Abstract

This undergraduate research was conducted because although there are multiple static RGB-D, stereo, and ego-motion datasets, none specifically address the problem of motion parameters and object velocity estimation. We introduce I-MOVE, the first publicly available RGB-D/stereo dataset for estimating velocities of independently moving objects. The dataset features various outdoor and indoor scenes of single and multiple moving objects. Compared to other datasets, a unique property of I-MOVE is that the 3D position and 3D velocity for each object is supplied for a variety of different settings / environments and objects / motions. The dataset includes training and test sequences captured from four different RGB-D camera views and three 4K-stereo setups. The data are also time synchronized with dual Doppler radars to provide velocity ground truth. Multiple scenes are designed for high-quality ground truth computations with increasing levels of complexity. The I-MOVE dataset also includes complex scenes from moving pedestrians to multiple flying drones captured with the seven stereo cameras. We look forward to the constructive feedback on the idea of the dataset and its collection process.

Introduction

“What we see depends mainly on what we look for.” John Lubbock, The Beauties of Nature and the Wonders of the World We Live in, 1892.

The above quote has stood the test of time in the field of computer vision. In the realm of still imagery, several problems have and continue to be addressed, such as image classification and object detection. Similarly, with the rise in popularity of videos, problems such as tracking (Atev et al. 2005; Sadeghian, Alahi, and Savarese 2017; Jiang et al. 2018; Kim, Li, and Rehg 2018), localization and mapping (Zhang et al. 2018; Brosh et al. 2019), action recognition, as well as sentiment analysis have been identified (Chang et al. 2019; Piergiovanni and Ryoo 2019). In this undergraduate work, we present the relatively unexplored task of motion parameter estimation. Even though motion parameters are useful for a large variety of applications, estimating them from videos has not been studied extensively. Motion parameters are a necessary component in numerous applications such as robotic navigation (Chuang et al. 2018; noa) and collision detection (Atev et al. 2005; Gandhi and Trivedi 2006; Heyman 2019). Because having related data to this problem has become so necessary, many synthetic environments have been created (Fei et al. 2019; Zamora et al. 2016; Fan et al. 2018). These synthetic environments have greatly helped people approach the problem, but in the unconstrained real world environments these tasks are much more complicated than the research environments. In problems such as collision detection it is not only necessary to take into account your directional velocity but that of other objects as well. In order to accomplish this it is required that you have the three dimensional directional velocity of each of the objects. Similarly, in robotics if one wishes to enable a robot to interact with an independently moving object (ex flying ball or frisbee or independent drone), the motion parameters of these objects need to be accurately estimated in order to understand the trajectory. Another important application area for motion parameter estimation is sports. In numerous sports performance analysis of athletes, relies on velocity and acceleration information. Most obviously, sports where speed is the main component (running, biking, swimming, etc.), but also for sports such as weightlifting where the athletes are looking for their lift force and acceleration in order to calculate the best feasible posture or lift technique. Similarly, motion parameters may also be useful for training purposes in various sports such as skiing, snowboarding or skateboarding.

While most of the constrained application areas could either document motion parameter information using specialized sensors, such as IMUs (Inertial Measurement Unit) or from egocentric videos, these are not viable for unconstrained scenarios because both of these approaches need to have access to the object in motion. With a task such as estimating the instantaneous velocity of a flying ball neither IMUs nor egocentric videos may be used (Chatzitofis, Zarpalas, and Daras 2018; Chatzitofis et al. 2013; Einfalt, Zecha, and Lienhart 2018; Lee and Kitani 2016). Since views from cameras can be easily accessible for such problems they become the logical choice to create a more useful and robust method for motion parameter estimation.

Since, to the best of our knowledge none of the current datasets provide velocity estimation information or other vital motion parameter ground truth for such complicated tasks, we introduce a new dataset (I-MOVE). Our dataset focuses on aiding motion parameter estimation for outdoor and
indoor objects of various kinds. In order to enable research in incremental steps for such a hard problem our dataset includes three types of vision sensors, providing RGB, Stereo and RGB-D data. The ground-truth velocities are obtained via Doppler radars. The data is also collected in sufficient amounts from varied sources of cameras to enable training of supervised deep learning methods. The objects also vary throughout the scenes, for example some scenes are based upon a person’s movement while others are designed for easier predictable movements, e.g. a rolling ball or pendulum as can be seen in Figure 1. Unlike any of the existing datasets, we also provide ground truth measurements of 3D velocity parameters for each moving object in each scene using Doppler radars. Depending on the object’s path / movement we will use either one or two Doppler radars (for more unpredictable movements where triangulation is necessary to estimate the velocity). In order to validate the accuracy of the Doppler radars we use them in simple scenes, where the velocity information can be easily verified by using the laws of physics. Such scenes include rolling objects down inclined planes, motion of a pendulum, and falling/lying objects.

