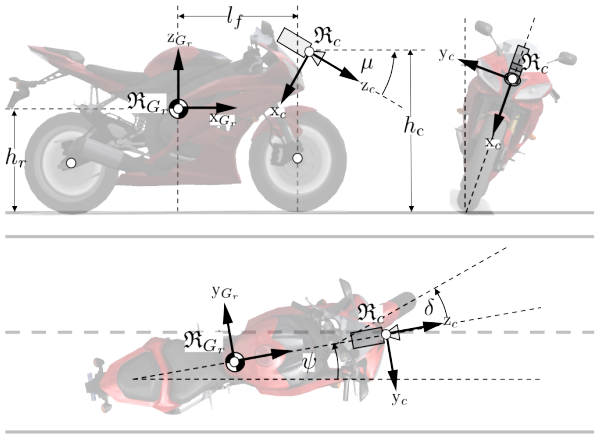


Fig. 1: Illustration of the proposed motorcycle-camera-IMU system. Note the difference in positive xy-directions for $\mathfrak{R}_{G_r}$ and $\mathfrak{R}_c$. The steer angle $\delta(t)$ is shown to clarify that the camera is fixed to the rear body of the motorcycle.

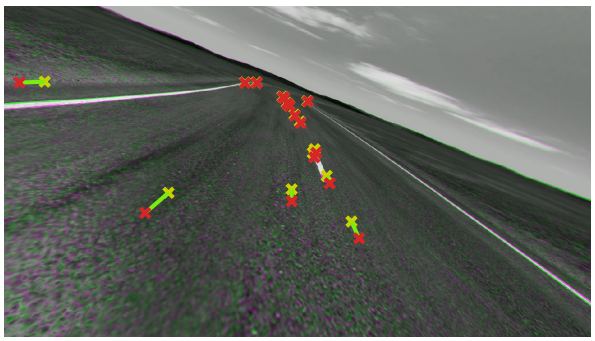


Fig. 2: Sequential images superimposed with matched features highlighted. Note that features tend to cluster around lane markings.

III. FEATURE EXTRACTION

In this context, we refer to *features* as specific areas within multiple images of a single scene that can be identified by computer vision algorithms. These areas typically consist of points, corners, or patterns that strongly contrast with their surroundings. In the past, detecting features in road scenes has been difficult due to the overall uniformity of the scene. However, with advancements in digital camera technology and embedded processing capabilities, it is now hypothesized that enough surface details of roads can be captured to reliably extract features. The feature detection process may work even more effectively on roads with deteriorated lane markings and surfaces that have become uneven over time.

The Harris corner detector [14] is one of the earliest developed and most robust feature detection algorithms. To briefly review, a small window of pixels is passed over the entire image. A window is considered a corner if the intensity of the pixels within it changes significantly in all directions. We use image gradients obtained by convolving with the Sobel kernels to construct a structure tensor and compute a so-called *cornerness* score for each pixel. Finally, we apply thresholding and non-maximum suppression to retain the most significant corners. There are many more modern feature detectors such as the Scale-Invariant Features Transform (SIFT) and the Features from Accelerated Segment Test (FAST). While we may choose to implement one of these in our approach at a later date, a comparison of feature detectors is not the focus of this work.

In order for a corner to be considered a feature, we must be confident that it appears in the image taken at the next time step from our current image. To this end, we make use of *feature matching* algorithms. A popular approach is to convert each corner into a *descriptor* which encodes the differences in pixel intensity within each corner window as a binary vector. Matching features can then be found by simply computing the Hamming distance. Algorithms that use this method include BRIEF and BRISK. For this work, we chose the *Fast Retina Keypoint* (FREAK) [15] due to its ubiquity, though it would be interesting to see how others compare in extracting features from road surfaces.

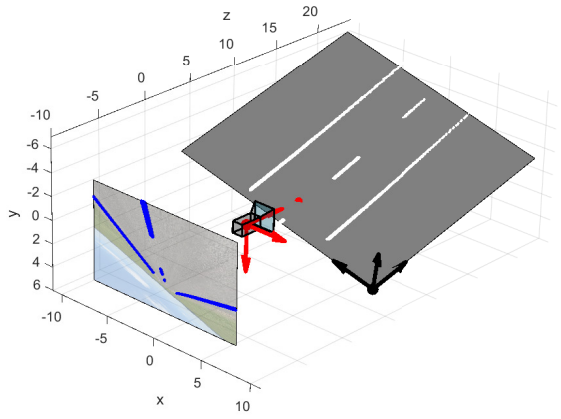


Fig. 3: Illustration of the image to world mapping: The image plane has been re-scaled for illustrative purposes. The lane markings (in blue) are ray-casted from the camera to the world plane. The point in red is the projection of the camera's optical center onto the world plane.

