CONTEXTUALLY AWARE MONITORING OF ASSETS

RELATED APPLICATIONS

[0001] This application claims the benefit of priority from U.S. Provisional Application No. 61/234,600, filed August 17, 2009. This application also claims the benefit of priority from U.S. Provisional Application No. 61/291,232, filed December 30, 2009. Each of these provisional applications is incorporated by reference herein in its entirety.

TECHNICAL FIELD

[0002] This subject matter is related generally to providing in-transit visibility of shipments in real-time.

BACKGROUND

[0003] A wireless tracking device or "tag" can use various technologies such as Global Positioning System (GPS), Radio Frequency Identification (RFID), and General Packet Radio Service (GPRS), to track and report movements of an asset (e.g., a shipping container) on which the device is mounted. Conventional wireless monitoring devices report locations to a tracking service on a set schedule, regardless of whether the location data is needed by users of the tracking service.

SUMMARY

[0004] An apparatus, method and system for contextually aware monitoring of an asset's journey through a supply chain are disclosed. In some implementations, contextually aware monitoring can include monitoring of the supply chain tradelane with tracking devices including one or more sensors for determining location, velocity, heading, vibration, acceleration (e.g., 3D acceleration), or any other sensor that can monitor the environment of the shipping container and provide contextual awareness. The contextual awareness can be enabled by geofencing and recursive algorithms, which allow dynamic modification of the tracking device behavior. Dynamic modification can reduce performance to save power (e.g., save battery usage) and lower costs. Dynamic modification can increase performance where it matters in the supply chain for improved reporting accuracy or frequency or recognition of supply chain events. Dynamic modification can adapt performance such as wireless communications to the region or location of the tracking device. Intelligent connection to wireless carriers can be performed, where the device can determine when it should report

particular events, based in part on the type of the event and the cost and resources required for reporting.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] FIG. 1 illustrates an example buyer, seller, and tracking device provider interacting in a shipping scenario.

[0006] FIG. 2 is a block diagram of an example tracking device system that associates a tracking device with an asset and monitors and tracks the asset using data received from the tracking device in accordance with at least one implementation.

[0007] FIG. 3 is a block diagram of a contextually aware tracking device coupled to an asset in accordance with at least one implementation.

[0008] FIG. 4 is a flow diagram of the Hidden Markov Model (HMM) method of determining supply chain context to dynamically adjust the operation of the device of FIG. 3 in accordance with at least one implementation.

[0009] FIG. 5 is a diagram illustrating the technical details of the HMM.

[0010] FIG. 6 is a diagram showing the sequence of hidden states and their corresponding observable states.

[0011] FIG. 7 illustrates example pseudo-code for implementing a Forward-Backward method.

[0012] FIG. 8 illustrates a Viterbi method using a recursive approach.

[0013] FIG. 9 is a simple illustration of the state and state transitioning of the HMM where the initial state is known in accordance with at least one implementation.

[0014] FIG. 10 is a block diagram of a hybrid model structure in accordance with at least one implementation.

[0015] FIG. 11 is an example of Viterbi method pseudocode.

[0016] FIG. 12 is an example of Forward-Backward method pseudocode for the hybrid model of FIG. 9.

[0017] FIGS. 13A-13C is a block diagram of multiple cases of behavior that could be dynamically adjusted by a processor method updating the operations of the tracking device of FIG. 3.

[0018] FIG. 14 is a flow diagram of a system of storing and retrieving contextual data for the tracking device from FIG. 3 in accordance with at least one implementation.

[0019] FIG. 15 illustrates nested geofencing around a shipping port.

[0020] FIG. 16 is a flow diagram of the Nested Geofence method of determining supply chain context to dynamically adjust the operation of the device of FIG. 3 in accordance with at least one implementation .

[0021] FIG. 17 are multiple flow diagrams of the implementation of the Nested Geofence method of FIG. 16 to learn new context.

[0022] FIG. 18 is a flow diagram of the Sensor Pattern Matching method of determining supply chain context to dynamically adjust the operation of the device of FIG. 3 in accordance with at least one implementation.

[0023] FIG. 19 is a detailed flow diagram of the implementation of the Sensor Pattern Matching method of FIG. 18 to learn new context.

[0024] FIG. 20 is a flow diagram showing the combined logic determining when the tracking device is to use the Nested Geofence method of FIG. 16, the HMM method of FIG. 4, or the Sensor Pattern Matching method of FIG. 18.

[0025] FIG. 21 illustrates the integrated system of using the tracking device coupled to an asset using the combined logic of FIG. 20 and reporting tracking and sensor data or supply chain alerts to a central repository.

DETAILED DESCRIPTION

Overview of Asset Tracking

[0026] FIG. 1 illustrates an example buyer 102, seller 104, and tracking device provider 106 interacting in a shipping scenario. An example asset 108 is shipped from the seller 104 to the buyer 102 on an example conveyance 110. In some implementations, the asset is an intermodal shipping container, however the asset can also be, for example, equipment, or other items capable of being monitored or tracked. Examples of conveyances include, but are not limited to, trucks, trains, ships, and airplanes. Examples of assets include, but are not limited to, containers such as dry-van containers, refrigerated containers, ISO tanks, trailers, box trucks, and unit load devices (ULDs).

[0027] In general, either the buyer 102 or the seller 104 sends a request to the tracking device provider 106 requesting tracking of the shipment of the asset 108. The tracking device provider 106 arranges for a selected tracking device 114 to be sent from tracking device pool 112 to the location from where the asset is being shipped (e.g., a warehouse of the seller 104). The tracking device pool 112 is a collection of available tracking devices. Each tracking device in the tracking device pool 112 is a tracking device that can be used to track an asset. At the location where the tracking device is shipped (the "origin location") the tracking device 114 can be affixed or coupled to the asset 108, thus securely sealing the asset 108. An

example tracking device is the Savi Networks SN-LSE-01, which is a GPS-based Location + Security + Environmental tracking device. The tracking devices do not have to use GPS, but can alternatively or additionally receive location information using various location technologies including, but not limited to: additional Global Satellite Navigation Systems (GNSS), location from cellular towers, or location from other wireless networks such as WiFi networks.

