

US Patent & Trademark Office

Patent Public Search | Text View

United States Patent Application Publication

20250256705

Kind Code

A1

Publication Date

August 14, 2025

Inventor(s)

Hines; Gregory Gordon Douglas et al.

SYSTEMS AND METHODS FOR DETECTING VEHICLE COLLISIONS

Abstract

Systems and methods for detecting vehicle collisions are provided. The methods involve operating at least one processor to: receive telematics data originating from a telematics device installed in a vehicle, the telematics data including acceleration data; detect a putative collision event based on the acceleration data exceeding a predetermined acceleration threshold; identify a portion of the acceleration data associated with the putative collision event, the portion of the acceleration data spanning from a time prior to the putative collision event to a time subsequent to the putative collision event; identify at least one impulse in the portion of the acceleration data based on a predetermined jerk threshold; use a trained classifier on the at least one impulse to determine that the putative collision event is a collision event; and in response to determining the collision event, trigger at least one action responsive to the collision event.

Inventors: Hines; Gregory Gordon Douglas (Mississauga, CA), Gaffney; Benjamin John Gordon (Wellesley, CA)

Applicant: Geotab Inc. (Oakville, CA)

Family ID: 94532750

Appl. No.: 19/040304

Filed: January 29, 2025

Related U.S. Application Data

us-provisional-application US 63638671 20240425

us-provisional-application US 63551581 20240209

Publication Classification

Int. Cl.: B60W30/09 (20120101); B60W30/095 (20120101); G08G1/16 (20060101)

Background/Summary

CROSS-REFERENCE TO RELATED APPLICATIONS [0001] This application claims priority to U.S. Provisional Patent Application No. 63/551,581 filed Feb. 9, 2024 and titled “SYSTEMS AND METHODS FOR DETECTING VEHICLE COLLISIONS” and U.S. Provisional Patent Application No. 63/638,671 filed Apr. 25, 2024 and titled “SYSTEMS AND METHODS FOR DETECTING VEHICLE COLLISIONS”, the contents of which are incorporated herein by reference for all purposes.

FIELD

[0002] The embodiments described herein generally relate to vehicles, telematics devices, and telematics data, and in particular, to detecting vehicle collisions using telematics data.

BACKGROUND

[0003] The following is not an admission that anything discussed below is part of the prior art or part of the common general knowledge of a person skilled in the art.

[0004] Traffic accidents, collisions, or crashes involving vehicles can have serious consequences. Crashes can result in serious injury or even death to road users, such as vehicle passengers, cyclists, and pedestrians. Collisions can also result in significant damage to personal property. Timely identification or detection of vehicle collisions can help to remedy and minimize the repercussions of collisions. However, it can be difficult to detect vehicle collisions in an accurate manner.

SUMMARY

[0005] The following introduction is provided to introduce the reader to the more detailed discussion to follow. The introduction is not intended to limit or define any claimed or as yet unclaimed invention. One or more inventions may reside in any combination or sub-combination of the elements or process steps disclosed in any part of this document including its claims and figures.

[0006] In accordance with a broad aspect, there is provided a method for detecting vehicle collisions. The method involves operating at least one processor to: receive telematics data originating from a telematics device installed in a vehicle, the telematics data including acceleration data; detect a putative collision event based on the acceleration data exceeding a predetermined acceleration threshold; identify a portion of the acceleration data associated with the putative collision event, the portion of the acceleration data spanning from a time prior to the putative collision event to a time subsequent to the putative collision event; identify at least one impulse in the portion of the acceleration data based on a predetermined jerk threshold; use a trained classifier on the at least one impulse to determine that the putative collision event is a collision event; and in response to determining the collision event, trigger at least one action responsive to the collision event.

[0007] In some embodiments, the trained classifier can be a trained k-nearest neighbor classifier.

[0008] In some embodiments, the trained classifier can be a trained decision tree classifier.

[0009] In some embodiments, the trained decision tree classifier can be a trained decision tree ensemble classifier.

[0010] In some embodiments, the trained decision tree classifier can be a trained gradient boosted decision tree classifier.

[0011] In some embodiments, the trained classifier can be used on a maximum magnitude of the at least one impulse.

[0012] In some embodiments, the trained classifier can be used on a duration of the at least one impulse.

[0013] In some embodiments, the trained classifier can be used on an area under of the curve of the at least one impulse.

[0014] In some embodiments, the trained classifier can be used on a deviation of the portion of the acceleration data associated with the putative collision event.

[0015] In some embodiments, the method can further involve operating the at least one processor to: smooth the portion of the acceleration data prior to identifying the at least one impulse.

[0016] In some embodiments, the at least one impulse can include a plurality of impulses and the trained classifier can be used on each impulse in the plurality of impulses.

[0017] In some embodiments, detecting the putative collision event can involve: determining whether the acceleration data exceeds a predetermined acceleration threshold for a duration exceeding a predetermined time threshold.

[0018] In some embodiments, identifying the at least one impulse can involve detecting a start and end to each impulse based on the predetermined jerk threshold.

[0019] In some embodiments, triggering the at least one action can involve transmitting at least one notification to at least one user associated with the vehicle.

[0020] In some embodiments, the telematics data can further include location data; and the trained classifier can be used on a portion of the location data associated with the putative collision event to determine that the putative collision event is a collision event.

[0021] In some embodiments, the location data can be GPS data.

[0022] In some embodiments, the at least one processor can be remotely located from the telematics device.

[0023] In some embodiments, the telematics device can include the at least one processor.

[0024] In some embodiments, triggering the at least one action can involve: transmitting a request to confirm whether the collision event occurred to a computing device associated with the vehicle; and receiving a response to the request confirming whether the collision event occurred.

[0025] In some embodiments, the method can further involve operating the at least one processor to: retrain the trained classifier based on the at least one impulse and the response.

[0026] In some embodiments, the trained classifier can be a first classifier; and the method can further involve operating the at least one processor to train a second classifier based on the at least one impulse and the response.

[0027] In some embodiments, the second classifier can be trained using a larger amount of training data as compared to the first classifier.

[0028] In some embodiments, the second classifier can be a neural network.

[0029] In accordance with a broad aspect, there is provided a system for detecting vehicle collisions. The system includes: at least one data store and at least one processor in communication with the at least one data store. The at least one data store is operable to store telematics data originating from a telematics device installed in a vehicle. The telematics data includes acceleration data. The at least one processor is operable to: receive the telematics data; detect a putative collision event based on the acceleration data exceeding a predetermined acceleration threshold; identify a portion of the acceleration data associated with the putative collision event, the portion of the acceleration data spanning from a time prior to the putative collision event to a time subsequent to the putative collision event; identify at least one impulse in the portion of the acceleration data based on a predetermined jerk threshold; use a trained classifier on the at least one impulse to determine that the putative collision event is a collision event; and in response to determining the collision event, trigger at least one action responsive to the collision event.

[0030] In some embodiments, the trained classifier can be a trained k-nearest neighbor classifier.

[0031] In some embodiments, the trained classifier can be a trained decision tree classifier.

[0032] In some embodiments, the trained decision tree classifier can be a trained decision tree

ensemble classifier.

[0033] In some embodiments, the trained decision tree classifier can be a trained gradient boosted decision tree classifier.

[0034] In some embodiments, the trained classifier can be used on a maximum magnitude of the at least one impulse.

[0035] In some embodiments, the trained classifier can be used on a duration of the at least one impulse.

[0036] In some embodiments, the trained classifier can be used on an area under of the curve of the at least one impulse.

[0037] In some embodiments, the trained classifier can be used on a deviation of the portion of the acceleration data associated with the putative collision event.

[0038] In some embodiments, the at least one processor can be operable to: smooth the portion of the acceleration data prior to identifying the at least one impulse.

[0039] In some embodiments, the at least one impulse can include a plurality of impulses and the trained classifier can be used on each impulse in the plurality of impulses.

[0040] In some embodiments, detecting the putative collision event can involve: determining whether the acceleration data exceeds a predetermined acceleration threshold for a duration exceeding a predetermined time threshold.

[0041] In some embodiments, identifying the at least one impulse can involve, detecting a start and end to each impulse based on the predetermined jerk threshold.

[0042] In some embodiments, triggering the at least one action can involve transmitting at least one notification to at least one user associated with the vehicle.

[0043] In some embodiments, the telematics data can further include location data; and the trained classifier can be used on a portion of the location data associated with the putative collision event to determine that the putative collision event is a collision event.

[0044] In some embodiments, the location data can be GPS data.

[0045] In some embodiments, the at least one processor can be remotely located from the telematics device.

[0046] In some embodiments, the telematics device can include the at least one processor.

[0047] In some embodiments, triggering the at least one action can involve: transmitting a request to confirm whether the collision event occurred to a computing device associated with the vehicle; and receiving a response to the request confirming whether the collision event occurred.

[0048] In some embodiments, the at least one processor can be operable to: retrain the trained classifier based on the at least one impulse and the response.

[0049] In some embodiments, the trained classifier can be a first classifier; and the at least one processor can be operable to train a second classifier based on the at least one impulse and the response.

[0050] In some embodiments, the second classifier can be trained using a larger amount of training data as compared to the first classifier.

[0051] In some embodiments, the second classifier can be a neural network.

[0052] In accordance with a broad aspect, there is provided a method for training a classifier to detect vehicle collisions. The method involves operating at least one processor to: receive telematics data originating from a plurality of telematics devices installed in a plurality of vehicles, the telematics data including acceleration data; receive label data, the label data identifying a plurality of collision events, a plurality of non-collision events, and a portion of the telematics data associated with each collision event and non-collision event; for each portion of the telematics data, identify at least one impulse in a corresponding portion of the acceleration data based on a predetermined jerk threshold; for each portion of the telematics data associated with a collision event, select a single impulse to be associated with the collision event; and train a classifier to determine whether an event is a collision event based on an impulse associated with the event using

the at least one impulse for each portion of the telematics data associated with a non-collision event and the single impulse for each portion of the telematics data associated with a collision event.

[0053] In some embodiments, the classifier can be a k-nearest neighbor classifier.