IV. IMAGE TO ROAD MAPPING

Referring to Figure 3, let $\mathfrak{R}_r$ be an inertial reference frame aligned with the road surface. A feature i on the road is located at $\boldsymbol{p}_i^c(t) = \begin{bmatrix} x & y & z \end{bmatrix}^{\mathrm{T}}$ at the origin of $\mathfrak{R}_r$. Recall that in $\mathfrak{R}_c$, z is the distance along the camera's optical axis. The same feature appears in an image taken at timestep t at the pixel location $\mathbf{p}_i(t) = \begin{bmatrix} u & v & f \end{bmatrix}^{\mathrm{T}}$ where f is the camera's focal length. We seek the position of a feature given its pixel location and the extrinsic and intrinsic parameters. For brevity, we drop the dependence on t henceforth.

A. Extrinsic parameters

The *extrinsic* parameters tell us the camera's orientation *pose* in the scene. The rotation matrix $\mathbf{R}_{cr}$ represents the orientation of $\mathfrak{R}_c$ in $\mathfrak{R}_r$ and the translation vector $\boldsymbol{t}_{r,c}^r$ represents the position of the camera above the road in $\mathfrak{R}_r$. Referencing Figure 1 once again, we define the following sequence of elementary rotations: First, a static rotation to align the axes of the camera such that the z-axis is aligned with the optical axis while the x and y-axes are aligned with the image plane, followed by another static rotation about the local y-axis to account for the camera tilt.

$$\mathbf{R}_{cr} = \mathbf{R}_y(\theta)\,\mathbf{R}_x(\phi)\,\mathbf{R}_y(\mu)\,\mathbf{R}_x(\pi/2)^{\mathrm{T}}\,\mathbf{R}_y(\pi/2) \tag{1}$$

Meanwhile, $\boldsymbol{t}_{r,c}^r$ is found by rotating the height of the camera above the road when the motorcycle is upright by the pitch and roll angles similar to the above where $\boldsymbol{e}_3$ is the third column of $\mathbf{I}_3$, the identity matrix.

$$\boldsymbol{t}_{r,c}^r = \mathbf{R}_y(\theta)\,\mathbf{R}_x(\phi)\,(h\boldsymbol{e}_3) \tag{2}$$

B. Intrinsic parameters

The *intrinsic* parameters of the camera tell us about its optical properties. Assuming a pinhole model, the intrinsic parameters can be summarized in the matrix $\mathbf{K}$ where f_u and f_v are the horizontal and vertical focal length parameters relating the breadth and depth of each pixel in the image sensor to f. The location of the camera principal point is given by u_0 and v_0 denoting the horizontal and vertical offsets in pixels from the image origin at the top left corner.

$$\mathbf{K} = \begin{bmatrix} f_u & 0 & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

If the $\mathbf{K}$ is known, one can convert the pixel location of a feature to *normalized image coordinates* by multiplying by the inverse:

$$\bar{\mathbf{p}}_i = \mathbf{K}^{-1}\mathbf{p}_i \tag{4}$$

C. Ray casting

Knowing the extrinsic and intrinsic parameters and assuming a locally flat road, we can map the pixel location of a feature to its position on the road in meters by casting from the feature's location on the image plane, through the camera focal point and finally to the intersection point with the road plane. Begin by defining any point $\boldsymbol{p}_0^r$ on the road in $\mathfrak{R}_r$ which is expected to be in the camera's field of view. Define the position of this point in the camera frame using the extrinsic parameters:

$$\boldsymbol{p}_0^c = \mathbf{R}_{cr}^{\mathrm{T}}\left(\boldsymbol{p}_0^r - \boldsymbol{t}_{r,c}^r\right) \tag{5}$$

Defining the normal vector of the road plane in $\mathfrak{R}_c$ as $\boldsymbol{n}^c = \mathbf{R}_{cr}^{\mathrm{T}}\boldsymbol{e}_3$, we can compute the distance parameter λ defining how far along the ray to cast from the camera to reach the intersection point with the road plane:

$$\lambda = \frac{\left(\bar{\mathbf{p}}_i - \boldsymbol{p}_0^c\right)^{\mathrm{T}}\boldsymbol{n}^c}{\bar{\mathbf{p}}_i^{\mathrm{T}}\boldsymbol{n}^c} \tag{6}$$