[0028] The selected tracking device 114 can be coupled to the asset 108 before the asset begins its journey and/or re-coupled to the asset 108 during the journey (e.g., after authorized custom inspections). During the journey, the tracking device 114 can be programmed to wake up periodically, initiate communication with the tracking device provider 106, and send event notifications to the tracking device provider 106. In general, each event notification can include an identification of the event (or event type), a location of the asset 108 when the event occurred, and additional details of the event such as a date and/or time when the event occurred, the status of the asset 108 before, during, or after the event, or details on the movement of the asset (e.g., accelerometer or velocimeter readings from the tracking device coupled to the asset). The event information can be stored by the tracking device provider 106, for example, in an event database. The tracking device 114 reports various events, including for example, security events, environmental events, process events, and tracking events. Security events can indicate that the asset 108 or tracking device 114 may have been tampered with. For example, the tracking device 114 can report when a vertical or horizontal bolt securing the tracking device to a container is cut (indicating that the asset was opened). Other types of tampers can also be detected (e.g., shock intrusion or light inside the asset that exceeds a threshold). Environmental events can indicate that one or more environmental variables (e.g., temperature, humidity, shock, acceleration) are beyond an acceptable range (e.g., a range specified by the user). Process events indicate that various procedural events in the journey of the asset have occurred. For example, process events can indicate that a tracking device 114 has been attached to the asset 108 or detached from the asset 108 (e.g., that the asset 108 is beginning or ending its tracking device journey). Process events can also indicate other shipment events in the journey of the asset 108 (e.g., procedural events in the journey of the asset 108), including, but not limited to, that the asset 108 has been stuffed (e.g., filled with contents), that the asset 108 has been sealed, that the asset 108 has been flagged for customs inspection, that customs inspection of the asset 108 has begun, that customs inspection of the asset 108 has ended, that the asset 108 is in a shipping yard, that the asset 108 has left a shipping yard, that the asset 108 has sailed, that the asset 108 has been

berthed, and that the asset 108 has been unsealed. Tracking events are periodic reports of the location of tracking device 114. For example, the tracking device 114 can send a report of its current location according to a schedule, for example, at fixed intervals of time, regardless of whether any other events have been issued. A tracking system (e.g., system 200 of FIG. 2) can process the tracking events to determine when an asset has entered or left a predefined area. For example, the system 200 can define geofences (e.g., a virtual perimeter) around important locations along the journey of the asset 108 (e.g., ports) and the tracking device 114 or the system 200 can determine that the asset has entered or left a given location when the tracking device 114 enters or leaves a geofence.

[0029] In some implementations, the tracking device provider 106 processes the various event notifications received from the tracking device 114 and provides notifications to the buyer 102 and/or the seller 104 and/or other parties. The notifications can be based, in part, on additional information received from the buyer 102 and/or the seller 104, for example, a description of the business of the buyer 102 and/or seller 104, a description of the contents of the asset 108, or a description of a transaction relevant to the contents of the asset 108.

[0030] In some implementations, the tracking device also processes commands (e.g., Over-the-Air (OTA) commands) received from the tracking device provider 106 during a communication session between the tracking device and servers operated by the tracking device provider 106.

Example Tracking Device System

[0031] FIG. 2 is a block diagram of an example tracking device system 200 that associates a tracking device with an asset and monitors and tracks the asset using data received from the tracking device. The system 200 commissions (associates) tracking devices to assets, decommissions (disassociates) tracking devices from assets, provides notifications of events (e.g., security, environmental, process, and tracking events), and can reset tracking device status remotely.

[0032] In some implementations, the system 200 can include one or more Zero Client Commissioning (ZCC) input devices 202, an information service 204, one or more end user systems 206, Tracking device Logistics Personnel (TL Personnel) 208, one or more assets 210, one or more tracking devices 211 affixed or coupled to the one or more assets 210, an event server 212, an event database 213, a Tracking device Pool Management System (TPMS) 214, a tracking device database 216, a message server 218, a transaction (TXN) server 224, and a failed transaction database 226.

[0033] The ZCC input devices 202 are used to commission and decommission tracking devices to assets. The ZCC input devices 202 can be any suitable communication device, including, but not limited to, mobile phones, land phones, email devices, and portable computers. The ZCC input devices 202 communicate with the information service 204 through the message server 218 using a variety of communication modes, including but not limited to: Integrated Voice Response (IVR), Short Message Service (SMS), email, hand-held application, Web interface, and Electronic Data Interchange (EDI) or any other form of electronic message sharing. The ZCC input devices 202 can be operated by various actors having various roles in the supply chain, including but not limited to: dock workers, longshoreman, logistics service providers, freight forwarders, field agents, customs agents, and any other personnel involved in the tracking of an asset.

[0034] The information service 204 allows end user systems 206 to track the status of assets 210 in real-time, integrates enterprise data for end user systems, and performs supply chain analysis, including generating supply chain management statistics. The transaction server 224 runs a tracking application that receives event location/status transaction messages (e.g., event notifications) or reports from the event server 212 and applies business logic 222 to the transactions for validating and maintaining associations between tracking device identifiers and asset identifiers. Successful transactions are posted against assets and tracking devices. Failed transactions and reason codes are written to an exception queue in the failed transaction database 226.

[0035] The information service 204 can use a portal (not shown) to provide Web forms to end user systems 206 (e.g., a browser on a PC or mobile device). The Web forms can provide an input mechanism for a user to commission or decommission tracking devices and can provide an output mechanism for users to receive real-time tracking and status information regarding assets and events.

[0036] The tracking device 211 wakes up periodically to initiate communication with the event server 212 and to send event notifications to the event server 212. In general, each event notification includes an identification of the event (or event type), a location of the asset when the event occurred, and optionally additional details of the event such as the status of the asset before, during, or after the event. The event notification can also include an identification of the tracking device, or an identification of the asset to which the tracking device is coupled. The event information can be stored in the event database 213. The tracking device 211 reports various events, including for example, security events,

environmental events, process events, tracking events, and location events, as described above with reference to FIG. 1.