[0054] In some embodiments, the classifier can be a decision tree classifier.

[0055] In some embodiments, the classifier can be a decision tree ensemble classifier.

[0056] In some embodiments, the decision tree classifier can be a gradient boosted decision tree classifier.

[0057] In some embodiments, the classifier can be trained using a maximum magnitude of the at least one impulse for each portion of the telematics data associated with a non-collision event and a maximum magnitude of the single impulse for each portion of the telematics data associated with a collision event.

[0058] In some embodiments, the classifier can be trained using a duration of the at least one impulse for each portion of the telematics data associated with a non-collision event and a duration of the single impulse for each portion of the telematics data associated with a collision event.

[0059] In some embodiments, the classifier can be trained using an area under of the curve of the at least one impulse for each portion of the telematics data associated with a non-collision event and an area under the curve of the single impulse for each portion of the telematics data associated with a collision event.

[0060] In some embodiments, the classifier can be trained using a deviation of the acceleration data for each portion of the telematics data associated with a non-collision event and a deviation of the acceleration data for each portion of the telematics data associated with a collision event.

[0061] In some embodiments, the method can further involve operating the at least one processor to: smooth the corresponding portion of the acceleration data prior to identifying the at least one impulse.

[0062] In some embodiments, the at least one impulse can include a plurality of impulses and the classifier can be trained using the plurality of impulses for each portion of the telematics data associated with a non-collision event and the single impulse for each portion of the telematics data associated with a collision event.

[0063] In some embodiments, identifying the at least one impulse can involve, detecting a start and end to each impulse based on the predetermined jerk threshold.

[0064] In some embodiments, the telematics data can further include location data; and the classifier can be further trained using a corresponding portion of the location data.

[0065] In some embodiments, the location data can be GPS data.

[0066] In some embodiments, the at least one processor can be remotely located from the telematics device.

[0067] In some embodiments, the telematics device can include the at least one processor.

[0068] In accordance with a broad aspect, there is provided a system for training a classifier to detect vehicle collisions. The system includes: at least one data store operable to store telematics data and label data and at least one processor in communication with the at least one data store. The telematics data originates from a plurality of telematics devices installed in a plurality of vehicles. The telematics data includes acceleration data. The label data identifies a plurality of collision events, a plurality of non-collision events, and a portion of the telematics data associated with each collision event and non-collision event. The at least one processor is operable to: receive the telematics data; receive the label data; for each portion of the telematics data, identify at least one impulse in a corresponding portion of the acceleration data based on a predetermined jerk threshold; for each portion of the telematics data associated with a collision event, select a single impulse to be associated with the collision event; and train a classifier to determine whether an event is a collision event based on an impulse associated with the event using the at least one impulse for each portion of the telematics data associated with a non-collision event and the single impulse for each portion of the telematics data associated with a collision event.

[0069] In some embodiments, the classifier can be a k-nearest neighbor classifier.

[0070] In some embodiments, the classifier can be a decision tree classifier.

[0071] In some embodiments, the classifier can be a decision tree ensemble classifier.

[0072] In some embodiments, the decision tree classifier can be a gradient boosted decision tree classifier.

[0073] In some embodiments, the classifier can be trained using a maximum magnitude of the at least one impulse for each portion of the telematics data associated with a non-collision event and a maximum magnitude of the single impulse for each portion of the telematics data associated with a collision event.

[0074] In some embodiments, the classifier can be trained using a duration of the at least one impulse for each portion of the telematics data associated with a non-collision event and a duration of the single impulse for each portion of the telematics data associated with a collision event.

[0075] In some embodiments, the classifier can be trained using an area under of the curve of the at least one impulse for each portion of the telematics data associated with a non-collision event and an area under the curve of the single impulse for each portion of the telematics data associated with a collision event.

[0076] In some embodiments, the classifier can be trained using a deviation of the acceleration data for each portion of the telematics data associated with a non-collision event and a deviation of the acceleration data for each portion of the telematics data associated with a collision event.

[0077] In some embodiments, the at least one processor can be operable to: smooth the corresponding portion of the acceleration data prior to identifying the at least one impulse.

[0078] In some embodiments, the at least one impulse can include a plurality of impulses and the classifier can be trained using the plurality of impulses for each portion of the telematics data associated with a non-collision event and the single impulse for each portion of the telematics data associated with a collision event.

[0079] In some embodiments, identifying the at least one impulse can involve, detecting a start and end to each impulse based on the predetermined jerk threshold.

[0080] In some embodiments, the telematics data can further include location data; and the classifier can be further trained using a corresponding portion of the location data.

[0081] In some embodiments, the location data can be GPS data.

[0082] In some embodiments, the at least one processor can be remotely located from the telematics device.

[0083] In some embodiments, the telematics device can include the at least one processor.

[0084] In accordance with a broad aspect, there is provided a non-transitory computer readable medium having instructions stored thereon executable by at least one processor to implement any one of the methods herein.

Description

BRIEF DESCRIPTION OF THE DRAWINGS

[0085] Several embodiments will be described in detail with reference to the drawings, in which:

[0086] FIG. 1 is a block diagram of various components interacting with an example fleet management system, in accordance with an embodiment;

[0087] FIG. 2 is a block diagram of an example fleet management system interacting with an example telematics device and example vehicle, in accordance with an embodiment;

[0088] FIG. 3 is a block diagram of an example computing device interacting with an example fleet management system, in accordance with an embodiment;

[0089] FIG. 4 is a flowchart of an example method for detecting vehicle collisions, in accordance with an embodiment;

[0090] FIG. 5 is a plurality of graphs of example telematics data, in accordance with an embodiment;

[0091] FIG. 6 is a graph of example acceleration data containing an example putative collision event, in accordance with an embodiment;

[0092] FIG. 7 is a graph of example acceleration data containing example impulses, in accordance with an embodiment;

[0093] FIG. 8 is a graph of an example training dataset for a k-nearest neighbor classifier, in accordance with an embodiment;

[0094] FIG. 9A is a graph of an example impulse, in accordance with an embodiment;

[0095] FIG. 9B is a graph of an example impulse, in accordance with an embodiment;

[0096] FIG. 9C is a graph of example acceleration data, in accordance with an embodiment;

[0097] FIG. 10 is an example graphical user interface displaying a notification of a collision event, in accordance with an embodiment;

[0098] FIG. 11 is an example graphical user interface displaying a request to confirm a collision event, in accordance with an embodiment

[0099] FIG. 12 is a block diagram of an example artificial neural network, in accordance with an embodiment;

[0100] FIG. 13 is a flowchart of an example method for training a classifier to detect vehicle collisions, in accordance with an embodiment;

[0101] FIG. 14 is a block diagram of an example decision tree, in accordance with an embodiment; and

[0102] FIG. 15 is a block diagram of an example decision tree ensemble, in accordance with an embodiment.

[0103] The drawings, described below, are provided for purposes of illustration, and not of limitation, of the aspects and features of various examples of embodiments described herein. For simplicity and clarity of illustration, elements shown in the drawings have not necessarily been drawn to scale. The dimensions of some of the elements may be exaggerated relative to other elements for clarity. It will be appreciated that for simplicity and clarity of illustration, where considered appropriate, reference numerals may be repeated among the drawings to indicate corresponding or analogous elements or steps.

DETAILED DESCRIPTION

[0104] Various systems or methods will be described below to provide an example of an embodiment of the claimed subject matter. No embodiment described below limits any claimed subject matter and any claimed subject matter may cover methods or systems that differ from those described below. The claimed subject matter is not limited to systems or methods having all of the features of any one system or method described below or to features common to multiple or all of the apparatuses or methods described below. It is possible that a system or method described below is not an embodiment that is recited in any claimed subject matter. Any subject matter disclosed in a system or method described below that is not claimed in this document may be the subject matter of another protective instrument, for example, a continuing patent application, and the applicants, inventors or owners do not intend to abandon, disclaim or dedicate to the public any such subject matter by its disclosure in this document.

[0105] Referring to FIG. 1, there is shown an example asset management system **110** for managing a plurality of assets equipped with a plurality of telematics devices **130**. In operation, the telematics devices **130** can gather various data associated with the assets (i.e., telematics data) and share the telematics data with the asset management system **110**. The asset management system **110** can be remotely located from the telematics devices **130** and the assets.

[0106] For ease of exposition, various examples will now be described in which the assets are vehicles **120** and the asset management system **110** is referred to as a fleet management system **110**. However, it should be appreciated that the systems and methods described herein may be used

to manage other forms of assets in some embodiments. Such assets can generally include any apparatuses, articles, machines, and/or equipment that can be equipped and monitored by the telematics devices **130**. For example, other assets may include shipping containers, trailers, construction equipment, generators, and the like. The nature and format of the telematics data may vary depending on the type of asset.

[0107] The vehicles **120** may include any machines for transporting goods or people. The vehicles **120** can include motor vehicles, such as, but not limited to, motorcycles, cars, trucks, and/or buses. The motor vehicles can be gas, diesel, electric, hybrid, and/or alternative fuel. In some cases, the vehicles **120** may include other kinds of vehicles, such as, but not limited to, railed vehicles (e.g., trains, trams), watercraft (e.g., ships, boats), aircraft (e.g., airplanes, helicopters), and/or spacecraft. Each vehicle **120** can be equipped with a telematics device **130**. Although only three vehicles **120** having three telematics devices **130** are shown in the illustrated example for ease of illustration, it should be appreciated that there can be any number of vehicles **120** and telematics devices **130**. In some cases, the fleet management system **110** may manage hundreds, thousands, or even millions of vehicles **120** and telematics devices **130**.