**Related Datasets**

Vision based velocity estimation has been studied for decades (Hinedi 1988; Nakazawa, Ishihara, and Inooka 2003; Mahapatra and Mehrotra 2000). In recent years, with application of computer vision algorithms to the domains of robotics (Xia et al. 2018; Hua et al. 2018) and autonomous driving (cars (Fangjun Jiang and Zhiqiang Gao 2000; Kampelmhler, Miller, and Feichtenhofer 2018) and drones (Chuang et al. 2018)), the number of works attempting to estimate motion parameters has grown dramatically (Coskun et al. 2017; Prez-Ortiz et al. 2003). As a result the need for these datasets has also grown greatly (Zhu et al. 2018; Sturm et al. 2012; Kim et al. 2018; Chen, Jafari, and Kehtarnavaz 2015; Sigal, Balan, and Black 2010). Many of these works differ in the primary purpose of the dataset and the objects for which motion parameters are estimated. In this section we will first recognize the datasets that either are aimed for motion parameter estimation or a related task. These datasets generally either use RGB data or data acquired from non vision systems. Then we discuss the most similar datasets within the RGB-D realm.

Figure 1: **Setup for Dataset Collection** The above picture describes the setup used for the data collection process for a moving pendulum, which is just one of the various moving objects in our dataset. Our dataset aims to allow vision-based estimation the velocity of moving objects from any of the three types of sensors. The first set are three pairs of high-resolution stereo cameras, 4K hardware synchronized Go-pros in custom mounts, with 2.92mm, 4.3 and 5.2mm lenses respectively. The second set are Intel RealSense RGB-D cameras (415 and 435), with active illumination. The third sensor is a passive stereo system (ZED). The final video is a grayscale active illumination depth sensor (MYNT Eye S). The same setup is used for data collection on all our moving objects. In order to acquire the ground truth velocity we use one or two Doppler radars with varying positions relative to the moving object(s). The doppler radars provide the instantaneous velocity of the pendulum as depicted in the bottom plot of the figure. Since the pendulum is moving towards and away from the radar it provides a sinusoidal instantaneous velocity. For further details on the setup please refer Section.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity Ground Truth</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Moving Object(s)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Any Specific Motion Parameter Ground Truth</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Images / Video Provided</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Indoor Images</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Outdoor Images</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Multiple Cameras Used</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Multiple Perspectives of Same Object</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>RGB-D</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Passive STEREO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 1: COMPARISON OF THE CURRENTLY AVAILABLE DATASETS

Above we summarize some of the currently available datasets. It should be observed that none of the available datasets provide a velocity ground truth which is a big contribution of our I-MOVE dataset. The closest dataset to ours is the HumanEva dataset which unlike ours only contains humans.

**RGB or Motion Parameter Only Datasets**

The Human Activity Recognition dataset (Anguita et al. 2013) provides potentially useful data to address the motion parameter estimation problem. The dataset contains smartphone accelerometer information collected by numerous people performing various tasks such as sitting, walking, and going up stairs. However, this dataset contains no images / video and was created with the intention of creating a model that could predict activity solely from the accelerometer information. Since the aim of the problem presented in this paper is to estimate the motion parameters of an object from a video, this dataset cannot be utilized for its addressal.

Another human activity recognition based dataset is the UTD-MHAD (Chen, Jafari, and Kehtarnavaz 2015) which contains both IMU and video information. This dataset contains 27 actions performed by 8 subjects (4 females and 4 males). Each subject repeated each action 4 times. The actions, such as knock on door, sit to stand, and stand to sit, are fairly limiting in movement, and hence do not make them as desirable to estimate motion parameters.

Another interesting dataset is the HumanEva dataset (Sigal, Balan, and Black 2010), which is a synchronized video and motion capture dataset. It consists of 4 subjects performing a set of six predefined actions three times (twice with video and motion capture, and once with motion capture alone). This dataset was intended to be used to improve existing three dimensional pose estimation, but it may also be used for motion parameter estimation.