[0037] The event server 212 periodically receives event notifications from the tracking device 211. The event server can process location information in the notifications. The event server 212 also constructs and sends commands (e.g., OTA commands) to the tracking device 211. Some notification management functions performed by the event server 212 include but are not limited to: checking incoming notifications for syntax errors and population of mandatory fields, checking the accuracy of location information in incoming notifications, sorting or sequencing notifications logically before forwarding the notifications to the information service 204, and constructing output transactions that comply with processing logic.

[0038] In some implementations, the TPMS 214 maintains an inventory of tracking devices in the tracking device database 216. The TPMS 214 also maintains the association of the asset identifier (ID) and tracking device ID and the logical state or status of each tracking device, such as 'In Use,' 'Available,' 'Begin Journey', 'End Journey', etc. The TPMS 214 also maintains the allocation and availability of tracking devices for logistics and pre-positioning purposes, and may track the health of tracking devices stored in inventory.

[0039] In some implementations, the TPMS 214 allows TL personnel 208 to perform housekeeping functions, such as tracking device forecasts, ordering new tracking devices, detecting lost tracking devices, billing management, salvage and environmental disposal of failed tracking devices, inventory tracking, customer help desk and financial accounting. The TPMS 214 allows TL personnel 208 to monitor the state of a tracking device 211 'in journey', trouble shoot causes for failure in communicating with the event server 212, and locate lost tracking devices. The TPMS 214 provides analytic tools to monitor tracking device network performance (e.g., GPS/GPRS coverage/roaming area for specific trade lanes).

[0040] The tracking device system 200 is one example infrastructure. Other infrastructures are also possible which contain more or fewer subsystems or components than shown in FIG. 2. For example, one or more of the servers or databases shown in FIG. 2 can be combined into a single server or database. As another example, tracking devices can be associated with assets using dedicated handheld devices.

Example Contextually Aware Tracking Device

[0041] FIG. 3 is a block diagram of an example contextually aware tracking device 211. A microprocessor 302 controls the operations of the tracking device 301 that is coupled with the asset 300. The microprocessor can run on different clocks at different times. For example, the microprocessor can run off of a high speed clock when operating, or can run off of a slow speed clock when in sleep mode to conserve power. The microprocessor 302 controls a Global Satellite Navigation System (GNSS) module 304 that is connected to a satellite navigation receive antenna 303. The microprocessor 302 can be awakened by a vibration sensor 306, and can read 3D acceleration measurements from an accelerometer 305. The microprocessor 302 controls a wireless communications module 307 that is connected to a wireless communications transmit/receive antenna 308.

Hidden Markov Model

Introduction to the Hidden Markov Model

[0042] FIG. 4 is a flow diagram of top level logic of the HMM. The tracking device of FIG. 3 receives tracking data (400). Once the tracking device receives tracking data, it utilizes the logic of the HMM to determine if the behavior of the tracking device needs to be adjusted (401). If it does determine that the behavior of the tracking device needs to be adjusted (402), then the behavior of the tracking device is dynamically adjusted (403). The tracking device may increase or decrease the rate with which it gathers tracking data 400.

[0043] FIG. 5 is a diagram illustrating the technical details of the HMM. In this example case, the HMM is finite which means that the space (X) of hidden states 501 of the hidden Markov chain and the set (O) of the observational outputs 500 are both finite. The *transitional probability* of going from hidden state x_i to state x_j is labeled here as a_{ij} and the *conditional probability* of finding an observation, O_k, at state x_i is labeled as b_{ik}.

[0044] FIG. 6 is a diagram showing a sequence of hidden states 601 and their corresponding observable states 600. The distinguishing feature of a HMM is that the sequence of hidden states 601 (x_1, x_2, and so on) are not directly observable. Thus, they can only be predicted using the trained HMM and the sequence of observable states 600. Below is an outline of the HMM that will support later descriptions.

Definitions:

- Observation Sequence: $O = (O_1, O_2, O_3, ..., O_N)$
- Model: λ
- Most likely state at time t: q_t
- $\alpha_t(i)$ accounts for the partial observations up to time t ($O_1, O_2, O_3, ..., O_t$) and state x_i at time t.
- $\beta_t(i)$ accounts for the remaining observations ($O_{t+1}, O_{t+2}, O_{t+3}, ..., O_T$) given state x_i at time t.
- $\gamma_t(i)$ is the probability of being in state x_i at time t, given the Observation sequence O and the model λ. In other words, $\gamma_t(i) = P(q_t = x_i \mid O, \lambda)$.

Since $\gamma_i(t)$ is a probability measure, $\sum_{i=1}^{N} \gamma_i(t) = 1$. We can find

$$\gamma_i(t) = \frac{\alpha_i(t)\beta_i(t)}{P(O \mid \lambda)} = \frac{\alpha_i(t)\beta_i(t)}{\sum_{i=1}^{N} \alpha_i(t)\beta_i(t)}, \text{ where } q_t = \arg\max_{1 \le i \le N}[\gamma_i(t)] \text{ for } 1 \le t \le T.$$

Train the Model with the EM algorithm

[0045] To train the model, we can use a method called the generalized Expectation Maximization (EM), or Baum-Welch, method. This method examines each journey and counts transitions between states for each tracking device. Then it will normalize the probabilities by multiplying each set by a normalizing constant which will ensure that the probability distribution corresponding to each state adds up to 1.

Smoothing out the Data with the Forward-Backward algorithm

[0046] The HMM is most efficient when implementing the training using a method called the Forward-Backward method. Example pseudo-code for implementing the Forward-Backward method is illustrated in FIG. 7. In this method, one uses observations to filter, predict and smooth out the present and past states. This not only makes past observations better, but it also makes future observations more accurate.