[0108] The telematics devices **130** can be standalone devices that are removably installed in the vehicles **120**, such as, but not limited to, vehicle tracking devices. Alternatively, the telematics devices **130** can be integrated or embedded components that are integral with the vehicles **120**, such as, but not limited to, telematic control units (TCUs). The telematics devices **130** can gather various telematics data from the vehicles **120** and share the telematics data with the fleet management system **110**. The telematics data may include any information, parameters, attributes, characteristics, and/or features associated with the vehicles **120**. For example, the telematics data can include, but is not limited to, location data, speed data, acceleration data, engine data, brake data, transmission data, fluid data (e.g., oil, coolant, and/or washer fluid), energy data (e.g., battery and/or fuel level), odometer data, vehicle identifying data, error/diagnostic data, tire pressure data, seatbelt data, and/or airbag data. In some cases, the telematics data may include information related to the telematics devices **130** and/or other devices associated with the telematics devices **130**.

[0109] The fleet management system **110** can process the telematics data collected from the telematics devices **130** to provide various analysis, predictions, reporting, and alerts. For example, the fleet management system **110** can process the telematics data to gain additional information regarding the vehicles **120**, such as, but not limited to, trip distances/times, idling times, harsh braking/driving, usage rate, and/or fuel economy. Various data analytics and machine learning techniques may be used by the fleet management system **110** to process the telematics data. The telematics data can then be used to manage various aspects of the vehicles **120**, such as, but not limited to, route planning, vehicle maintenance, driver compliance, asset utilization, and/or fuel management. In this manner, the fleet management system **110** can improve the productivity, efficiency, safety, and/or sustainability of the vehicles **120**.

[0110] A plurality of computing devices **150** can provide access to the fleet management system **110** to a plurality of users **160**. This may allow the users **160** to manage and track the vehicles **120**, for example, using various telematics data collected and/or processed by the fleet management system **110**. The computing devices **150** can be any computers, such as, but not limited to, personal computers, portable computers, wearable computers, workstations, desktops, laptops, smartphones, tablets, smartwatches, PDAs (personal digital assistants), and/or mobile devices. The computing devices **150** can be remotely located from the fleet management system **110**, telematics devices **130**, and vehicles **120**. Although only three computing devices **150** operated by three users **160** are shown in the illustrated example for ease of illustration, it should be appreciated that there can be any number of computing devices **150** and users **160**. In some cases, the fleet management system **110** may service hundreds, thousands, or even millions of computing devices **150** and users **160**.

[0111] The fleet management system **110**, telematics devices **130**, and computing devices **150** can communicate through one or more networks **140**. The networks **140** may be wireless, wired, or a

combination thereof. The networks **140** may employ any communication protocol and utilize any communication medium. For example, the networks **140** may include, but is not limited to, Wi-Fi™ networks, Ethernet networks, Bluetooth™ networks, NFC (near-field communication) networks, radio networks, cellular networks, and/or satellite networks. The networks **140** may be private, public, or a combination thereof. For example, the networks **140** may include, but is not limited to, LANs (local area networks), WANs (wide area networks), and/or the Internet. The networks **140** may also facilitate communication with other devices and systems that are not shown.

[0112] The fleet management system **110** can be implemented using one or more computers. For example, the fleet management system **110** may be implemented using one or more computer servers. The servers can be distributed across a wide geographical area. In some embodiments, the fleet management system **110** may be implemented using a cloud computing platform, such as Google Cloud Platform™ or Amazon Web Services™. In other embodiments, the fleet management system **110** may be implemented using one or more dedicated computer servers.

[0113] Reference will now be made to FIG. **2** to further explain the operation of the fleet management system **110**, telematics devices **130**, and vehicles **120**. In the illustrated example, the fleet management system **110** is in communication with a telematics device **130** that is installed in a vehicle **120**.

[0114] As shown, the fleet management system **110** can include one or more processors **112**, one or more data stores **114**, and one or more communication interfaces **116**. Each of these components may communicate with each other. Each of these components may be combined into fewer components or divided into additional subcomponents. Two or more of these components and/or subcomponents may be distributed across a wide geographical area.

[0115] The processors **112** can control the operation of the fleet management system **110**. The processors **112** can be implemented using any suitable processing devices or systems, such as, but not limited to, CPUs (central processing units), GPUs (graphics processing units), FPGAs, (field programmable gate arrays), ASICs (application specific integrated circuits), DSPs (digital signal processors), NPU (neural processing units), QPU (quantum processing units), microprocessors, and/or controllers. The processors **112** can execute various computer instructions, programs, and/or software stored on the data stores **114** to implement various methods described herein. For example, the processors **112** may process various telematics data collected by the fleet management system **110** from the telematics device **130**.

[0116] The data stores **114** can store various data for the fleet management system **110**. The data stores **114** can be implemented using any suitable data storage devices or systems, such as, but not limited to, RAM (random access memory), ROM (read only memory), flash memory, HDD (hard disk drives), SSD (solid-state drives), magnetic tape drives, optical disc drives, and/or memory cards. The data stores **114** may include volatile memory, non-volatile memory, or a combination thereof. The data stores **114** may include non-transitory computer readable media. The data stores **114** can store various computer instructions, programs, and/or software that can be executed by the processors **112** to implement various methods described herein. The data stores **114** may store various telematics data collected from the telematics device **130** and/or processed by the processors **112**.

[0117] The communication interfaces **116** can enable communication between the fleet management system **110** and other devices or systems, such as the telematics device **130**. The communication interfaces **116** can be implemented using any suitable communication devices or systems. For example, the communication interfaces **116** may include various physical connectors, ports, or terminals, such as, but not limited to, USB (universal serial bus), Ethernet, Thunderbolt, Firewire, SATA (serial advanced technology attachment), PCI (peripheral component interconnect), HDMI (high-definition multimedia interface), and/or DisplayPort. The communication interfaces **116** can also include various wireless interface components to connect to wireless networks, such

as, but not limited to, Wi-Fi™, Bluetooth™, NFC, cellular, and/or satellite. The communication interfaces **116** can enable various inputs and outputs to be received at and sent from the fleet management system **110**. For example, the communication interfaces **116** may be used to retrieve telematics data from the telematics device **130**.

[0118] As shown, the telematics device **130** also can include one or more processors **132**, one or more data stores **134**, and one or more communication interfaces **136**. Additionally, the telematics device **130** can include one or more sensors **138**. Each of these components may communicate with each other. Each of these components may be combined into fewer components or divided into additional subcomponents.

[0119] The processors **132** can control the operation of the telematics device **130**. Like the processors **112** of the fleet management system **110**, the processors **132** of the telematics device **130** can be implemented using any suitable processing devices or systems. The processors **132** can execute various computer instructions, programs, and/or software stored on the data stores **134**. For example, the processors **132** can process various telematics data gathered from the vehicle components **122** or the sensors **138**.

[0120] The data stores **134** can store various data for the telematics device **130**. Like the data stores **114** of the fleet management system **110**, the data stores **134** of the telematics device **130** can be implemented using any suitable data storage devices or systems. The data stores **134** can store various computer instructions, programs, and/or software that can be executed by the processors **132**. The data stores **134** can also store various telematics data gathered from the vehicle components **122** or the sensors **138**.

[0121] The communication interfaces **136** can enable communication between the telematics device **130** and other devices or systems, such as the fleet management system **110** and vehicle components **122**. Like the communication interfaces **116** of the fleet management system **110**, the communication interfaces **136** of the telematics device **130** can be implemented using any suitable communication devices or systems. The communication interfaces **136** can enable various inputs and outputs to be received at and sent from the telematics device **130**. For example, the communication interfaces **136** may be used collect telematics data from the vehicle components **122** and sensors **138** or to send telematics data to the fleet management system **110**. The communication interfaces **136** can also be used to connect the telematics device **130** with one or more accessory devices **170**.

[0122] The sensors **138** can detect and/or measure various environmental events and/or changes. The sensors **138** can include any suitable sensing devices or systems, including, but not limited to, location sensors, velocity sensors, acceleration sensors, orientation sensors, vibration sensors, proximity sensors, temperature sensors, humidity sensors, pressure sensors, optical sensors, and/or audio sensors. When the telematics device **130** is installed in the vehicle **120**, the sensor **138** can be used to gather telematics data that may not be obtainable from the vehicle components **122**. For example, the sensors **138** may include a satellite navigation device, such as, but not limited to, a GPS (global positioning system) receiver, which can measure the location of the vehicle **120**. As another example, the sensor **138** may include accelerometers, gyroscopes, magnetometers, and/or IMUs (inertial measurement units), which can measure the acceleration and/or orientation of the vehicle **120**.

[0123] In some cases, the telematics device **130** may operate in conjunction with one or more accessory devices **170** that are in communication with the telematics device **130**. The accessory devices **170** can include expansion devices that can provide additional functionality to the telematics device **130**. For example, the accessory devices **170** may provide additional processing, storage, communication, and/or sensing functionality through one or more additional processors, data storages, communication interfaces, and/or sensors (not shown). The accessory devices **170** can also include adapter devices that facilitate communication between the communication interface **136** and the vehicle interfaces **124**, such as a cable harness.

[0124] The telematics device **130** can be installed within the vehicle **120**, removably or integrally. One or more accessory devices **170** can also be installed in the vehicle **120** along with the telematics device **130**. As shown, the vehicle **120** can include one or more vehicle components **122** and one or more vehicle interfaces **124**. Each of these components may be combined into fewer components or divided into additional subcomponents.

[0125] The vehicle components **122** can include any subsystems, parts, and/or subcomponents of the vehicle **120**. The vehicle components **122** can be used to operate and/or control the vehicle **120**. For example, the vehicle components **122** can include, but are not limited to, powertrains, engines, transmissions, steering, braking, seating, batteries, doors, and/or suspensions. The telematics device **130** can gather various telematics data from the vehicle components **122**. For example, the telematics device **130** may communicate with one or more ECUs (electronic control units) that control the vehicle components **122** and/or one or more internal vehicle sensors.

[0126] The vehicle interfaces **124** can facilitate communication between the vehicle components **122** and other devices or systems. The vehicle interfaces **124** can include any suitable communication devices or systems. For example, the vehicle interfaces **124** may include, but is not limited to, ODB-II (on-board diagnostics) ports and/or CAN (controller area network) buses. The vehicle interfaces **124** can be used by the telematics device **130** to gather telematics data from the vehicle components **122**. For example, the communication interfaces **136** of the telematics device **130** can be connected to the vehicle interfaces **124** to communicate with the vehicle components **122**. In some cases, an accessory device **170**, such as a wire harness, can provide the connection between the communication interface **136** and the vehicle interface **124**.