Finally, the position of the feature in $\mathfrak{R}_r$ is found using the vector form of the straight line equation and expressed in the road frame using the extrinsic parameters. Note that the vector $\boldsymbol{e}_0 - \bar{\mathbf{p}}_i$ points from the camera focal point towards the feature's pixel location on the image plane.

$$\boldsymbol{p}_i^c = \bar{\mathbf{p}}_i + \lambda\left(\boldsymbol{e}_0 - \bar{\mathbf{p}}_i\right) \tag{7}$$

$$\boldsymbol{p}_i^r = \mathbf{R}_{cr}\boldsymbol{p}_i^c + \boldsymbol{t}_{r,c}^r \tag{8}$$

Repeating the above for all $i \in 1 \ldots n$ features in a given image produces the coordinates of the features in $\mathfrak{R}_r$. Figure 4 illustrates an example of mapped feature points superimposed onto a bird's eye view of the road generated by IPM. Comparing with the same scene in perspective shown in Figure 2, it is intuitive to see where the corners in that figure have been mapped to in the bird's eye view. The lane markings of the same scene are also projected onto the road plane depicted in Figure 3.

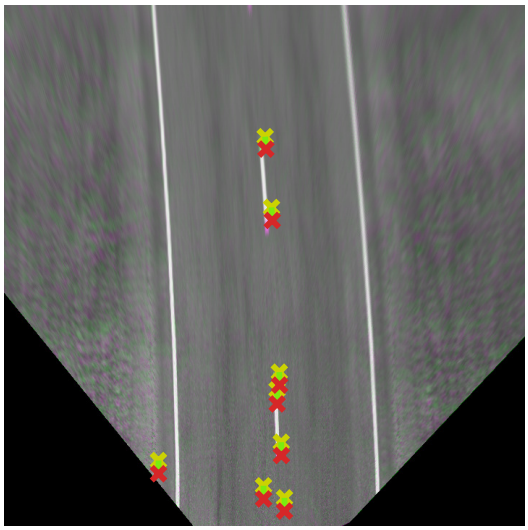


Fig. 4: Feature points remapped to the world plane.

V. TRANSLATIONAL VELOCITY ESTIMATION

We now aim to estimate the motion of the motorcycle relative to the feature positions computed earlier. Consider a feature $\boldsymbol{p}_i^c$ which is visible in the image at two consecutive time steps $k-1$ and k where the time between these two steps is τ, the system sample time. We define the *optical flow* of this feature trivially as:

$$\mathbf{u}_{i,k} = \left(\frac{1}{\tau}\right)(\mathbf{p}_{i,k-1} - \mathbf{p}_{i,k}) \tag{9}$$

Meanwhile, the position of the feature on the road in the camera frame $\boldsymbol{p}_i^c = \begin{bmatrix} x & y & z \end{bmatrix}^{\mathrm{T}}$ is related to the location of its projection on the image plane $\mathbf{p}_i = \begin{bmatrix} u & v & f \end{bmatrix}^{\mathrm{T}}$ by the following relation. Note that if normalized image coordinates are used instead, the dependencies on the f are removed.

$$\mathbf{p}_i = \left(\frac{f}{z}\right)\boldsymbol{p}_i^c \tag{10}$$

Next, we introduce the kinematic relationship between the camera frame and the feature in $\mathfrak{R}_c$. Note how the translational and angular components $\dot{\boldsymbol{p}}_c^c = \begin{bmatrix} \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^{\mathrm{T}}$ and $\boldsymbol{\omega}^c = \mathbf{R}_{cG_r}^{\mathrm{T}}\boldsymbol{\omega}^{G_r}$ are the negative of what they usually are in most classical mechanics textbooks.

$$\boldsymbol{v}^c = -\dot{\boldsymbol{p}}_c^c - \boldsymbol{\omega}^c \times \boldsymbol{p}_i^c \tag{11}$$

Differentiating (10) tells us the relationship between the optical flow and the velocity of the camera in the camera frame [16]. Substituting our definitions for the velocity and feature projection onto the image plane into this equation yields the following where $\mathbf{L}$ is the *interaction matrix* relating velocity to optical flow [17].

$$\mathbf{u}_i = \frac{f}{z^2}\left(z\boldsymbol{v}_c^c - v_z^c\boldsymbol{p}_i^c\right) \tag{12}$$

$$\mathbf{u}_i = \mathbf{L}\begin{bmatrix}\dot{\boldsymbol{p}}_c^c \\ \boldsymbol{\omega}^c\end{bmatrix} \tag{13}$$