[0047] For the forward method, we define the forward variable, $\alpha_t(i)$, first: $\alpha_t(i) = P(O_1, O_2, O_3, ..., O_t, q_t = x_i \mid \lambda)$, where $\alpha_t(i)$ is the probability of the observation sequence $O_1, O_2, O_3, ..., O_t$ and x_i at time t, given the model λ. This probability can be found using a recursive formula:

Step 1: Initialize

$$\alpha_1(i) = \pi_i b_i(o_1) \qquad 1 \le i \le N$$

Step 2: Induction

$$a_{t+1}(j) = \left[\sum_{i-1}^{N} \alpha_t(i)\alpha_{ij}\right] b_j(o_{t+1}) \qquad for\ all\ t = 1,2,...,T-1;\ 1 \le j \le N$$

Step 3: Termination

$$P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

[0048] For the Backward method, we define the forward variable, $\beta_t(i)$, first: $\beta_t(i) = P(O_{t+1}, O_{t+2}, ..., O_T, q_t = x_i \mid \lambda)$, where $\beta_t(i)$ is the probability of the observation sequence $O_{t+1}, O_{t+2}, ..., O_T$ from time t+1 to T and x_i at time t=T, given the model λ. This probability can be found using a recursive formula:

Step 1: Initialize

$$\beta_T(i) = 1 \qquad 1 \le i \le N$$

Step 2: Induction

$$\beta_t(j) = \sum_{j-1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \quad for\ all\ t = T-1, T-2, ...1;\ 1 \le i \le N$$

Step 3: Termination

$$P(O \mid \lambda) = \sum_{i=1}^{N} \pi_i b_i(O_1) \beta_1(i)$$

Making Predictions with the Viterbi method

[0049] In order to use the HMM to make predictions, we can use the Viterbi method. The Viterbi method chooses the best state sequence that maximizes the likelihood of the state sequence for the observation sequence. The Viterbi method can be implemented using a recursive approach which is illustrated in FIG. 8. This method can be used to predict the best state sequence and is one way to implement the HMM.

[0050] The Viterbi method is illustrated as follows: To initialize the recursion, we first define the state variables 800 which define the base case and will feed the subsequent recursive routine. The recursive functions 801 will terminate when the end of the finite state space is reached and the maximums are found 802. This can be used to derive $q_i(t)$ 803

which is the best score (highest probability) along a single path, at time t, which accounts for the first t observations and ends in state x_i. In other words, we can see:

$q_i(t) = \max_{q_1, q_2, \ldots, q_{t-1}} P[q_1 q_2 \ldots q_t = i, O_1 O_2 \ldots O_t \mid \lambda]$. Also, by induction we have,

$q_j(t+1) = [\max_i q_i(t) a_{ij}] \cdot b_j(O_{t+1})$.

Implementing the HMM

[0051] FIG. 9 is a simple illustration of the state and state transitioning of the HMM. In our example HMM implementation, the initial state 900 is known. For each transition from the initial state to a new state, there is an associated transition probability 901. Also, from each state, there is an associated conditional probability 903. The conditional probability 903 is the probability of receiving a particular set of observations 902 given the state. As the HMM is trained, using these probabilities, accurate predictions can be made about the hidden states associated with them. Ultimately, with high probability, they can be used to determine things like the most likely sequence of state transitions.

[0052] FIG. 10 is a block diagram showing the structure of the hybrid model implementation of the HMM. In this implementation of the Hybrid Model, the model 1003 is first trained 1002 on past historical trip data 1000. Once it is trained, the model can be initialized 1001 in the tracking device depicted in FIG. 3. Once the model 1003 is initialized, the tracking device can use the real time observations 1006 as well as the model to predict 1005 the most probable path 1007. Once the most probable path prediction is made, the dynamic behavior of the tracking device can be updated 1008 in accordance with the best prediction 1007. The model 1003 can then be updated 1010 with the newest observation 1009. Since the model 1003 has been updated 1010, then the next prediction from the updated model 1004 will be improved, thereby, allowing the tracking device to be more contextually aware.

[0053] FIG. 11 is a more detailed outline of the Viterbi method outlined above. The Viterbi method is recursive because there are recursive relationships between the states in the HMM according to the Markov property. The Viterbi method returns the best prediction 1104 for the next hidden state at time T for an implementation of the HMM. The method is initialized with assumed values for the initial state 1100 and initial probability 1101. Then, it computes the probability of each state transition 1102 from the current state over the entire set of observations given the predictions for which transition will be most likely. This computation occurs until the entire state space is examined and a maximum probability is

found 1103. To use this process to find the most probable path prediction, we can keep employing this method for each consecutive state, keeping pointers from each state prediction to its previous state, until we reach the final state.

[0054] The forward-backward method is illustrated in more detail in FIG. 12. This method is used to smooth estimates. To initialize this method, it takes in probabilities regarding the first state as well as the initial set of observational values 1200. The forward-backward method uses a vector of forward probabilities up to the current time and returns a set of smoothed estimates up to the current time 1201. The method first filters the data using the set of observational values 1202. Then, the forward probabilities are used along with the backward probabilities to normalize and smooth the data 1203. This can be used but is not limited to training the HMM.

[0055] FIGS. 13A-13C illustrate various examples of dynamically updating tracking behavior, based upon the updated predictions of the HMM. This update of dynamic behavior can include increasing the frequency of collecting location data to increase granularity of the location data 1300, or decreasing the frequency of collecting tracking data to reduce power consumption by the tracking device 1301. This update of dynamic behavior can include increasing the frequency of sending communication reports to reduce the latency of the reports 1302, or decreasing the frequency of sending communication reports to reduce power consumption by the tracking device 1303. This update of dynamic behavior can include using higher accuracy modes of a navigation receiver in the tracking device to increase accuracy of the location data 1304, or lower accuracy modes of the navigation receiver to reduce power consumption by the tracking device 1305. This update of dynamic behavior can include selecting a communication channel from a plurality of available communication channels, and sending communication reports on the selected communication channel based on the region or location of the device 1306. This update of dynamic behavior can include determining whether it is cost-effective to report the event at the particular time based at least in part on the type of the event and the cost and resources required for reporting based on the region or location of the device 1307. This update of dynamic behavior can include determining the list of preferred communication channels and the list of excluded communication channels corresponding to the particular geographic region in which the asset is currently located 1308. This update of dynamic behavior can include determining an occurrence of a supply chain event 1309, including a gate in or gate out to or from a supply chain node 1310. This update of dynamic behavior can include determining an occurrence of change of custody of the asset being shipped 1311. This update of dynamic behavior can

include determining an updated dynamic estimated time of arrival of the asset at the destination 1312. This update of dynamic behavior can include determining an occurrence of a stuck shipment exception condition 1313.