**RGB-D Datasets**

While there are a variety of RGB-D datasets, to the best of our knowledge there is no RGB-D dataset that contains velocity ground truth to accurately evaluate an algorithms performance. The most similar dataset is the one proposed in the paper, A Benchmark for the Evaluation of RGB-D SLAM Systems (Sturm et al. 2012). The dataset contains the color and depth images from a Microsoft Kinect sensor along with the ground-truth trajectory of the sensor. The ground-truth trajectory was obtained from a high-accuracy motion-capture system with eight high-speed tracking cameras plus the accelerometer data from the Kinect. However, since the Kinect has limited performance in outdoor environments, the dataset was restricted to indoor use only. Moreover, the dataset only contained a single type of object and hence even if someone would attempt to create a system for motion parameter estimation on this dataset it may not generalize well on other objects. The DIML RGB-D Dataset (Kim et al. 2018) also contains data collected with a Kinect, however this database does include indoor and outdoor video in addition to object segmentation making it more plausible to conduct tests for motion parameter estimation purposes. But this dataset too does not contain any velocity ground truth. As well as since the dataset only contains single camera views any system created to estimate motion parameters on this dataset may not translate well to data from a different camera source.

**The I-MOVE Dataset**

Though our dataset collection process is ongoing, we have completed our initial round of dataset collection, which included deciding upon the sensors and the conditions for data collection as well as recording the initial physics based setups to mathematically verify the quality and plausibility of our velocity estimation. Through this paper we attempt to get a feedback / input from the computer vision community on any shortcoming or missing aspects in our initial setup.
While the HumanEva dataset uses humans as the moving subjects whose motion is recorded with motion capture, the dataset is primarily aimed for key point tracking (Fig 2(a)). These keypoints may be used for motion parameter estimation but the limited motion of the subjects does not make it interesting enough for motion estimation task. Moreover, the dataset is restricted to indoors not providing enough lighting variations for a good relevance to the real world scenarios. To overcome the short coming of the HumanEva dataset we propose I-MOVE where the subjects are captured in an unconstrained environment with a variety of movements. As visible from the Fig 2(b) and 2(c). The variations in the type of camera also provide considerable variations to the same scenes, these variations may also prove useful for training a deep network based approach.

Compared to the reviewed datasets, I-MOVE is unique in the following ways:

(a) To the best of our knowledge, this dataset is the first to focus on and provide object motion parameters.

(b) I-MOVE also contains a variety of objects in both indoor and outdoor scenes.

(c) Each scene is captured with a variety of cameras from different viewing angles. This variety of data provides the necessary means for developing more robustness approaches.

(d) We also provide the ground truth velocity for each moving object in the scene using doppler radars.

(e) The performance of the doppler radars is validated in controlled experiments where the results can be verified using basic laws of physics.

The velocity ground truth is also thoroughly proofed and tested to ensure accuracy with physics based examples and settings to allow completely mathematical based calculations to be done by hand and compared against.

Setup

The apparatus used for data collection was meticulously crafted to ensure the most reliable results in various locations / scenes. The seven cameras and two radars were mounted on a 14 gauge angle bar as can be seen in Figure 1, this allows us to adjust the various heights of the apparatus (cameras) as needed to give us more accuracy and reliability when collecting data on an uneven surface.

The cameras and radars were mounted identically at each location such that each camera’s individual abilities can also be evaluated (range, quality, accuracy, etc.) in different settings. The cameras used were three GoPro Hero 3 stereo rigs (six GoPros in total because there are two GoPros per stereo setup), along with two Intel RealSense cameras (a 415 and 435), a ZED camera, and lastly a MYNT Eye S. The radars used were OmniPreSense doppler radars, which provide velocity of objects within their 78° wide beams.

Due to the wide variety in object size, scene layouts, and environments where data was collected, the apparatus was created to accommodate these differences. The cameras all have a different field of view and so the order and spacing of them was vital to collect the data as best possible. Two of the GoPro setups have modified lenses giving them a field of view of 54.1° and 64.7°, respectively. The standard / unmodified GoPro stereo set up has a horizontal field of view of 122.6°. The Intel RealSense 435 has a field of view of 85° horizontally, while the RealSense 415 version has 63°, and finally, the ZED is capable of 90° viewing horizontally. We also had additional depth information recorded with a MYNT Eye S (that has a 122° field of view horizontally) to allow for better camera to camera comparisons; however, the MYNT we used is not RGB / only monochromatic.