Note that the interaction matrix is an underdetermined system of equations relating the $\mathbb{R}^6$ vector of translational and angular velocity of the motorcycle to the $\mathbb{R}^2$ vector of optical flow. This is due to $\mathbf{u} = \begin{bmatrix}\dot{\mathrm{u}}_i & \dot{\mathrm{v}}_i\end{bmatrix}^{\mathrm{T}}$ where the z-term is dropped because we assume the focal length f of the camera never changes.

$$\mathbf{L} = \begin{bmatrix} \frac{f}{z} & 0 & -\frac{x}{z} & -\frac{xy}{f} & f+\frac{x^2}{f} & -y \\ 0 & \frac{f}{z} & -\frac{y}{z} & -f-\frac{y^2}{f} & \frac{xy}{f} & x \end{bmatrix} \tag{14}$$

As mentioned in Section II, we have access to the gyroscope of the IMU. Thus, we can use it to estimate the angular velocity of the camera since they both rest in the rear frame of the motorcycle. Separating the angular component of the interaction matrix from the translational yields the following:

$$\boldsymbol{b}_i = \mathbf{u}_i - \begin{bmatrix} -\frac{xy}{f} & f+\frac{x^2}{f} & -y \\ -f-\frac{y^2}{f} & \frac{xy}{f} & x \end{bmatrix}\boldsymbol{\omega}^c \tag{15}$$

While this system may be under-determined, we can assemble an overdetermined system by stacking $\mathbf{b}$ for all n features detected in the image.

$$\mathbf{b} = \begin{bmatrix}\boldsymbol{b}_1^{\mathrm{T}} & \dots & \boldsymbol{b}_n^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}} \tag{16}$$

Similarly for the remaining translational component of $\mathbf{L}$, we can stack these for all n features to complete our overdetermined system.

$$\boldsymbol{a}_i = \begin{bmatrix} \frac{f}{z} & 0 & -\frac{x}{z} \\ 0 & \frac{f}{z} & -\frac{y}{z} \end{bmatrix} \tag{17}$$

$$\mathbf{A} = \begin{bmatrix}\boldsymbol{a}_1^{\mathrm{T}} & \dots & \boldsymbol{a}_n\end{bmatrix}^{\mathrm{T}} \tag{18}$$

Finally, it is obvious we have a linear system of equations relating which can be solved easily using linear system solvers such as the *Least Squares* method.

$$\mathbf{A}\dot{\boldsymbol{p}}_c^c = \mathbf{b} \tag{19}$$

One more small kinematic adjustment is to express this estimate of the motorcycle's velocity in the rear body rame $\mathfrak{R}_{G_r}$ where the IMU is located. Referring to the transformations introduced Section IV-A, we define the following transformation to convert the estimated velocity to the rear body frame:

$$\mathbf{R}_c^{G_r} = \mathbf{R}_y(\mu)\,\mathbf{R}_x(\pi/2)^{\mathrm{T}}\,\mathbf{R}_y(\pi/2) \tag{20}$$

$$\boldsymbol{t}_{c,G_r}^{G_r} = l_f\boldsymbol{e}_1 + (h_{c0} - h_{G_r}) \tag{21}$$

The transformation on the velocity is performed as per the definitions of the velocity adjoint mapping [18].

$$\boldsymbol{v}^{G_r} = \mathbf{R}_{cG_r}^{\mathrm{T}}\boldsymbol{v}^c - \mathbf{R}_{cG_r}^{\mathrm{T}}\left(\boldsymbol{t}_{c,G_r}^{G_r} \times \boldsymbol{\omega}_c^c\right) \tag{22}$$

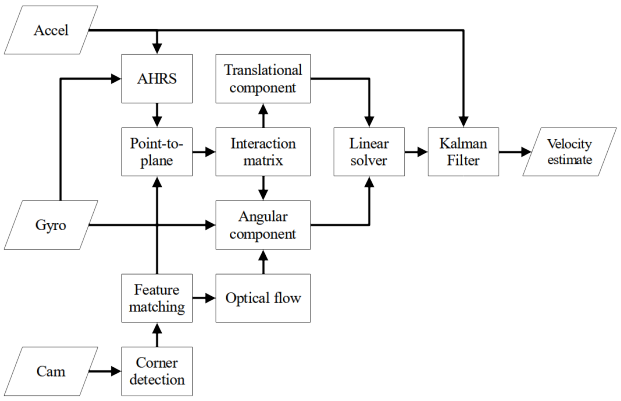


Fig. 5: Block diagram of the proposed visual-inertial velocity estimator

VI. SENSOR FUSION

From this point onwards, the velocity estimation algorithm is the same as published in our previous works [12]. To review, the inertial component of our visual-inertial estimator consists of integrating the measurements from the IMU's accelerometer to obtain an additional estimate of the motorcycle's velocity. This process is termed *dead reckoning*. A major drawback of this process is the accumulation of errors over time causing the estimate to drift away from the ground truth.