[0056] Other actions to update dynamics behavior of the tracking device based upon the context of the asset shipment are possible, in other embodiments beyond the specific examples shown in FIG. 13.

[0057] Multiple HMM-enabled tracking devices can share data by traversing the same tradelanes. FIG. 14 shows the tracking device being activated 1400 to start retrieving contextual data. After its journey is completed and the tracking device is deactivated 1401, the tracking device will arrive 1402 at a central location where its data will be uploaded 1403 to a central HMM repository 1404. This HMM repository 1404 can be used but is not limited to processing all incoming data from various devices and storing the data. Also, using the HMM repository 1404, data can be downloaded into any new tracking devices that can use the data. In this way, other tracking devices that are departing 1406 will be able to download 1405 and use any data relevant to its journey that was gathered from any previously arriving tracking devices 1402.

Structure of the Hybrid Model

[0058] Due to the battery constraints on the tracking device, location data can be collected at discrete intervals rather than in a continuous manner. This restricts the amount of data that the device can accrue to train the HMM. Before the tracking device can confidently use the HMM, the tracking device can gather data across its state space. If the model defines the state space as all locations in range from the tracking device, the state space would be large. This means that to train the device on a tradelane, the device would need to perform many journeys on a tradelane before it is able to use the HMM to make accurate predictions.

[0059] An additional limitation due to battery power consumption, is that the tracking device is constrained as to its available processing power. With a large state space, the HMM would require a large amount of memory and processing power from the tracking device in order to make predictions.

[0060] To reduce the time that it takes to train the model on a tradelane and to reduce the amount of time and power that it takes to process HMM predictions on the device, the HMM can be implemented as a Hybrid Model. In some implementations, the Hybrid model uses geofence zones as quantized states, to significantly reduce the state space.

Technical Overview of the Hybrid Model

[0061] The Hybrid Model may be described as one specific implementation. Other implementations are also possible. The *State Space* $x \in X$ at any time t for an interval n is the range of possible locations and activities $x_n^t = \{gz_n^t, a_n^t\}$ where $gz \in$ Geofence Zone and $a \in$ {stationary, in motion} denotes activity. In this case, the state of being in a geofence zone is not directly observable because of the "noise," or inaccuracies, in the location measurements and the deviations of the asset vehicle from the predicted path. Each observation, O, for a given time t is a measurable collection of data: sensor data and tracking data.

[0062] The formula for the transition probability of this model is give here (Equation 1):

(1) $$P(X^t = x_j \mid X^{t-1} = x_i) = P(a_j, gz_j \mid a_i, gz_i) = P(a_j \mid a_i, gz_i) \cdot P(gz_j \mid gz_i, a_i)$$

Transition of Zone: $P(gz_j \mid gz_i, a_i)$ is the probability of transition to a new geofence zone at time t given the previous geofence zone and previous activity.

Transition of Activity: $P(a_j \mid a_i, gz_i)$ models the probability of whether or not the tracking device is moving at time t. The model is initialized with the information that, in most cases, moving tracking devices continue to move and non-moving ones continue not to move.

Nested Geofence Models

[0063] The Hidden Markov Method and Hybrid Model provide context for tradelanes in which historical data is gathered to train the model. This historical data will not be available when first commencing shipments through a new tradelane. This historical data will also not be available, for shipments that stray off a planned tradelane due to supply chain exception events into new tradelanes without historical data. A Nested Geofences model can provide a capability for recognizing context without the need to train on previous historical data.

[0064] FIG. 15 illustrates a notional example of nested geofences around the Port of Oakland terminal. The distance between the geofences is drawn close together to aid in visibility, whereas in practice the geofences might be much farther apart. Note that an irregularly shaped terminal can be represented by multiple touching rectangular geofences. As a tracking device on an asset enters successive internal geofences, the context of the shipment can be recognized to be approaching closer and closer to the center node of the geofence. Conversely, exiting successive geofences can be used to recognized departure away from the center node of the geofence.

[0065] FIG. 16 illustrates top level logic of the Nested Geofences method. Tracking data 1600 is used by a nested geofences model 1601. If the nested geofences model 1601

determines that the dynamic behavior of the tracking device should be adjusted 1602, then the adjustment is made to the dynamic behavior of the tracking device 1603.

[0066] FIG. 17 illustrates a next level of detail of the logic of the Nested Geofences method. A shipment can be determined to enter a geofence 1700, exit a geofence 1702 or persist within a geofence past a time threshold 1704. Entering a geofence 1700 is based upon being outside of a geofence for one location update, then inside the geofence for the next location update. Similarly, exiting a geofence 1702 is based upon being inside of a geofence for one location update, then outside the geofence for the next location update. Persisting within a geofence 1704 is based upon no change to a geofence status past a time threshold. Due to variance in accuracy of locating a tracking device on a shipment, there can be false alarms of entering a geofence, or missed detects of entering a geofence. These false geofence statuses can be suppressed by integrating over time, to require N consecutive locations inside or outside of a geofence to declare a state change.

[0067] Based upon the geofence status, the tracking device dynamic behavior may be updated based upon entering a geofence 1701, exiting a geofence 1703 or persisting within a geofence 1705.

[0068] Various actions can be defined to be taken to update the behavior of a tracking device, based upon the trigger of entering a geofence 1700. This update of dynamic behavior 1701 can include increasing the frequency of collecting location data to increase granularity of the location data. This update of dynamic behavior 1701 can include increasing a frequency of sending communication reports to reduce a latency of the reports. This update of dynamic behavior 1701 can include using higher accuracy modes of a navigation receiver in the tracking device to increase accuracy of the location data. This update of dynamic behavior 1701 can include selecting a communication channel from a plurality of available communication channels based upon the region or location of the device. This update of dynamic behavior 1701 can include determining whether it is cost-effective to report an event at the particular time based at least in part on a type of the event and the cost and resources required for reporting for the region or location of the device. This update of dynamic behavior 1701 can include utilizing preferred communication channels or excluding communication channels corresponding to a particular geographic region in which the asset is currently located. This update of dynamic behavior 1701 can include determining an occurrence of a supply chain event, including a gate in to a supply chain node. This update of dynamic behavior 1701 can include determining an occurrence of change of custody of the

asset being shipped. This update of dynamic behavior 1701 can include determining an updated dynamic estimated time of arrival of the asset at the destination.