[0127] Reference will now be made to FIG. **3** to further explain the operation of the fleet management system **110** and computing devices **150**. In the illustrated example, the fleet management system **110** is in communication with a computing device **150**. As shown, the computing device **150** also can include one or more processors **152**, one or more data stores **154**, and one or more communication interfaces **156**. Additionally, the computing device **150** can include one or more displays **158**. Each of these components can communicate with each other. Each of these components may be combined into fewer components or divided into additional subcomponents.

[0128] The processors **152** can control the operation of the computing device **150**. Like the processors **112** of the fleet management system **110** and the processors **132** of the telematics device **130**, the processors **152** of the computing device **150** can be implemented using any suitable processing devices or systems. The processors **152** can execute various computer instructions, programs, and/or software stored on the data stores **154** to implement various methods described herein. For example, the processors **152** may process various telematics data received from the fleet management system **110** and/or the telematics device **130**.

[0129] The data stores **154** can store various data for the computing device **150**. Like the data stores **114** of the fleet management system **110** and the data stores **134** of the telematics device **130**, the data stores **154** of the computing device **150** can be implemented using any suitable data storage devices or systems. The data stores **154** can store various computer instructions, programs, and/or software that can be executed by the processor **152** to implement various methods described herein. The data stores **154** may store various telematics data received from the fleet management system **110** and/or the telematics device **130**.

[0130] The communication interfaces **156** can enable communication between the computing device **150** and other devices or systems, such as the fleet management system **110**. Like the communication interfaces **116** of the fleet management system **110** and the communication interfaces **136** of the telematics device **130**, the communication interfaces **156** of the computing device **150** can be implemented using any suitable communication devices or systems. The communication interfaces **156** can enable various inputs and outputs to be received at and sent from the computing device **150**. For example, the communication interfaces **116** may be used to retrieve telematics data from the fleet management system **110**.

[0131] The displays **158** can visually present various data for the computing device **150**. The displays **158** can be implemented using any suitable display devices or systems, such as, but not limited to, LED (light-emitting diode) displays, LCDs (liquid crystal displays), ELDs (electroluminescent displays), plasma displays, quantum dot displays, and/or cathode ray tube (CRT) displays. The displays **158** can be an integrated component that is integral with the computing device **150** or a standalone device that is removably connected to the computing device **150**. The displays **158** can present various user interfaces for various computer applications, programs, and/or software associated with various methods described herein. For example, the displays **158** may display various visual representations of the telematics data.

[0132] Referring now to FIG. **4**, there is shown an example method **400** for detecting vehicle collisions. The collision detection method **400** can be implemented at the fleet management system **110** (e.g., by at least one processor **112** executing instructions stored on at least one data store **114**). An advantage of implementing at least a portion of the collision detection method **400** at the fleet management system **110** (i.e., remote from telematics devices **130** and computing devices **150**) is that less processing may be executed at the telematics devices **130** and/or computing devices **150**. Hence, the hardware complexity and cost of the telematics devices **130** and/or computing devices **150** can be reduced. Furthermore, it may be easier to update and/or modify software running on the fleet management system **110** as compared to the telematics devices **130** and/or computing devices **150**. However, it should be appreciated that the collision detection method **400** may also be implemented, at least in part, using one or more telematics devices **130**, one or more computing devices **150**, or a combination thereof in some embodiments. That is, the collision detection method **400** may be implemented by any of the one or more processors **112**, **132**, **152** executing instructions stored on any of the one or more data stores **114**, **134**, **154**.

[0133] The collision detection method **400** generally involves using a trained classifier to detect collisions based on various features extracted from telematics data originating from the telematics devices **130**. The trained classifier is a machine learning or artificial intelligence model that the inventors recognized and realized could more effectively detect collisions compared to conventional approaches, which are typically heuristics-based. The inventors recognized and realized that by selecting specific features of the telematics data, such as impulses in acceleration data, particular models could be utilized to effectively detect collisions, even if the model was trained using a relatively small dataset. The inventors also recognized and realized that the collision detection method **400** could be used to collect additional training data, which could be used to retrain the classifier and/or train other classifiers, including more complex classifiers requiring larger amounts of training data. The inventors recognized and realized that this could improve the speed and accuracy of collision detection, and could therefore help to remedy and minimize the repercussions of vehicle collisions.

[0134] At **402**, telematics data can be received. The telematics data can be received from or by a telematics device **130**, fleet management system **110**, and/or computing device **150**. For example, the telematics data can be received from any of the one or more data stores **114**, **134**, **154**. The telematics data can originate from a telematics device **130** installed in a vehicle **120**, for example, being collected or generated by the telematics device **130** during operation of the vehicle **120**. As described herein, the telematics data can include various information pertaining to the operation of the vehicle **120**. For example, the telematics data may include acceleration data representing the acceleration of the vehicle **120** at various points in time during a trip performed by the vehicle **120**. The acceleration data can originate from an accelerometer of the telematics device **130** and/or the vehicle **120**. As another example, the telematics data may include location data (e.g., GPS data) representing the location of the vehicle **120** at various points in time during a trip performed by the vehicle **120**. The location data can originate from a GPS receiver of the telematics device **130** and/or the vehicle **120**.

[0135] FIG. **5** shows various examples of telematics data **500** that can be received at **402**. In the

illustrated example, the telematics data **500** includes various time series of data points corresponding to various vehicle parameters. Each data point can represent the value of a particular vehicle parameter at a given time. In the illustrated example, the telematics data includes speed data **502**, acceleration data **504**, latitude data **506**, longitude data **508**, and elevation data **510**. [0136] The speed data **502** can represent the velocity or rate of change of change of position with respect to time of the vehicle **120**. The speed data **502** can be measured by a speed sensor or speedometer of the vehicle **120** and/or telematics device **130** or derived from position data, such as the latitude data **506**, longitude data **508**, and/or elevation data **510**.

[0137] The acceleration data **504** can represent the rate of change of velocity with respect to time of the vehicle **120**. The acceleration data **504** can be measured by an acceleration sensor or accelerometer of the vehicle **120** and/or telematics device **130** or derived from position data, such as the latitude data **506**, longitude data **508**, and/or elevation data **510**, and/or the speed data **502**.

[0138] The latitude data **506**, longitude data **508**, and elevation data **510** can generally represent the location or position of the vehicle **120**. More specifically, the latitude data **506** can describe the north-south position of the vehicle **120**, the longitude data **508** can describe the east-west position of the vehicle **120**, and the elevation data **510** can describe the vertical position or height of the vehicle **120**. The latitude data **506**, longitude data **508**, and elevation data **510** can be measured by a position or location sensor of the vehicle **120** and/or telematics device **130**, such as a GPS (Global Positioning System) receiver.

[0139] It should be appreciated the telematics data **500** received at **402** is not limited to the examples shown in FIG. 5. That is, more, fewer, or different types of telematics data **500** may be received at **402**. In some embodiments, the telematics data **500** may include only a single vehicle parameter, such as acceleration data **504**. In other embodiments, the telematics data **500** may include other or additional vehicle parameters, including those not shown in FIG. 5. Generally, the telematics data **500** can vary depending on the type of trained classifiers employed by the collision detection method **400** and the telematics data **500** can include any type of data originating from the telematics device **130** that is suitable or otherwise compatible with the classifier.

[0140] At **404**, a putative collision event can be detected. The putative collision event can represent a potential, probable, or plausible, but unconfirmed, collision experienced by the vehicle **120**. The putative collision event can be detected based on acceleration data **504** exceeding a predetermined acceleration threshold. A sudden acceleration or deceleration of the vehicle **120** can indicate or suggest that the vehicle **120** has been involved in a collision. The predetermined acceleration threshold can represent an acceleration value above which a collision is presumed or suspected to have occurred.

[0141] The predetermined acceleration threshold can be determined empirically and/or heuristically and can vary, depending on the desired sensitivity of detection. A higher acceleration threshold may produce fewer false positives arising from high acceleration values that are not associated with a collision, for example, caused by potholes, speed bumps, and/or harsh driving. However, a higher acceleration threshold may also result in more false negatives arising from low impact collisions having low acceleration values. Conversely, a lower acceleration threshold may produce fewer false negatives, but may also result in more false positives. Examples of acceleration threshold values may include 0.5G, 1G, 1.5G, 2G, 2.5G, 3G, etc. where G is 9.81 m/s.², the acceleration due to Earth's gravity. Various other acceleration thresholds may be suitable.

[0142] In some embodiments, the putative collision event can be detected based on the acceleration data **504** exceeding the predetermined acceleration threshold for a duration exceeding a predetermined time threshold. In other words, the duration or length of time of the acceleration data **504** exceeding the predetermined acceleration threshold can also be evaluated when determining whether a putative collision event has occurred. Like the acceleration threshold, the predetermined time threshold can be determined empirically and/or heuristically and can vary, depending on the desired sensitivity of detection. Longer time periods may reduce false positives arising from rapid,

short-duration accelerations that are unrelated to a collision, for example, caused by potholes, speed bumps, and/or harsh driving. On the other hand, shorter time periods may reduce false positives arising from rapid, short-duration accelerations that are in fact related with a collision. Examples of time threshold durations may include 1 s, 1.5 s, 2 s, 2.5 s, 3 s, 3.5 s, etc. Various other time thresholds may be suitable.

[0143] FIG. 6 shows an example of acceleration data **504** that can be used to detect a putative collision event **600** at **404**. In the illustrated example, the acceleration data **504** exceeds a predetermined acceleration threshold **602** for a duration **614** exceeding a predetermined time threshold **612**. Accordingly, a putative collision event **600** can be detected in the illustrated example. As will be appreciated, the acceleration threshold **602** and time threshold **612** can be tuned based on the desired sensitivity of detection.