Given our purposeful placement based on the field of view of the cameras, see Fig. 3, we were able to obtain a two foot spacing between each camera allowing for fairly significant
**Calibration and Synchronization of Sensors**

Intrinsic and extrinsic distortion parameters of the sensors were found using the checkered board approach commonly done with OpenCV (Romero-Ramirez, Muoz-Salinas, and Medina-Carnicer 2018; Datta, Kim, and Kanade 2009). Due to the variation in lens and distortion for every single camera used in our rig, it is necessary that the OpenCV calibration approach is done separately for each camera at each location or every time there is a significant change in lighting or background. For some of the cameras such as the ZED and Intel RealSense, calibration options were available within the SDKs, so these were used when possible. In addition to calibration, synchronization was also essential due to the fact that the same motion ground truth was used for different cameras and perspective. For this reason it is also crucial to ensure the radars providing the ground (that is not obtainable via physics based setups / equations) are time synced with the cameras so that the ground truth can be accurately applied to the appropriate frame from each stereo camera setup. In order to do this we had the same data collection device (an HP XPS 13) used to collect the information from both OmniPreSense radars also to collect the Intel RealSense 435 data. Both radars are set to return their timestamp information in addition to the speed data and these timestamps are synchronize with the Intel RealSense 435’s data. This synchronization allows us to use a flash event, where we utilize a camera flash that lasts 4 milliseconds allowing the moment to be visually captured by all the stereo cameras and using the frame(s) with flash to appropriately sync the velocity information to each frame.

### Data Collection

Our data collection process was intended to include a significant variety in objects, object motion, object velocity, scenes / environments and lighting, while still ensuring maximum accuracy in the ground truth for segmentation and velocity of the object. The dataset was also designed to allow the same ground truth to be applicable to multiple views. The six cameras used ensure a fairly significant variety in camera perspective / viewing location in addition to the difference in field of view. The variety in cameras also allows the dataset user to compare the performance between them if they wish.

Currently, the dataset features ten different objects: person, car, dog, skateboarder, skateboard, biker, ball, drone, pedestrian and RC Car. These objects differ widely not only in shape and size but also motion paths making them suitable to train and test upon. We have numerous scenes that involve the different objects in their respective environments with their frequent movements and motion. However, in an attempt to ensure that a model may accurately learn the motion parameters regardless of the object itself the I-MOVE dataset also contains objects traveling on the same path with an easily estimated velocity using physics. For example a variety of objects have been dropped, rolled down a ramp with a constant rolling base object, or swung on a pendulum. I-MOVE’s variety in objects, scenery, and motion parameters/movement paths is designed to help create the most robust velocity estimation models yet.

### Calculation and Verification of Velocity

The main purpose of this dataset is to provide the necessary data for better prediction of motion parameters, in particular the velocity of an object. Choosing the point of the object in which to document the velocity of is itself is often non-trivial. For example, when a person is walking they have multiple components (legs, arms, torso) moving at different velocities. In order to make our dataset as useful as possible for multiple applications and objects we have attempted to find the ground truth velocity for the center point of each object being tracked. Because of this, it is especially vital that the ground truth is as accurate as possible. To ensure the velocity of each object is correct we have additional purely physics based scenes put in place, some of these include dropping an object, rolling a ball (or rollable object) down a ramp, and swinging a pendulum with the object attached at the end.

With these known physics based environments it becomes possible to use physics equations to find the instantaneous velocities for each set up, and to use this information to help perfect the radar setup helping ensure that the triangulation of the radars used for other setups is accurate. This fine tuning of the radars allows us to provide more accurate velocity ground truths for the non-easily physics calculated.

---

<table>
<thead>
<tr>
<th>S-GoPro</th>
<th>S-GoPro</th>
<th>S-GoPro</th>
<th>RealSense 435</th>
<th>RealSense 415</th>
<th>ZED</th>
<th>MYNT Eye S</th>
</tr>
</thead>
<tbody>
<tr>
<td>(122.6°)</td>
<td>(64.7°)</td>
<td>(54.1°)</td>
<td>(83°)</td>
<td>(85°)</td>
<td>(90°)</td>
<td>(122°)</td>
</tr>
</tbody>
</table>

**Figure 3: Schematic for Horizontal Field of View of Each Camera** Each box represents a 1 foot x 1 foot square and as can be seen in the figure the cameras are placed two feet apart horizontally. The figure also shows that object has to be slightly less than five feet away from the center camera (Intel RealSense 435) in order to be in the field of view of all the cameras.
set ups. The equations used for each of the setups required the more complex instantaneous velocity calculations to be used as opposed to the more common final velocity equations. This is because we wanted to obtain velocity for each frame/image within the videos collected.

The velocity data for the object drop was computed using the commonly known equation:

\[ V = \frac{1}{2} gt^2 \]  

(1)

For this equation \( g \) (gravity) is 9.8 meters per second squared and \( t \) is the time since object was released. Because all cameras are returning time stamp information the velocity is easily calculated by pinpointing the timestamp of the moment the object was released and using the time difference between that frame(s) and the future frames in which the ball is falling as \( t \).