[0069] Various actions can be defined to be taken to update the behavior of a tracking device, based upon the trigger of exiting a geofence 1702. This update of dynamic behavior 1703 can include reducing the frequency of collecting location data to reduce power consumption by the tracking device. This update of dynamic behavior 1703 can include reducing a frequency of sending communication reports to reduce power consumption by the tracking device. This update of dynamic behavior 1703 can include using lower accuracy modes of the navigation receiver to reduce power consumption by the tracking device. This update of dynamic behavior 1703 can include selecting a communication channel from a plurality of available communication channels based upon the region or location of the device. This update of dynamic behavior 1703 can include determining whether it is cost-effective to report an event at the particular time based at least in part on a type of the event and the cost and resources required for reporting for the region or location of the device. This update of dynamic behavior 1703 can include utilizing preferred communication channels or excluded communication channels corresponding to a particular geographic region in which the asset is currently located. This update of dynamic behavior 1703 can include determining an occurrence of a supply chain event, including a gate out from a supply chain node. This update of dynamic behavior 1703 can include determining an occurrence of change of custody of the asset being shipped. This update of dynamic behavior 1703 can include determining an updated dynamic estimated time of arrival of the asset at the destination.

[0070] Various actions can be defined to be taken to update the behavior of a tracking device, based upon the trigger of persisting within a geofence 1704. This update of dynamic behavior 1705 can include reducing the frequency of collecting location data to reduce power consumption by the tracking device. This update of dynamic behavior 1705 can include reducing a frequency of sending communication reports to reduce power consumption by the tracking device. This update of dynamic behavior 1705 can include using lower accuracy modes of the navigation receiver to reduce power consumption by the tracking device. This update of dynamic behavior 1705 can include determining whether it is cost-effective to report an event at the particular time based at least in part on a type of the event and the cost and resources required for reporting for the region or location of the device. This update of dynamic behavior 1705 can include determining an updated dynamic estimated time of arrival of the asset at the destination. This update of dynamic behavior 1705 can include determining an occurrence of a stuck shipment exception condition.

[0071] These updates of dynamic behavior 1705 can be based on accumulating data from multiple tracking devices, and downloading contextual data to each tracking device.

[0072] Other actions to update dynamics behavior of the tracking device based upon the context of the asset shipment are possible, in other embodiments beyond the specific examples.

Sensor Model

[0073] The Hidden Markov Method, Hybrid Model and Nested Geofence Model provide context for shipments in motion. An additional capability is required to determine context of shipments at rest. FIG. 18 illustrates the top level logic of the Sensor Pattern Matching method. Sensor data 1800 is used by the Sensor Pattern Matching model 1801. The model 1801 will match a group of sensor data 1800 against a library of patterns. If the Sensor Pattern Matching model 1801 finds a match and determines that the dynamic behavior of the tracking device should be adjusted 1802, then the adjustment is made to the dynamic behavior of the tracking device 1803.

[0074] FIG. 19 is a flow diagram showing the decision tree of the Sensor Pattern Matching method illustrated in FIG. 18. Sensor data is used by the model to determine whether the motion sensor values exceed a threshold 1900. Because of the need to conserve the life of the battery for operations, the tracking device is in a low power operational state. The tracking device uses the motion sensor as a trigger to wakeup to perform a specific action. If it is determined that the motion exceeds a threshold, the tracking device is triggered to wakeup and collect motion data 1901 (e.g., accelerometer data). The accelerometer data is then matched to patterns in a stored pattern library 1902. If a pattern match is found 1903, then the tracking device will classify the motion in the supply chain context 1904. This can lead to determining that the behavior of the tracking device needs to be updated.

[0075] Various actions can be defined to be taken to update the behavior of a tracking device, based upon matching a Sensor Pattern in a supply chain context. This update of dynamic behavior can include increasing the frequency of collecting location data to increase the granularity of the location data. This update of dynamic behavior can include reducing the frequency of collecting location data to reduce power consumption by the tracking device. This update of dynamic behavior can include increasing the frequency of sending communication reports to reduce the latency of the reports. This update of dynamic behavior can include reducing the frequency of sending communication reports to reduce power consumption by the tracking device. This update of dynamic behavior can include using higher accuracy modes of a navigation receiver in the tracking device to increase accuracy of

the location data. This update of dynamic behavior can include using lower accuracy modes of the navigation receiver to reduce power consumption by the tracking device. This update of dynamic behavior can include determining whether it is cost-effective to report an event at the particular time based at least in part on a type of the event and the cost and resources required for reporting for the region or location of the device. This update of dynamic behavior can include determining an occurrence of a supply chain event, including a gate in to or a gate out from a supply chain node, or including a crane load to a ship or a crane load from a ship. This update of dynamic behavior can include determining an occurrence of change of custody of the asset being shipped. This update of dynamic behavior can include determining an updated dynamic estimated time of arrival of the asset at the destination. This update of dynamic behavior can include determining an occurrence of a stuck shipment exception condition.

[0076] These updates of dynamic behavior can be based on accumulating data from multiple tracking devices; and downloading contextual data to each tracking device.

[0077] Other actions to update dynamics behavior of the tracking device based upon the context of the asset shipment are possible, in other embodiments beyond the specific examples.

Combined Models

[0078] The logic behind the system that combines the Nested Geofence method, the Hybrid Model method and the Sensor Pattern Matching method is illustrated in the flow diagram of FIG. 20. The tracking device is enabled with the Hybrid model (2000) and acquires historical data 2001 and tracking and sensor data 2002. At a given time in its journey, using this data, it determines whether it is in motion or stationary (2003). If the tracking device is in motion, it looks to see if there exists historical data that it can use for its current location (2004). If it does not hold historical data for its current location (2004), then it relies on the Nested Geofence method (2005), which requires no historical data to use; otherwise, it uses the Hybrid Model (2006), which will use both the historical data and the current tracking and sensor data to make accurate predictions. Finally, if the tracking device is stationary, it will utilize the Sensor Pattern Matching method (2007).