[0144] An advantage of detecting putative collision events **600** at **404**, prior to using a trained classifier, is that less data may be processed by the classifier. The inventors recognized and realized that by detecting of potential, probable, or plausible collision events **600**, the trained classifier could be used on smaller portions of acceleration data **504**. This can improve the efficiency and performance of the classifier. The inventors also recognized and realized that initial detection of putative collision events **600** could reduce the number of false positives detected by the trained classifier.

[0145] At **406**, a portion of the acceleration data **504** associated with the putative collision event **600** can be identified. The portion of the acceleration data **504** can span from a time prior to the putative collision event **600** to a time subsequent to the putative collision event **600**. That is, the portion of the acceleration data **504** can include acceleration data **504** from before, after, and during the putative collision event **600**. Put another way, the portion of the acceleration data **504** can include data outside of or in addition to the acceleration data **504** exceeding the predetermined acceleration threshold for the duration exceeding a predetermined time threshold (i.e., the acceleration data **504** used to detect the putative collision event **600**).

[0146] FIG. 6 shows an example portion **620** of the acceleration data **504** associated with the putative collision event **600**. As shown, the portion **620** of the acceleration data spans from a time **622** prior to the putative collision event **600** to a time **624** subsequent to the putative collision event **600**. Also, the portion **620** of the acceleration data **504** includes acceleration data **504** outside of or in addition to the acceleration data used to detect the putative collision event **600**.

[0147] The length of the portion **620** of the acceleration data **504** can be defined based on one or more predetermined time periods. For example, a first predetermined time period **630** can specify an amount of time prior to the putative collision event **600** and a second predetermined time period **632** can specify an amount of time after the putative collision event **600**. The predetermined time periods **630**, **632** can be same length or different lengths. The predetermined time periods **630**, **632** can be determined empirically and/or heuristically, and can vary depending on the trained classifier. Examples of predetermined time periods may include 1 s, 1.5 s, 2 s, 2.5 s, 3 s, 3.5 s, etc. Various other time periods may be suitable.

[0148] At **408**, at least one impulse can be identified in the portion **620** of the acceleration data **504**. Each impulse can be defined by a rapid, transient change in acceleration magnitude, such as a rapid increase or decrease from a baseline acceleration value, followed by a rapid return to the baseline acceleration value. The at least one impulse can be identified based on jerk, or the rate of change of acceleration with respect to time. A predetermined jerk threshold can be used to identify the at least one impulse. For example, the predetermined jerk threshold can be used detect the start and end of each impulse. The start and end of each impulse can have jerk values that exceed the predetermined jerk threshold, since the start and end of an impulse generally contain the greatest jerk values in the impulse. Since the start and end of an impulse bound and define the impulse, they can be used to identify or detect the impulse. The jerk threshold can be determined empirically and/or heuristically, and can vary depending on the desired identification sensitivity. Examples of jerk

thresholds may include 0.25 G/s, 0.5 G/s, 1 G/s, 1.25 G/s, 1.5 G/s, etc. where G is 9.81 m/s.², the acceleration due to Earth's gravity. Various other jerk thresholds may be suitable.

[0149] FIG. 7 shows a portion **620** of acceleration data **504** associated with a putative collision event **600**. As shown, the portion **620** of the acceleration data **620** can contain one or more impulses **710**. The impulses **710** can be detected based on jerk, or the rate of change of the acceleration data **504** with respect to time. More specifically, the impulses **710** can be detected by detecting the start and end **720** of each impulse **710**. In the illustrated example, the jerk at the start and end **720** of each impulse **710** are indicated. The start and end **720** of each impulse **710** can be detected by detecting that the indicated jerk values exceed a predetermined jerk threshold.

[0150] In the illustrated example, three impulses **710** are identified in the portion **620** of the acceleration data **504** associated with the putative collision event **600**. However, it should be appreciated that the portion **620** of the acceleration data **504** may contain fewer or additional impulses **710** and any number of impulses **710** may be identified. In some cases, the portion **620** of the acceleration data **504** may only contain a single impulse **710**. In other cases, the portion **620** of the acceleration data **504** may contain a plurality of impulses **710**.

[0151] In some embodiments, the portion **620** of the acceleration data **504** associated with the putative collision event **600** can be smoothed prior to detecting the at least one impulse **710**. Smoothing the acceleration data **504** prior to impulse detection can help to remove false positives arising from noisy acceleration data. However, over-smoothing may increase false positives by creating digital artifacts or increase false negatives by eliminating actual or real impulses **710**. The type and amount of smoothing can be determined empirically and/or heuristically and can vary. Various smoothing algorithms, separately or in combination, may be used, such as, but not limited to, moving averages, LOWESS (locally weighted scatterplot smoothing), Savitzky-Golay filtering, kernel smoothing, exponential smoothing, running medians, Fourier transforms, wavelet smoothing, etc.

[0152] At **410**, a trained classifier can be used on the at least one impulse **710** to determine whether the putative collision event **600** is a collision event. The trained classifier may be used on each impulse **710** in the at least one impulse or a subset of the at least one impulse **710**. In some cases, the trained classifier may be used on a single impulse **710** in the at least one impulse **710**. For example, the trained classifier may be used on the largest impulse **710**. In other cases, the trained classifier may be used on every impulse **710**.

[0153] Various types of models can be trained to classify putative collision events **600**, including, but not limited to, machine learning or artificial intelligence models. The inventors recognized and realized that machine learning or artificial intelligence models could potentially be more effective at collision detection compared to traditional heuristics-based approaches for several reasons. First, machine learning models can automatically discover complex patterns and relationships within data that may not be easily identified by human-designed heuristics. Second, machine learning models can be more easily scaled to handle larger or more complex datasets. Third, machine learning models may be better at generalizing unseen data based on underlying patterns, whereas heuristics can often only handle specific scenarios. Fourth, machine learning models can be retrained as more data becomes available, allowing them to adapt and improve over time. Finally, machine learning models may potentially reduce human bias.

[0154] The classifier may be trained using supervised, unsupervised, semi-supervised, reinforcement, or other types of learning. Various models may be used, separately or in combination, such as, but not limited to, artificial neural networks, decision trees, support-vector machines, nearest neighbors, linear regression, logistical regression, Bayesian networks, random forests, genetic algorithms, ensemble models, and the like. In general, the classifier can be any model that is trained to determine whether an impulse **710** corresponds a collision event, without being explicitly programmed to do so. It should be appreciated that, in some embodiments, the classifier can be a regressor that predicts a probability of collision. The probability of collision can

then be thresholded to determine whether the putative collision event is a collision event or a non-collision event.

[0155] In some embodiments, the trained classifier is a trained k-nearest neighbor classifier. K-nearest neighbors is a supervised machine learning algorithm that makes predictions based on proximity. A k-nearest neighbors classifier determines classification based a majority vote of the “k” closest training examples. The value of k can vary, depending on desired variance and bias, and can be determined empirically and/or heuristically. An advantage of using k-nearest neighbors is that the model can require a relatively small amount of training data, as compared to other types of machine learning or artificial intelligence models. The inventors recognized and realized that labeled data pertaining to collisions can be difficult to obtain. Collisions generally do not occur frequently, and when they do occur, drivers and fleet managers can be hesitant to report collisions. Collision data that is readily available is often unreliable or inaccurate. The inventors recognized and realized that a k-nearest neighbors classifier could accurately classify collisions using a limited training dataset.

[0156] FIG. 8 shows an example k-nearest neighbor classifier **900** that can be used at **410**. As shown, the k-nearest neighbor classifier **900** uses a labeled training dataset **902**. The labeled training dataset includes a plurality of training impulses that are labeled as non-collision events **912** and a plurality of training impulses that are labeled as collision events **914**. In operation, the k-nearest neighbors classifier **900** assigns impulses **710** to the class that is most common among its k-nearest neighbours in the labeled training dataset **902**. That is, an impulse **710** is classified as a collision event if the majority of its k-nearest neighbors are labeled as collision events **914**. Likewise, an impulse **710** is classified as a non-collision event if the majority of its k-nearest neighbors are labeled as non-collision events **912**. A putative collision event **600** can be classified as a collision event if any of the impulses **710** is classified as a collision event. A putative collision event can be classified as a non-collision event if all of the impulses **710** are classified as a non-collision event.

[0157] In the illustrated example, the k-nearest neighbor classifier **900** classifies impulses **710** based on the maximum magnitude **800** and duration **850** of the impulses **710**. The maximum magnitude **800** of an impulse **710** can define the peak or max acceleration value of the impulse **710** and represent the intensity of the impulse **710**. The duration of the impulse **850** can represent the width or time extent or duration of an impulse **710**. FIG. 9A shows an example illustration of an impulse **710** and its maximum magnitude **800** and duration **850**. The inventors recognized that the maximum magnitude **800** and duration **850** of impulses **710** were features that could be effectively used by a classifier to distinguish between collision and non-collision events. Generally, the larger and longer the impulse **710** is, the more likely a collision has occurred. Conversely, the smaller and shorter an impulse **710** is, the less likely a collision has occurred.

[0158] It should be appreciated that the trained classifier may classify putative collision events **600** based on additional or other features, or other combinations of features, in some embodiments. In general, any feature extracted from the telematics data, including other features that may be extracted from the acceleration data **504**, may be used by the trained classifier to determine whether the putative collision event **600** is a collision event. In some cases, some or all of the features may be normalized or otherwise scaled to improve the accuracy of the classifier.

[0159] In some embodiments, the trained classifier may use an area under the curve of an impulse **710** to classify the impulse **710**. In other words, the trained classifier may use an integral of the impulse **710**. The area under of the curve of the impulse **710** can represent the intensity and duration of the putative collision event **600**. FIG. 9B shows an example impulse **710** and the area under the curve **860** of the impulse **710**. The inventors recognized and realized that the area under the curve of an impulse **710** could be effectively used by a classifier to distinguish between collision and non-collision events. Generally, the larger the area of the impulse **710** is, the more likely a collision has occurred. Conversely, the smaller the area of an impulse **710** is, the less likely

a collision has occurred.