We propose to mitigate this drift by fusing the dead reckoned velocity with the estimate obtained from visual odometry using a *Linear Time-Invariant*LTI Kalman filter. The prediction step is performed by integrating the accelerometer reading to obtain an *a priori* estimate of the motorcycle's velocity. The measurement is obtained from the visual odometry algorithm. We select the model matrices of our Kalman filter as so: The state transition matrix $\boldsymbol{F} = \mathbf{I}_2$, the control matrix $\boldsymbol{B} = \tau\mathbf{I}_2$ and the measurement matrix $\boldsymbol{H} = \mathbf{I}_2$. Recall from Section V that τ is the system sample time. We denote the prediction and measurement covariance matrices as $\boldsymbol{Q}$ and $\boldsymbol{R}$ respectively and define their values in the next section. The steps of the full estimator described in this work are shown in Figure 5.

VII. EXPERIMENTAL RESULTS

We evaluated the performance of our algorithm using the BikeSim mechanical simulation software for motorcycles. The estimator was implemented in MATLAB using image and sensor simulation datasets from BikeSim. The geometric parameters for the camera were set to $h_c = 1.372\,\mathrm{m}$ and $\mu = 15°$. The physical parameters of the motorcycle were left to their defaults in the *Big Sports*-type motorcycle in BikeSim.

The resolution of the camera was set to standard HD with a horizontal *Field of View* (FoV) of $80°$. While the wide FoV is necessary to capture long distances where the road diverges, it also introduces the possibility of points outside the road and even points not in the road plane (e.g. the sky) being extracted as features. To mitigate this, the estimator was implemented in a manner such that points outside of a fixed *Region of Interest* (ROI) in front of the motorcycle on the plane are removed from the feature set.

Following a noise variance analysis of the visual odometry component of our system, we set the prediction and measurement covariance matrices to $\boldsymbol{Q} = (5\mathrm{E}{-}03)\,\mathbf{I}_2$ and $\boldsymbol{R} = 100\mathbf{I}_2$ respectively. To simulate the phenomenon of integrator drift, we introduced *Additive Gaussian White Noise* (AGWN) to the accelerometer measurement with a *Signal to Noise Ratio* (SNR) of 20 ($\sim 13\,\mathrm{dB}$). Note that even low-cost *Micro Electro Mechanical Systems* (MEMS) accelerometers have SNRs ranging from $60\,\mathrm{dB}$ to $90\,\mathrm{dB}$, albeit with frequencies thousands of times higher than a typical camera. This value was chosen to make the dead-reckoned estimate diverge quickly and test our vision system's ability to keep the estimate close to the ground truth while economizing on dataset size.

A. Scenarios

We evaluated the performance of our algorithm in four scenarios. In the first scenario, the motorcycle *slaloms* between two lanes of a dual carriageway at $50\,\mathrm{km/h}$. In the second and third scenarios, the motorcycle navigates an S-shaped bend at $80\,\mathrm{km/h}$ and $110\,\mathrm{km/h}$ respectively. In the final scenario, the motorcycle performs a *Double Lane Change* (DLC) maneuver on a straight road. The variety of speeds was chosen to assess the estimator's ability to cope with small and large optical flows of features. The Slalom maneuver was chosen to assess the estimator's ability to cope with large magnitudes of lateral velocity. The two S-bend scenarios on the other hand reflect the most pertinent driving situations for our ARAS development. Finally, the DLC could be used to evaluate the estimator's ability to cope with high-speed overtaking situations.

B. Results

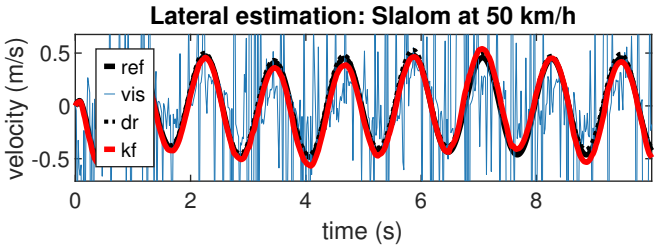
We now present our experimental results from BikeSim. The plots show the ground truth velocity (ref), the estimate obtained from the just visual component on the estimator (vis), the estimate obtained by dead-reckoning the accelerometer measurement (dr) and finally the Kalman-filtered estimate (kf).