[0079] FIG. 21 illustrates a system with a processor, which is embedded in a tracking device 2101 and attached to an asset 2100. The tracking device 2101 uses a communication interface 2103 to send and receive data to and from a central repository 2104. The processor used by the tracking device acquires sensor and tracking data and is configured to utilize the HMM, the Nested Geofences or Sensor Pattern Matching models 2102. The processor then

uses the conclusions gathered from the models to adjust the dynamic behavior of the tracking device 2101, using the communication interface 2104 to make reports to a central repository 2104.

[0080] The combined models provide a comprehensive set of methods to determine when to dynamically update the behavior of a tracking device. This update of dynamic behavior can include increasing the frequency of collecting location data to increase the granularity of the location data. This update of dynamic behavior can include reducing the frequency of collecting location data to reduce power consumption by the tracking device. This update of dynamic behavior can include increasing the frequency of sending communication reports to reduce the latency of the reports. This update of dynamic behavior can include reducing the frequency of sending communication reports to reduce power consumption by the tracking device. This update of dynamic behavior can include using higher accuracy modes of a navigation receiver in the tracking device to increase accuracy of the location data. This update of dynamic behavior can include using lower accuracy modes of the navigation receiver to reduce power consumption by the tracking device. This update of dynamic behavior can include selecting a communication channel from a plurality of available communication channels based upon the region or location of the device. This update of dynamic behavior can include determining whether it is cost-effective to report an event at the particular time based at least in part on a type of the event and the cost and resources required for reporting for the region or location of the device. This update of dynamic behavior can include utilizing preferred communication channels or excluding communication channels corresponding to a particular geographic region in which the asset is currently located. This update of dynamic behavior can include determining an occurrence of a supply chain event, including a gate in to or a gate out from a supply chain node, or including a crane load to a ship or a crane load from a ship. This update of dynamic behavior can include determining an occurrence of change of custody of the asset being shipped. This update of dynamic behavior can include determining an updated dynamic estimated time of arrival of the asset at the destination. This update of dynamic behavior can include determining an occurrence of a stuck shipment exception condition.

[0081] These updates of dynamic behavior can be based on accumulating data from multiple tracking devices; and downloading contextual data to each tracking device.

[0082] Other actions to update dynamics behavior of the tracking device based upon the context of the asset shipment are possible, in other embodiments beyond the specific examples.

[0083] The features described above can be implemented in digital electronic circuitry, or in computer hardware, firmware, software, or in combinations of them. The features can be implemented in a computer program product tangibly embodied in a computer readable medium, e.g., in a machine-readable storage device, for execution by a programmable processor; and method steps can be performed by a programmable processor executing a program of instructions to perform functions of the described implementations by operating on input data and generating output.

[0084] The described features can be implemented advantageously in one or more computer programs that are executable on a programmable system including at least one programmable processor coupled to receive data and instructions from, and to transmit data and instructions to, a data storage system, at least one input device, and at least one output device. A computer program is a set of instructions that can be used, directly or indirectly, in a computer to perform a certain activity or bring about a certain result. A computer program can be written in any form of programming language (e.g., Objective-C, Java), including compiled or interpreted languages, and it can be deployed in any form, including as a stand-alone program or as a module, component, subroutine, or other unit suitable for use in a computing environment.

[0085] Suitable processors for the execution of a program of instructions include, by way of example, both general and special purpose microprocessors, and the sole processor or one of multiple processors or cores, of any kind of computer. Generally, a processor will receive instructions and data from a read-only memory or a random access memory or both. The essential elements of a computer are a processor for executing instructions and one or more memories for storing instructions and data. Generally, a computer will also include, or be operatively coupled to communicate with, one or more mass storage devices for storing data files; such devices include magnetic disks, such as internal hard disks and removable disks; magneto-optical disks; and optical disks. Storage devices suitable for tangibly embodying computer program instructions and data include all forms of non-volatile memory, including by way of example semiconductor memory devices, such as EPROM, EEPROM, and flash memory devices; magnetic disks such as internal hard disks and removable disks; magneto-optical disks; and CD-ROM and DVD-ROM disks. The processor and the memory can be supplemented by, or incorporated in, ASICs (application-specific integrated circuits).

[0086] To provide for interaction with a user, the features can be implemented on a computer having a display device such as a CRT (cathode ray tube) or LCD (liquid crystal

display) monitor for displaying information to the user and a keyboard and a pointing device such as a mouse or a trackball by which the user can provide input to the computer.

[0087] The features can be implemented in a computer system that includes a back-end component, such as a data server, or that includes a middleware component, such as an application server or an Internet server, or that includes a front-end component, such as a client computer having a graphical user interface or an Internet browser, or any combination of them. The components of the system can be connected by any form or medium of digital data communication such as a communication network. Examples of communication networks include, e.g., a LAN, a WAN, and the computers and networks forming the Internet.

[0088] The computer system can include clients and servers. A client and server are generally remote from each other and typically interact through a network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other.

[0089] A number of implementations have been described. Nevertheless, it will be understood that various modifications may be made. For example, elements of one or more implementations may be combined, deleted, modified, or supplemented to form further implementations. As yet another example, the logic flows depicted in the figures do not require the particular order shown, or sequential order, to achieve desirable results. In addition, other steps may be provided, or steps may be eliminated, from the described flows, and other components may be added to, or removed from, the described systems. Accordingly, other implementations are within the scope of the following claims.

CLAIMS

WHAT IS CLAIMED IS:

1. A computer-implemented method, comprising:

receiving tracking data for an asset from a tracking device coupled to the asset, the tracking data including a location of the asset and a timestamp corresponding to the location;

utilizing a Hidden Markov Model to assess a shipping context of the asset based on the received tracking data;

determining that a dynamic behavior of the tracking device is to be adjusted based on the shipping context; and

adjusting the dynamic behavior of the tracking device for the asset.