[0160] In some embodiments, the trained classifier may classify a putative collision event **600** based on a deviation of the portion **620** of acceleration data **504** associated with the putative collision event **600**. The deviation of the acceleration data **504** can represent the dispersion, variability, scatter, or spread of the acceleration data **504**. For example, the deviation may be a range, standard deviation, variance, etc. The inventors recognized and realized that the deviation of acceleration data could be effectively used by a classifier as a feature to distinguish between collision and non-collision events. Generally, the smaller the deviation is, the more likely an actual collision has occurred. Conversely, the larger the deviation is, the less likely an actual collision has occurred, and the more likely there is noise or other unwanted or erroneous data. In some cases, the deviation can include the number of times the acceleration data **504** crosses a particular axis or zero point. In some cases, the deviation can include the number of times the acceleration data **504** exceeds the predetermined acceleration threshold. FIG. **9C** shows an example portion **620** of acceleration data **504** associated with a putative collision event **600** having deviation **870**.

[0161] In some embodiments, the trained classifier may be used on a portion of location data (such as latitude **506**, longitude **508** or elevation data **510**) associated with the putative collision event **600**. The inventors recognized and realized that location data may be used as a feature by the classifier to distinguish between collision events (where a vehicle **120** is likely to come to a stop) from non-collision events (where a vehicle **120** is unlikely to come to a stop). In some embodiments, the location data used by the classifier can be an indication of whether the vehicle **120** came to a stop subsequent to the putative collision event **600**.

[0162] It should be appreciated that various vectors (e.g., velocity, acceleration, impulse, jerk, etc.) described herein may refer to a net vector or a component vector. That is, various vectors described herein may refer to an overall vector or a vector in a particular direction. For example, the acceleration of a vehicle **120** may, in some cases, refer to acceleration in a single direction, such as along an x-axis (or the direction of travel of the vehicle **120**). For instance, the area under the curve of an impulse **710** may refer to the area under the curve of an impulse **710** in the x direction. In other cases, the acceleration of a vehicle **120** can refer to acceleration in multiple directions, such as along both the x-axis and y-axis (or lateral direction of the vehicle **120**). For instance, the deviation of the acceleration can refer to the deviation the acceleration in the x and y directions. Finally, the acceleration of a vehicle **120** may also, in some cases, refer to the net or overall acceleration of the vehicle **120**. For example, the maximum magnitude **800** of an impulse **710** can refer to the net or overall acceleration in x, y, and z directions.

[0163] In some embodiments, the trained classifier is a decision tree classifier. Decision trees are a supervised machine learning algorithm that models decisions and their possible consequences as a tree-like structure of nodes. An advantage of using a decision tree classifier is that it can accurately classify collisions using a limited training dataset. As described herein, the inventors recognized and realized that labeled data pertaining to collisions can be difficult to obtain.

[0164] FIG. **14** shows an example decision tree **1400**. As shown, the decision tree **1400** can have a hierarchical or tree-like structure made up of a plurality of nodes **1410** and a plurality of branches **1420** connecting the nodes **1410**. Each internal or decision node **1414** can represent a decision or test on an attribute, each branch **1420** can represent the outcome of that decision or test, and each end or leaf node **1416** can represent a classification. As shown, each path from the root node **1412** to a leaf node **1414** can represent a classification rule. The structure of the decision tree **1400** (e.g., the attributes evaluated at each node) can be established by a suitable training using suitable training data.

[0165] In some embodiments, the trained classifier is a decision tree ensemble classifier. In other words, the trained classifier can include a plurality of decision trees **1400**. Using multiple decision trees **1400** in combination can result in better predictive performance as compared to using any constituent decision tree **1400** alone. Examples of decision tree ensemble classifiers may include

bagged (or bootstrapped) decision trees, boosted decision trees, and stacked decision trees. However, it should be appreciated that various other types of decision tree ensemble classifiers may be employed.

[0166] In some embodiments, the trained classifier is a gradient boosted decision tree classifier. A gradient boosted decision tree generally involves a sequence of trees **1400**, where each tree **1400** attempts to correct the errors made by the previous trees **1400**.

[0167] FIG. **15** shows an example gradient boosted decision tree classifier **1500**. As shown, the classifier **1500** includes a plurality of decision trees **1400**. The prediction **1514** or output of each decision tree **1400** can be combined to provide an overall prediction or output **1650**. Residuals **1512** (differences between observed and estimated values) generated by a given decision tree **1400** can be provided as additional input to the subsequent decision tree **1400**. An advantage of using a gradient boosted decision tree classifier is that it can more accurately classify collisions as compared to a single decision tree, while still using a limited training dataset. As described herein, the inventors recognized and realized that labeled data pertaining to collisions can be difficult to obtain.

[0168] At **412**, various actions can be triggered based on the classification made by the trained classifier at **410**. Generally, no action is triggered if the putative collision event **600** is classified as a non-collision event. However, if the putative collision event **600** is classified as a collision event, one or more actions can be triggered.

[0169] In some embodiments, the one or more actions can involve notifying at least one user **160** associated with the vehicle **120** that a collision was detected. Various users **160** can be notified at **412**, separately or in combination, such as, but are not limited to, a driver of the vehicle **120**, a passenger of the vehicle **120**, a fleet manager of a fleet to which the vehicle **120** belongs, an insurance entity insuring the vehicle **120**, emergency services in the geographical area the vehicle **120** is located, roadside assistance services in the geographical area the vehicle **120**, and the like. A user **160** can be notified by a computing device **150** associated with the user **160**. The user **160** may be proximate to or in possession of the computing device **150**. For example, a user **160** may be notified by a notification that is transmitted to the computing device **150**. In some cases, a user **160** may be notified by a notification or alert that is displayed on the computing device **150**. A user may also be notified by a report that is accessible to the user **160** through the computing device **150**.

[0170] FIG. **10** shows an example notification **1004** that can be generated at **412**. In the illustrated example, the notification **1004** is displayed on a graphical user interface **1002** of a fleet management application **1000** that is accessible through a computing device **150**. For example, notification **1004** and graphical user interface **1002** can be displayed on display **158**. As shown, the notification **1004** can alert a fleet manager that one of the vehicles **120** in their fleet may have been in a collision. Although the fleet management application **1000** shown in the illustrated example is a browser-based web application, it should be appreciated that the notification **1004** may be sent to and displayed on other types of applications, such as, but not limited to, mobile applications, native desktop applications, etc.

[0171] In some embodiments, the one or more actions may involve transmitting a request to confirm whether the collision event occurred. The request can be transmitted to a computing device **150** associated with a user **160** associated with the vehicle **120**. For example, the request may be transmitted to a computing device **150** associated with a driver of the vehicle **120**, a passenger of the vehicle, or a fleet manager of a fleet to which the vehicle **120** belongs. Upon reviewing the request, the user **160** can submit a response confirming whether the collision event occurred, which can be transmitted back by the computing device **150** and received. An advantage of requesting and receiving responses confirming whether the collision event occurred is that the effectiveness of the classifier can be assessed and improved upon. For example, false positives can be identified and troubleshooted.

[0172] FIG. **11** shows an example request **1104** that can be transmitted at **412**. In the illustrated

example, the request **1104** is displayed on a graphical user interface **1002** of a mobile application **1100** that is accessible through a computing device **150**. For example, request **1104** and graphical user interface **1002** can be displayed on display **158**. As shown, the request can ask the user **160** to confirm whether a collision occurred. In the illustrated example, the user **160** can select one of the buttons **1110**, **1112** to transmit a response indicating whether a collision occurred. Although a mobile application **1100** is shown in the illustrated example, it should be appreciated that the request **1104** can be sent to and displayed on various other types of applications, such as, but not limited to, browser-based web applications, native desktop applications, etc.

[0173] In some embodiments, the responses can be used as additional training data. For example, the response confirming whether the collision event occurred can be used as a label for the telematics data used to detect the collision (i.e., the at least one impulse **710**). That is, the response and the at least one impulse **710** can be used to further train or retrain the classifier. The inventors recognized and realized that, because of the shortage of labeled collision data, it would be advantageous to be able gather additional training data from other sources. By retraining or further training the classifier using the additional training data, the accuracy of the classifier can be improved.

[0174] In some embodiments, the response and the at least one impulse **710** can be used to train another classifier. The other, or second classifier can be a different type of model as compared to the original or first classifier. For example, the first classifier may be a k-nearest neighbor classifier or decision tree classifier and the second classifier may be an artificial neural network. The second classifier can be trained using a larger amount of training data as compared to the first classifier. The inventors realized and recognized it may be impractical, difficult, or even impossible, to initially gather sufficient labeled collision data to train complex models requiring large training datasets, such as artificial neural networks. However, the inventors realized and recognized that additional labeled collision data could be collected based on feedback collected from users **160** in response to collisions detected by a model trained using a smaller dataset, such as a k-nearest neighbor classifier or decision tree classifier.

[0175] FIG. **12** shows an example artificial neural network **1200** that can be trained at **412**. The artificial neural network **1200** is a machine learning or artificial intelligence model that is based on the structure and function of a human brain. As shown, the artificial neural network **1200** can include a plurality of nodes or neurons **1202**. Each node **1202** can receive input signals, process them, and produce output signals. The nodes **1202** can be interconnected through various connections **1204** and be generally organized into a plurality of layers **1210**. In the illustrated example, the layers **1210** include an input layer **1212**, which can receive initial data, hidden layers **1214**, which perform a variety of computations, and an output layer **1216**, which can produce a final output. The connections **1204** between nodes **1202** can have different weights, representing the strength of the connection **1204**. The nodes **1202** can have bias parameters that affect their outputs. The artificial neural network **1200** can be trained by adjusting the biases of the nodes **1202** and the weights of the connections **1204** based on labeled training data. The number, arrangement, and nature of the nodes **1202**, connections **1204**, and layers **1210** can vary, depending on application of the artificial neural network **1200**. However, it should be appreciated that, typically, a large amount of training data is required to train an artificial neural network **1200**.