TABLE I: *Root Mean-Squared Error* (RMSE) values obtained for each scenario. All values are in $\mathrm{m/s}$

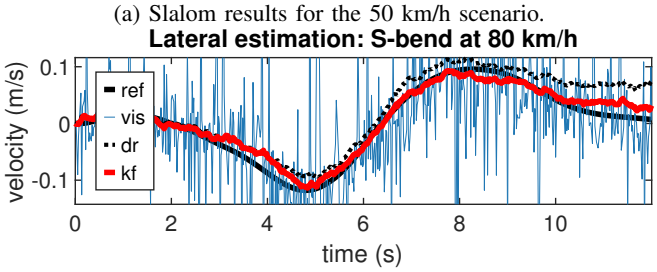
Scenario	Slalom	S-bend 80	S-bend 110	DLC
v_{lat}	0.0493	0.0133	0.0128	0.0135
v_{long}	0.3784	2.5247	1.1772	2.2064

C. Discussion

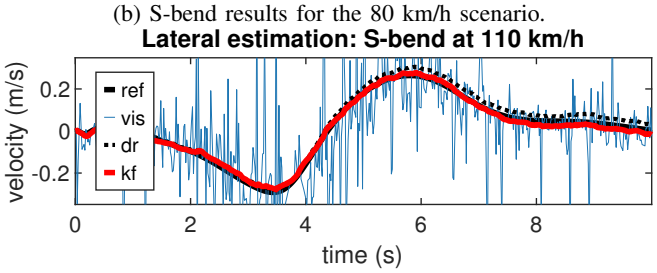
The preliminary assessment of our results is encouraging. We observe that our results compare favorably with the Kalman-based approach of Teerhuis and Jansen [19], even though our estimator lacks an a priori model. When compared against observer-based designs, our estimate during



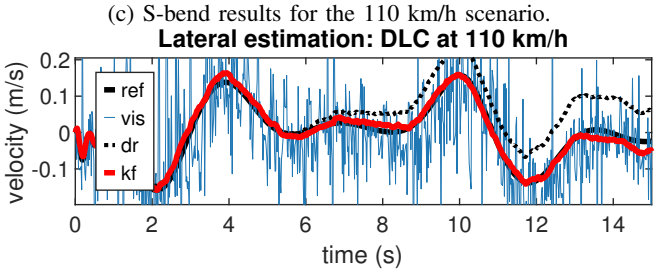
(a) Slalom results for the 50 km/h scenario.



(b) S-bend results for the 80 km/h scenario.

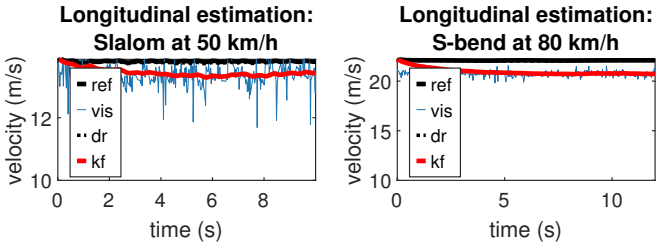


(c) S-bend results for the 110 km/h scenario.



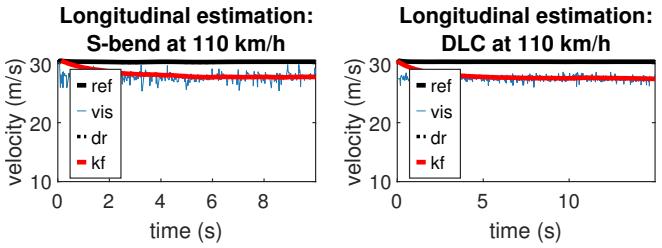
(d) DLC results for the 110 km/h scenario.

Fig. 6: Lateral velocity estimation results.



(a) Slalom results for the 50 km/h scenario.

(b) S-bend results for the 80 km/h scenario.



(c) S-bend results for the 110 km/h scenario.

(d) DLC results for the 110 km/h scenario.

Fig. 7: Longitudinal velocity estimation results.