2. The method of claim 1, wherein adjusting the dynamic behavior of the tracking device, comprises:

adjusting a frequency of collecting location data to increase granularity of the location data, or to reduce power consumption by the tracking device.

3. The method of claim 1, wherein adjusting the dynamic behavior of the tracking device , comprises:

adjusting a frequency of sending communication reports to reduce a latency of the reports, or to reduce power consumption by the tracking device.

4. The method of claim 1, wherein adjusting the dynamic behavior of the tracking device comprises:

using higher accuracy modes of a navigation receiver in the tracking device to increase accuracy of the location data, or lower accuracy modes of the navigation receiver to reduce power consumption by the tracking device.

5. The method of claim 1, wherein adjusting the dynamic behavior of the tracking device comprises:

selecting a communication channel from a plurality of available communication channels; and sending communication reports on the selected communication channel.

6. The method of claim 5, wherein it is determined whether it is cost-effective to report the event at the particular time is based at least in part on a type of the event and the cost and resources required for reporting.

7. The method of claim 5, wherein the list of preferred communication channels and the list of excluded communication channels correspond to a particular geographic region in which the asset is currently located.

8. The method of claim 1, wherein determining the shipping context comprises: determining an occurrence of a supply chain event.

9. The method of claim 8, wherein the supply chain event is at least one of a gate in to a supply chain node or gate out from a supply chain node.

10. The method of claim 1, wherein determining the shipping context comprises: determining an occurrence of change of custody of the asset being shipped.

11. The method of claim 1, wherein determining the shipping context comprises: determining an updated dynamic estimated time of arrival of the asset at the destination.

12. The method of claim 1, wherein determining the shipping context comprises: determining an occurrence of a stuck shipment exception condition.

13. The method of claim 1, further comprising: accumulating data from multiple tracking devices; and downloading contextual data to each tracking device.

14. A computer-implemented method, comprising:

receiving tracking data for an asset from a tracking device coupled to the asset, the tracking data including a location of the asset and a timestamp corresponding to the location;

utilizing nested geofences to assess a shipping context of the asset based on the received tracking data;

determining that a dynamic behavior of the tracking device is to be adjusted based on the shipping context; and

adjusting the dynamic behavior of the tracking device for the asset.

15. A computer-implemented method, comprising:

receiving sensor data for an asset from a tracking device coupled to the asset, the sensor data including one or more of vibration data, velocimeter data, and accelerometer data of the asset and a timestamp corresponding to the sensor data;

utilizing pattern matching to assess a shipping context of the asset based on the received sensor data;

determining that a dynamic behavior of a tracking device coupled to the asset is to be adjusted; and

adjusting the dynamic behavior of the tracking device for the asset.

16. The method of claim 15, wherein determining the shipping context comprises: determining an occurrence of a supply chain event.

17. The method of claim 16, wherein the supply chain event is at least one of a crane load to a ship or a crane unload from a ship.

18. A computer-implemented method, comprising:

receiving tracking data for an asset from a tracking device coupled to the asset, the tracking data including a location of the asset and a timestamp corresponding to the location;

receiving sensor data for an asset from the tracking device, the sensor data including one or more of vibration data, velocimeter data, and accelerometer data of the asset at the location;

determining if the shipment is stationary or in motion;

if the shipment is stationary, selecting a vibration or acceleration pattern matching model;

if the shipment is in motion, selecting at least one of a Hidden Markov model with historical data for a tradelane of the shipment, or a nested geofence model without historical data for the tradelane of the shipment;

assessing a shipping context of the asset based on the selected model and the received tracking or sensor data;

determining that a dynamic behavior of the tracking device is to be adjusted; and

adjusting the dynamic behavior of the tracking device for the asset.

19. A system for tracking an asset based on shipping context, the system comprising:

a communication interface configured for communicating with a tracking device coupled to the asset, the communication interface configured for receiving tracking data for the asset from the tracking device, the tracking data including a location of the asset and a timestamp corresponding to the location;

a processor coupled to the communication interface and configured for:

utilizing a Hidden Markov Model to assess a shipping context of the asset based on the received tracking data;

determining that a dynamic behavior of the tracking device is to be adjusted based on the shipping context; and

adjusting the dynamic behavior of the tracking device for the asset.

20. A system for tracking an asset based on shipping context, the system comprising:

a communication interface configured for receiving tracking data for an asset from a tracking device coupled to the asset, the tracking data including a location of the asset and a timestamp corresponding to the location;

a processor coupled to the communication interface and configured for:

utilizing nested geofences to assess a shipping context of the asset based on the received tracking data;

determining that a dynamic behavior of the tracking device is to be adjusted based on the shipping context; and

adjusting the dynamic behavior of the tracking device for the asset.

21. A system for tracking an asset based on shipping context, the system comprising:

a communication interface configured for receiving sensor data for an asset from one or more sensors from a tracking device coupled to the asset, the sensor data including one or more of vibration data, velocimeter data, and accelerometer data of the asset and a timestamp corresponding to the sensor data;

a processor coupled to the communication interface and configured for:

utilizing pattern matching to assess a shipping context of the asset based on the received sensor data;

determining that a dynamic behavior of a tracking device coupled to the asset is to be adjusted; and

adjusting the dynamic behavior of the tracking device for the asset.

22. A system for tracking an asset based on shipping context, the system comprising:

a communication interface configured for receiving tracking data for an asset from a tracking device coupled to the asset, the tracking data including a location of the asset and a timestamp corresponding to the location, and for receiving sensor data for an asset from the tracking device, the sensor data including one or more of vibration data, velocimeter data, and accelerometer data of the asset at the location;

a processor coupled to the communication interface and configured for:

determining if the shipment is stationary or in motion based on the sensor data or location data;

if the shipment is stationary, selecting a vibration or acceleration pattern matching model;

if the shipment is in motion, selecting at least one of a Hidden Markov model with historical data for a tradelane of the shipment, or a nested geofence model without historical data for the tradelane of the shipment;

assessing a shipping context of the asset based on the selected model and the received tracking or sensor data;

determining that a dynamic behavior of the tracking device is to be adjusted; and

adjusting the dynamic behavior of the tracking device for the asset.