In order to accurately calculate the rolling ball / object’s instantaneous velocity a more complex approach had to be used. The ramp itself has friction with the ball or rolling object so in order to make the ramp have as little friction a metal surface was placed over the wooden ramp supports. This reduction in friction allows us to use more common and less complicated physics equations. The final velocity (velocity when the rolling object reaches the end of the ramp) is calculated using the following equation:

\[ V_{\text{final}} = \sqrt{\frac{10}{7} gh} \]  

(2)

Where \( g \) is once again gravity and \( h \) is the height of the ramp.

Now that the final velocity is obtained, and given that we know the initial velocity is zero we can find the average acceleration using the equation,

\[ a = \frac{V_{\text{final}}}{\Delta t} \]  

(3)

by dividing the final velocity by the change in time (time it takes to reach the bottom of ramp) we can then use this average acceleration to find the velocity at any point between the object starting down the ramp and reaching the ground. To do this we use the following equation:

\[ V_i = at \]  

(4)

This equation multiplies the acceleration down the ramp by the time \( t \) since the release of the object / when the object started rolling, allowing us to calculate the instantaneous velocity of the object.

In order to calculate instantaneous velocity for a pendulum a series of calculations were also required. We know the length of the string \( L \) used for the pendulum and the gravitational acceleration \( g \) so we were able to find the period (time to complete a swing) by using the equation:

\[ P = 2\pi \sqrt{\frac{L}{g}} \]  

(5)

When the period \( P \) is found we can then use the information we have to find the instantaneous angle of the pendulum also known at \( \theta_i \).

\[ \theta_i = \theta_{\text{highest}} \cos\left(\frac{2\pi}{P} t\right) \]  

(6)
The equation uses $\theta_{\text{highest}}$, which is the highest point of the swing (or the initial drop angle). $P$, the period of the swing, and $t$, the time since $\theta_{\text{highest}}$ in which you are trying to find the angle for. Given this angle information we can then find the instantaneous velocity of the pendulum with the equation:

$$V_B = \sqrt{2(g\cos\beta - g\cos\alpha)}$$  \(\text{(7)}\)

$V_B$ is the velocity at the point we are attempting to find. The equation uses $g$, gravitational acceleration, $L$, the length of the pendulum string, $\alpha$, the angle from vertical in which the pendulum was released, and $\beta$ the angle from vertical the pendulum is currently at in comparison to $\alpha$. To help clarify this a visual aid accompanying this set up can be seen in Figure 5.

Now that we are able to solve for the instantaneous velocity of the pendulum we apply this to each time-step in the recorded pendulum data. The accuracy of this velocity data is also significantly improved by the fact that we applied this to pendulum drops of 20° or less making it a simple small amplitude pendulum problem and improving the data generated via the prior equations. These physics based setups / environments with known equations used to calculate velocities are also accompanied by two OmniPreSense radars allow us to provide the most accurate velocities we can.

**Conclusion**

“What we see depends mainly on what we look for”, and in this I-MOVE dataset we have chosen to look for that which is crucial to many applications. We presented a novel dataset intended to help researchers progress and refine their approaches to produce more robust motion parameter estimation, specifically the velocity of the object being tracked. We identify several drawbacks, and limitations with the existing datasets in addition to explaining the differences between our dataset and ground truths. We also explain how none of the preexisting datasets contain the necessary information to adequately approach the problem of single moving object velocity estimation. We detail our meticulously crafted setup and explain how ground truth estimation from a dedicated motion parameter sensor like the Doppler radar can be verified using controlled environments and basic laws of physics. To the best of our knowledge this is the first dataset that is directly target to the problem of motion parameter estimation on independently moving objects in a complicated environment.

In future works we plan to extend the dataset to contain more diverse environments, classes of objects, as well as complicated motion paths, such as a stunts by a gymnast and skateboarder or a flying Frisbee etc. With this additional information we hope to be able to make more robust models that may be applied to a wider range of applications. As an extension, the community may explore more motion parameters such as angular velocity or rotation in degrees, but it is much harder to obtain ground truths for such setups. Another extension may also be to estimate motion parameters for each limb of a subject rather than the complete body. Though such an addition will be useful for applications oriented towards sports it may not be easy to collect.

**Acknowledgement**

The work reported in this paper is supported by the National Science Foundation under Grant No. 1659788. Any opinions, findings and conclusions or recommendations expressed in this work are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

**References**


letes using the example of swimming. In continuous human pose estimation for performance analysis of athletes using the example of swimming. In 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), 446–455. IEEE.