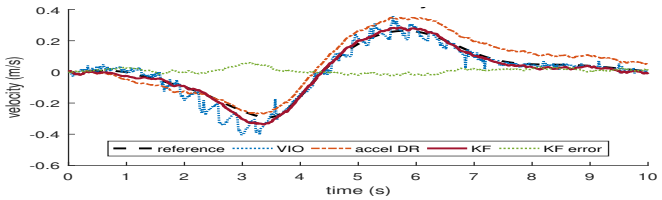


Fig. 8: Extract of the S-bend lateral velocity estimation results from [12] based on an IPM & ICP approach

the peak regions of the DLC outperforms the estimation by UI-HOSM [5], while performing the same maneuver. This comparison is particularly important since the results in [5] were also obtained using BikeSim. In [20], the observer's lateral velocity estimation absolute error magnitude during a DLC using a nominal LPV model was slightly over $0.12\,\mathrm{m/s}$. In contrast, our DLC result has an error of less than $0.014\,\mathrm{m/s}$, outperforming the LPV observer in that paper. We must note that, although the control inputs of the scenarios in [20] are generated using BikeSim, they are validated against a mathematical model in MATLAB. Thus, we can conclude that our estimator outperforms theirs even under the best conditions. In [6], a highly sophisticated multi-model observer based on Tagaki-Sugeno and Linear Matrix Inequalities (LMI) techniques is presented, demonstrating the most promising results for observer-based lateral velocity estimates to date. Although their observer reaches the correct final value in steady-state periods, it fails to reconstruct the waveforms of transients compared to our approach.

Beginning with the longitudinal velocity estimates in Figure 7, we observe in each case that there is a steady-state underestimation. It is obvious that this error increases as speed increases. This is a known issue of optical flow methods where large pixel displacements tend to degrade the performance of estimation. It is reasonable to assume that the longitudinal velocity can also be inferred from the engine speed measured by the vehicle speedometer.

Paying particular attention now to the S-bend results at $80\,\mathrm{km/h}$, we observe that noise at this speed is noticeably higher than for other scenarios and even the same scenario at a higher speed. We suspect that this is due in part to the well-known phenomenon of motorcycle *wobble* described in the stability analyses of both Sharp and Cossalter.

Despite this, there is considerable noise evident in all of our visual-inertial estimation measurements that arises from sources not previously discussed. A Fourier transform of the lateral velocity vision estimates' signal power spectrum reveals no particular frequencies contributing to the noise. Thus, we conclude that the noise present in the vision process is of the AGWN type. Dealing with outliers is a well-known challenge in feature-based motion estimation. One common solution is to use *RAndom Sample Consensus* (RANSAC) to detect them [21]. In future research, it would

be worthwhile to explore the impact of outliers on our results using this approach. In addition, our ability to obtain accurate results is limited by the BikeSim software itself, as the physics engine's simulated IMU readings and the visualizer's simulated image stream are asynchronous. This results in the sample time in (9) effectively becoming non-constant thus significantly corrupting our measurement. While increasing the sample rate of the video stream and the resolution of the camera may be a possible way to reduce measurement noise, we should also investigate if alternative feature detection algorithms, such as those mentioned in Section III lead to a reduction in noise.

Figure 8 shows the lateral velocity results from [12] for the same S-bend scenario at $110\,\mathrm{km/h}$ also tested in this work and shown in Figure 6c. This earlier attempt at a visual-inertial approach to estimating motorcycle velocity used point clouds of the center lane markings viewed from a bird's eye view generated by IPM as features and the *Iterative Closest Point* (ICP) algorithm to perform ego-motion estimation. We noted then that this approach suffers from two main drawbacks. Firstly, a low-frequency oscillation is present due to the sudden change in mean point cloud position from the sudden exit or entry of a new dashed lane marking. Second, we observe a significant deviation in the VIO measurement from the ground truth in the first trough of the waveform. While we cannot be certain of the source of this error, the ambiguity in virtual camera depth in the IPM formulation mentioned in Section I-A may have played a role. Returning to Figure 6c, we observe that this deviation is no longer present thus satisfying a key aim of this work.

VIII. CONCLUSIONS AND FUTURE WORKS

In this paper, we have presented a novel approach to estimating the velocity of a motorcycle traveling at high speeds using visual-inertial odometry. In order to address the shortcomings of previous approaches based on IPM, here we have used an intuitive mapping based on ray-casting to locate features on the road surface and formulated ego-motion as a linear system of equations that can be efficiently solved using least-squares or other linsolve algorithms.

A. Conclusions

A key aim of this work was to overcome errors in the peaks of transients present during our previous IPM approaches. Our results demonstrate that this new approach achieves this aim. Moreover, we have shown that our estimator outperforms comparable state-observer approaches.

B. Future Works

In future work, we plan to address the noise issues still present in our vision measurements. We hypothesize that our approach could be extended to estimate the front and rear wheel slip angles characterizing over and understeering situations thus providing the basis of an ESC-like ARAS. The particular structure of the ego-motion portion of our estimator potentially also allows for visual servoing control of a motorcycle. Further into the future, we hope to gain access to real-world datasets and eventually implement our estimator on hardware.