[0176] As shown in FIG. **4**, the collision detection method **400** can optionally be repeated or executed more than once. That is, after the actions are triggered at **412**, further telematics data can be received at **402**, further putative collision events can be detected at **404**, further acceleration data can be identified at **406**, further impulses can be detected at **408**, further collision events can be determined at **410**, and further actions can be triggered at **412**. In this manner, the collision detection method **400** can detect vehicle collisions in real time (or substantially real-time) as the telematics data is collected or generated by the telematics devices **130** during operation of the vehicles **120** and received.

[0177] Referring to FIG. 13, there is shown an example method **1300** for training a classifier to detect vehicle collisions. The collision classifier training method **1300** can be implemented at the fleet management system **110** (e.g., by at least one processor **112** executing instructions stored on at least one data store **114**). An advantage of implementing at least a portion of the collision classifier training method **1300** at the fleet management system **110** (i.e., remote from telematics devices **130** and computing devices **150**) is that less processing may be executed at the telematics devices **130** and/or computing devices **150**. Hence, the hardware complexity and cost of the telematics devices **130** and/or computing devices **150** can be reduced. Furthermore, it may be easier to update and/or modify software running on the fleet management system **110** as compared to the telematics devices **130** and/or computing devices **150**. However, it should be appreciated that the collision classifier training method **1300** may also be implemented, at least in part, using one or more telematics devices **130**, one or more computing devices **150**, or a combination thereof in some embodiments. That is, the collision classifier training method **1300** may be implemented by any of the one or more processors **112**, **132**, **152** executing instructions stored on any of the one or more data stores **114**, **134**, **154**.

[0178] The collision classifier training method **1300** generally involves training a classifier to detect vehicle collisions. The classifier is a machine learning or artificial intelligence model that the inventors recognized and realized could more effectively detect collisions compared to conventional approaches, which are typically heuristics-based. The inventors recognized and realized that by selecting specific features of the telematics data, such as impulses **710** in acceleration data **504**, particular models could be trained to effectively detect collisions, even if the model was trained using a relatively small dataset. The inventors recognized that models produced by training method **1300** could improve the speed and accuracy of collision detection, and could therefore help to remedy and minimize the repercussions of vehicle collisions. For example, the classifier training method **1300** can be used to generate trained classifiers that are used by the collision detection method **400** described herein.

[0179] At **1302**, telematics data can be received. The telematics data can be received from a telematics device **130**, fleet management system **110**, and/or computing device **150**. For example, the telematics data can be received from any of the one or more data stores **114**, **134**, **154**. The telematics data can originate from a plurality of telematics devices **130** installed in a plurality of vehicles **120**, for example, being collected or generated by the telematics devices **130** during operation of the vehicles **120**. As described herein, the telematics data can include various information pertaining to the operation of the vehicles **120**. The telematics data **500** can include various time series of data points corresponding to various vehicle parameters. Each data point can represent the value of a particular vehicle parameter at a given time. The telematics data can include the various examples of telematics data **500** previously described and shown in FIG. 5, including, but not limited to, speed data **502**, acceleration data **504**, latitude data **506**, longitude data **508**, and elevation data **510**. It should be appreciated that the telematics data **500** can vary depending on the type of model to be trained by the classifier training method **1300** and the telematics data can include any type of data originating from the telematics device **130** that is suitable for or otherwise compatible with training the model.

[0180] At **1304**, label data can be received. For example, label data can be received from any of the one or more data stores **114**, **134**, **154**. The label data can identify a plurality of collision events, a plurality of non-collision events, and a portion of the telematics data associated with each collision event and non-collision event. Put another way, the label data can identify a plurality of portions of the telematics data and indicate whether each portion of the telematics data is associated with a collision event or a non-collision event. Each portion of the telematics data can include telematics data from before, after, and during the respective collision or non-collision event. Put another way, each portion of the telematics data can span from a time prior to the respective collision or non-collision event to a time subsequent to the respective collision or non-collision event. The portions

of the telematics data can resemble or correspond to those identified in the collision detection method **400** at **406**, as previously described and shown in FIG. **6**.

[0181] At **1306**, impulses **710** can be detected. For each portion of the telematics data identified by the label data, at least one impulse **710** can be detected in a corresponding portion of acceleration data **504**. That is, at least one impulse **710** can be detected for each collision event and non-collision event. As described herein, each impulse **710** can be defined by a rapid, transient change in acceleration magnitude, such as a rapid increase or decrease from a baseline acceleration value, followed by a rapid return to the baseline acceleration value. The at least one impulse **710** can be identified based on jerk, or the rate of change of acceleration with respect to time, as previously described and shown in FIG. **7**, for example. A predetermined jerk threshold can be used to identify the at least one impulse **710**. For example, a predetermined jerk threshold can be used detect the start and end of each impulse **710** by detecting jerk values exceeding a predetermined jerk threshold. In some embodiments, the corresponding portion of the acceleration data **504** can be smoothed prior to detecting the at least one impulse.

[0182] At **1308**, for each portion of the telematics data associated with a collision event, a single impulse **710** can be selected to be associated with the collision event. As will be appreciated, each portion of the acceleration data **504** can contain more than one impulse **710**. However, not every impulse **710** may be directly associated with the collision event. These impulses **710** may represent other events that occurred before or after the collision event, such as a secondary collision, for example, caused by the impact of the primary collision event. The single impulse **710** can represent the impulse **710** that is most likely to be directly associated with collision event. Various criteria, alone or in combination, may be used to select the single impulse **710**. For example, the maximum acceleration magnitude **800** of the impulses **710** may be used to select the single impulses **710**. Likewise, the duration **850** of the impulses **710** may be used to select the single impulses. In some embodiments, the impulse having the largest maximum magnitude **800** and duration **850** can be selected as the single impulse. It should be appreciated that additional, fewer, or other criterion, alone or in combination, may be used to select the single impulse **710**.

[0183] At **1310**, a classifier can be trained to determine whether an event is a collision event based on an impulse **710** associated with the event. Various types of classifiers may be trained, including machine learning or artificial intelligence models. Generally, the model can be any type of model that is trained to determine whether an impulse **710** corresponds a collision event, without being explicitly programmed to do so. As described herein, the inventors recognized and realized that machine learning or artificial intelligence models could potentially be more effective at collision detection as compared to traditional heuristics-based approaches. The nature of the training can depend on the type of model being trained. For example, supervised, unsupervised, semi-supervised, reinforcement, or other types of learning may be employed.

[0184] Various features may be used to train the model. In some embodiments, the model can be trained based on impulses **710**. For example, the model can be trained using the at least one impulse **710** for each portion of the telematics data associated with a non-collision event and the single impulse **710** for each portion of the telematics data associated with a collision event. As described herein, the inventors recognized and realized it can be difficult to obtain labeled data pertaining to collisions. Accordingly, the inventors recognized and realized that a larger training dataset could be achieved by using all of the impulses **710** associated with non-collision events, while only using a single impulse **710** associated with collision events. In some cases, the model may be trained using a plurality of impulses **710** for each portion of the telematics data associated with a non-collision event. However, in some embodiments, the model may be trained using only a subset of the at least one impulse **710**. For example, the classifier may trained using on a single impulse **710** in the at least one impulse **710**. For instance, the classifier may be trained using the largest impulse **710**.

[0185] Various properties of the impulses **710** may be used to train the model. For example, the

model can be trained used on the maximum magnitude **800** and duration **850** of the at least one impulse **710** for each portion of the telematics data associated with a non-collision event and the single impulse **710** for each portion of the telematics data associated with a collision event. As described herein, the inventors recognized that the maximum magnitude **800** and duration **850** of impulses were features that could be effectively used by a model to distinguish between collision events (e.g., having larger impulses of longer duration) and non-collision events (e.g., having smaller impulses of shorter duration).

[0186] In some embodiments, additional or other features, or combinations of features, may be used to train the model. For example, the model may be trained using an area under of the curve of the at least one impulse **710** for each portion of the telematics data associated with a non-collision event and an area under the curve of the single impulse **710** for each portion of the telematics data associated with a collision event. As described herein, the inventors recognized and realized that the area under the curve of an impulse **710** could be effectively used by a model to distinguish between collision events (having larger areas) and non-collision events (having smaller areas).

[0187] The inventors recognized and realized that the deviation of the acceleration data **504** could be effectively used by a model to distinguish between collision events (having less deviation) and non-collision events (having more deviation when there is noise or other unwanted data). As well, the model may be trained using corresponding portions of the location data. The inventors recognized and realized that location data may be used as a feature by the classifier to distinguish between collision events (where a vehicle **120** is likely to comes to a stop) from non-collision events (where a vehicle **120** is unlikely to come to a stop). In general, any feature extracted from the telematics data, including other features that may be extracted from the acceleration data **504**, may be used to train the model. In some cases, some or all of the features may be normalized or otherwise scaled to improve the accuracy of the model.

[0188] In various embodiments, the classifier can be a k-nearest neighbor classifier. The inventors recognized and realized that a k-nearest neighbors classifier could accurately classify collisions using a limited training dataset. The k-nearest neighbor classifier can be trained using a supervised learning approach. More specifically, the k-nearest neighbor classifier can be trained by storing a labeled dataset. The labeled dataset includes a plurality of training examples. Each training example includes one or more features and a class label. An example of a k-nearest neighbor classifier **800** is shown in FIG. **8**. In the illustrated example, the training examples are impulses **710** and the features of the training examples are the maximum magnitude **800** and duration **850** of the impulses **710**. As shown, each training example is labeled with a class—a non-collision event **912** or a collision event **914**.

[0189] In other embodiments, the classifier may be another type of machine learning or artificial intelligence model. As described herein, the classifier can generally be any type of model that is trained to determine whether an event is a collision event, without being explicitly programmed to do so. For example, the classifier may be an artificial neural network **1200**, as described herein and shown in FIG. **12**.

[0190] In some embodiments, the classifier may be a decision tree classifier. Where the machine learning classifier is a decision tree, the training may involve using various entropy, information gain, and/or Gini impurity metrics, to determine the appropriate nodes and branching for the decision tree. For example, ID3 (Iterative Dichotomiser 3), C4.5, CART (Classification And Regression Tree), or similar algorithms may be utilized.

[0191] In some embodiments, the classifier may be a decision tree ensemble classifier. The training for a decision tree ensemble can depend on the type of ensemble. For example, for a gradient boosted decision tree, the training can involve building trees one at a time, where each new tree is created to account for the residuals or errors of the previous trees. Various algorithms can be used to train a gradient boosted decision tree, including but not limited to XGBoost (extreme Gradient Boosting), LightGBM, CatBoost, AdaBoost (Adaptive Boosting), and the like. As another example,

for a bootstrapped or bagged decision tree, the training can involve creating random bootstraps or subsets of the training data and constructing decision trees using the bootstraps.

[0192] As shown in FIG. 13, the training method 1300 can optionally be repeated or executed more than once. That is, after the classifier is trained at 1310, further telematics data can be received at 1302, further label data can be received at 1304, further impulses can be identified at 1306, further single impulses for collision events can be selected at 1308, and classifiers can be further trained at 1310. In this manner, the training method 1300 can train new classifiers or retrain existing classifiers in real-time (or substantially real-time) as the telematics data is collected or generated by the telematics devices 130 during operation of the vehicles 120 and received with label data.

[0193] It will be appreciated that for simplicity and clarity of illustration, where considered appropriate, reference numerals may be repeated among the figures to indicate corresponding or analogous elements. In addition, numerous specific details are set forth in order to provide a thorough understanding of the embodiments described herein. However, it will be understood by those of ordinary skill in the art that the embodiments described herein may be practiced without these specific details. In other instances, well-known methods, procedures and components have not been described in detail so as not to obscure the embodiments described herein. Also, the description is not to be considered as limiting the scope of the embodiments described herein.

[0194] It should also be noted that the terms “coupled” or “coupling” as used herein can have several different meanings depending in the context in which these terms are used. For example, the terms coupled or coupling may be used to indicate that an element or device can electrically, optically, or wirelessly send data to another element or device as well as receive data from another element or device. Furthermore, the term “coupled” may indicate that two elements can be directly coupled to one another or coupled to one another through one or more intermediate elements.

[0195] It should be noted that terms of degree such as “substantially”, “about” and “approximately” as used herein mean a reasonable amount of deviation of the modified term such that the end result is not significantly changed. These terms of degree may also be construed as including a deviation of the modified term if this deviation would not negate the meaning of the term it modifies.

[0196] In addition, as used herein, the wording “and/or” is intended to represent an inclusive-or. That is, “X and/or Y” is intended to mean X or Y or both, for example. As a further example, “X, Y, and/or Z” is intended to mean X or Y or Z or any combination thereof.

[0197] Furthermore, any recitation of numerical ranges by endpoints herein includes all numbers and fractions subsumed within that range (e.g., 1 to 5 includes 1, 1.5, 2, 2.75, 3, 3.90, 4, and 5). It is also to be understood that all numbers and fractions thereof are presumed to be modified by the term “about” which means a variation of up to a certain amount of the number to which reference is being made if the end result is not significantly changed.

[0198] The terms “an embodiment,” “embodiment,” “embodiments,” “the embodiment,” “the embodiments,” “one or more embodiments,” “some embodiments,” and “one embodiment” mean “one or more (but not all) embodiments of the present invention(s),” unless expressly specified otherwise.

[0199] The terms “including,” “comprising” and variations thereof mean “including but not limited to,” unless expressly specified otherwise. A listing of items does not imply that any or all of the items are mutually exclusive, unless expressly specified otherwise. The terms “a,” “an” and “the” mean “one or more,” unless expressly specified otherwise.

[0200] The example embodiments of the systems and methods described herein may be implemented as a combination of hardware or software. In some cases, the example embodiments described herein may be implemented, at least in part, by using one or more computer programs, executing on one or more programmable devices comprising at least one processing element, and a data storage element (including volatile memory, non-volatile memory, storage elements, or any combination thereof). Programmable hardware such as FPGA can also be used as standalone or in combination with other devices. These devices may also have at least one input device (e.g., a

pushbutton keyboard, mouse, a touchscreen, and the like), and at least one output device (e.g., a display screen, a printer, a wireless radio, and the like) depending on the nature of the device. The devices may also have at least one communication device (e.g., a network interface).

[0201] It should also be noted that there may be some elements that are used to implement at least part of one of the embodiments described herein that may be implemented via software that is written in a high-level computer programming language such as object-oriented programming. Accordingly, the program code may be written in C, C++ or any other suitable programming language and may comprise modules or classes, as is known to those skilled in object-oriented programming. Alternatively, or in addition thereto, some of these elements implemented via software may be written in assembly language, machine language or firmware as needed. In either case, the language may be a compiled or interpreted language.

[0202] At least some of these software programs may be stored on a storage media (e.g., a computer readable medium such as, but not limited to, ROM, magnetic disk, optical disc) or a device that is readable by a general or special purpose programmable device. The software program code, when read by the programmable device, configures the programmable device to operate in a new, specific and predefined manner in order to perform at least one of the methods described herein.

[0203] Furthermore, at least some of the programs associated with the systems and methods of the embodiments described herein may be capable of being distributed in a computer program product comprising a computer readable medium that bears computer usable instructions for one or more processors. The medium may be provided in various forms, including non-transitory forms such as, but not limited to, one or more diskettes, compact disks, tapes, chips, and magnetic and electronic storage.

[0204] The present invention has been described here by way of example only, while numerous specific details are set forth herein in order to provide a thorough understanding of the exemplary embodiments described herein. However, it will be understood by those of ordinary skill in the art that these embodiments may, in some cases, be practiced without these specific details. In other instances, well-known methods, procedures and components have not been described in detail so as not to obscure the description of the embodiments. Various modification and variations may be made to these exemplary embodiments without departing from the spirit and scope of the invention, which is limited only by the appended claims.

Claims

1. A method for detecting vehicle collisions, the method comprising operating at least one processor to: receive telematics data originating from a telematics device installed in a vehicle, the telematics data comprising acceleration data; detect a putative collision event based on the acceleration data exceeding a predetermined acceleration threshold; identify a portion of the acceleration data associated with the putative collision event, the portion of the acceleration data spanning from a time prior to the putative collision event to a time subsequent to the putative collision event; identify at least one impulse in the portion of the acceleration data based on a predetermined jerk threshold; use a trained classifier on the at least one impulse to determine that the putative collision event is a collision event; and in response to determining the collision event, trigger at least one action responsive to the collision event.
2. The method of claim 1, wherein the trained decision tree classifier is a trained gradient boosted decision tree classifier.
3. The method of claim 1, wherein the trained classifier is used on a maximum magnitude of the at least one impulse.
4. The method of claim 1, wherein the trained classifier is used on a duration of the at least one impulse.

5. The method of claim 1, wherein the trained classifier is used on an area under of the curve of the at least one impulse.
6. The method of claim 1, wherein the trained classifier is used on a deviation of the portion of the acceleration data associated with the putative collision event.
7. The method of claim 1, wherein detecting the putative collision event comprises: determining whether the acceleration data exceeds a predetermined acceleration threshold for a duration exceeding a predetermined time threshold.
8. The method of claim 1, wherein triggering the at least one action comprises: transmitting a request to confirm whether the collision event occurred to a computing device associated with the vehicle; and receiving a response to the request confirming whether the collision event occurred.
9. The method of claim 8, further comprising operating the at least one processor to: retrain the trained classifier based on the at least one impulse and the response.
10. The method of claim 8, wherein: the trained classifier is a first classifier; and the method further comprises operating the at least one processor to train a second classifier based on the at least one impulse and the response.
11. A non-transitory computer readable medium having instructions stored thereon executable by at least one processor to implement a method for detecting vehicle collisions, the method comprising operating the at least one processor to: receive telematics data originating from a telematics device installed in a vehicle, the telematics data comprising acceleration data; detect a putative collision event based on the acceleration data exceeding a predetermined acceleration threshold; identify a portion of the acceleration data associated with the putative collision event, the portion of the acceleration data spanning from a time prior to the putative collision event to a time subsequent to the putative collision event; identify at least one impulse in the portion of the acceleration data based on a predetermined jerk threshold; use a trained classifier on the at least one impulse to determine that the putative collision event is a collision event; and in response to determining the collision event, trigger at least one action responsive to the collision event.
12. A system for detecting vehicle collisions, the system comprising: at least one data store operable to store telematics data originating from a telematics device installed in a vehicle, the telematics data comprising acceleration data; at least one processor in communication with the at least one data store, the at least one processor operable to: receive the telematics data; detect a putative collision event based on the acceleration data exceeding a predetermined acceleration threshold; identify a portion of the acceleration data associated with the putative collision event, the portion of the acceleration data spanning from a time prior to the putative collision event to a time subsequent to the putative collision event; identify at least one impulse in the portion of the acceleration data based on a predetermined jerk threshold; use a trained classifier on the at least one impulse to determine that the putative collision event is a collision event; and in response to determining the collision event, trigger at least one action responsive to the collision event.
13. The system of claim 12, wherein the trained decision tree classifier is a trained gradient boosted decision tree classifier.
14. The system of claim 12, wherein the trained classifier is used on a maximum magnitude of at least one impulse.
15. The system of claim 12, wherein the trained classifier is used on a duration of the at least one impulse.
16. The system of claim 12, wherein the trained classifier is used on an area under of the curve of the at least one impulse.
17. The system of claim 12, wherein the trained classifier is used on a deviation of the portion of the acceleration data associated with the putative collision event.
18. The system of claim 12, wherein detecting the putative collision event comprises: determining whether the acceleration data exceeds a predetermined acceleration threshold for a duration exceeding a predetermined time threshold.

- 19.** The system of claim 12, wherein triggering the at least one action comprises: transmitting a request to confirm whether the collision event occurred to a computing device associated with the vehicle; and receiving a response to the request confirming whether the collision event occurred.
- 20.** The system of claim 19, wherein the at least one processor is operable to: retrain the trained classifier based on the at least one impulse and the response.
- 21.** The system of claim 19, wherein: the trained classifier is a first classifier; and the at least one processor is operable to train a second classifier based on the at least one impulse and the response.
-