REFERENCES

[1] ONSIR, "Le parc automobile des ménages," Observatoire national interministériel de la sécurité routière," resreport, Sept. 2021. [Online]. Available: https://www.onisr.securite-routiere.gouv.fr/etudes-et-recherches/vehicules/parc-des-vehicules/le-parc-deux-roues-motorises-des-menages

[2] A. D. Furlan, T. Kajaks, M. Tiong, M. Lavallière, J. L. Campos, J. Babineau, S. Haghzare, T. Ma, and B. Vrkljan, "Advanced vehicle technologies and road safety: A scoping review of the evidence," *Accident Analysis & Prevention*, vol. 147, 2020.

[3] V. Cossalter, *Motorcycle Dynamics*, 2nd ed. Lulu, Oct. 2006.

[4] R. S. Sharp, "The Stability and Control of Motorcycles," *Journal of Mechanical Engineering Science*, vol. 13, no. 5, pp. 316–329, 1971.

[5] L. Nehaoua, D. Ichalal, H. Arioui, J. Davila, S. Mammar, and L. M. Fridman, "An unknown-input hosm approach to estimate lean and steering motorcycle dynamics," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 7, pp. 3116–3127, Sept. 2014.

[6] M. Fouka, L. Nehaoua, H. Arioui, and S. Mammar, "Interconnected observers for a powered two-wheeled vehicles: Both lateral and longitudinal dynamics estimation," in *2019 IEEE 16th International Conference on Networking, Sensing and Control (ICNSC)*, May 2019, pp. 163–168.

[7] P. Corke, "An inertial and visual sensing system for a small autonomous helicopter," *Journal of Robotic Systems*, Jan. 2004.

[8] B. Liang and N. Pears, "Visual navigation using planar homographies," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, vol. 1, May 2002, pp. 205–210 vol.1.

[9] X. Song, L. D. Seneviratne, and K. Althoefer, "A Kalman Filter-Integrated Optical Flow Method for Velocity Sensing of Mobile Robots," *IEEE/ASME Transactions on Mechatronics*, vol. 16, no. 3, pp. 551–563, 2011.

[10] P.-M. Damon, H. Hadj-Abdelkader, H. Arioui, and K. Youcef-Toumi, "Inverse perspective mapping roll angle estimation for motorcycles," in *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, Nov. 2018, pp. 349–354.

[11] O. Alrazouk, A. Chellali, L. Nehaoua, and H. Arioui, "A geometric approach for estimating sideslip angle for powered two-wheeled vehicles," in *The 17th International Conference on Control, Automation, Robotics and Vision (ICARCV 2022)*, 2022.

[12] M. Pryde, L. Nehaoua, H. Hadj-Abdelkader, and H. Arioui, "Visual-inertial lateral velocity estimation for motorcycles using inverse perspective mapping," in *The 17th International Conference on Control, Automation, Robotics and Vision (ICARCV 2022)*, Dec. 2022.

[13] H. A. Mallot, H. H. Bülthoff, J. J. Little, and S. Bohrer, "Inverse perspective mapping simplifies optical flow computation and obstacle detection," *Biological Cybernetics*, vol. 64, no. 3, pp. 177–185, 1991.

[14] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of the Alvey Vision Conference*, 1988.

[15] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: Fast Retina Keypoint," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 510–517.

[16] E. Trucco and A. Verri, *Introductory techniques for 3-D computer vision*, M. Horton, B. M. de Leon, I. Zucker, and T. Robbins, Eds. Prentice-Hall, 1998.

[17] F. Chaumette and S. Hutchinson, "Visual servo control, Part I: Basic approaches," *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006. [Online]. Available: https://hal.inria.fr/inria-00350283

[18] K. M. Lynch and F. C. Park, *Modern Robotics*, 1st ed., C. U. Press, Ed. Cambridge University Press, May 2017.

[19] A. P. Teerhuis and S. T. H. Jansen, "Motorcycle state estimation for lateral dynamics," *Vehicle System Dynamics*, vol. 50, no. 8, pp. 1261–1276, 2012.

[20] P. M. Damon, H. Dabladji, D. Ichalal, L. Nehaoua, and H. Arioui, "Estimation of lateral motorcycle dynamics and rider action with Luenberger observer," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Nov. 2016, pp. 2392–2397.

[21] D. Scaramuzza, "1-point-RANSAC structure from motion for vehicle-mounted cameras by exploiting non-holonomic constraints," *International Journal of Computer Vision*, 2011.