

SYSTEM AND METHOD FOR ADAPTIVELY PREDICTING A ROAD SEGMENT ATTRIBUTE BASED ON A GRAPH INDICATIVE OF RELATIONSHIP BETWEEN A ROAD SEGMENT AND A DETECTION

Technical Field

[0001] The present invention relates generally to technology and, in particular, to a system and method for adaptively adaptively predicting a road segment attribute e.g. turn restrictions, speed limits based on a graph indicative of a relationship between a road segment and a detection. Examples of the detections include traffic signs that are collected from images retrieved from road segments.

Background

[0002] Having an up-to-date map is extremely useful in a variety of contexts and applications. The traditional way for updating maps can involve a tremendous amount of human effort. For that reason, there are various conventional methods of collecting relevant map information. One of the which is based on collecting, analysing and interpreting street level imagery.

[0003] There are multiple efforts done with an intention to extract relevant information from street level imagery for mapping. However, none of the conventional technique is able to adaptively predict road segments attribute e.g. turn restrictions, speed limits based on a graph indicative of a relationship between a road segment and a detection.for a selected road network. The relationship between the road segments and detections being represented as a graph. A need therefore exists to provide a system and method to address the above-mentioned problems.

Summary

[0004] According to a first aspect of the present disclosure, there is provided a compute-implemented method for adaptively predicting a road segment attribute based on a a graph that is indicative of a relationship between a road segment and a detection, the method comprising: identifying the road segment and its corresponding feature;

identifying the detection and its corresponding feature, the feature being one of detection confidence, geo-positioning confidence, distance from the camera; and

adaptively providing a graph on a relationship between the road segment and the detection based on the respective features of the road segment and the detection so as to predict the road segment attribute.

[0005] According to a second aspect of the present disclosure, there is provided a server a road segment attribute based on a graph that is indicative of a relationship between a road segment and a detection, the server comprising: at least one processor; and at least one memory including computer program code; the at least one memory and the computer program code configured to, with the at least one processor, cause the server at least to:

identify the road segment and its corresponding feature,

identify the detection and its corresponding feature, the feature being one of detection confidence, geo-positioning confidence, distance from the camera; and

adaptively provide a graph on a relationship between the road segment and the detection based on the respective features of the road segment and the detection so as to predict the road segment attribute.

Brief Description of the Drawings

[0006] Embodiments and implementations are provided by way of example only, and will be better understood and readily apparent to one of ordinary skill in the art from the following written description, read in conjunction with the drawings, in which:

[0007] Fig. 1 is a block diagram of a system for providing a graph in accordance with an aspect of the present disclosure.

[0008] Fig. 2 is a flow chart diagram of a method for adaptively providing a graph indicative of a relationship between a road segment and a detection by the graph generating server in the system of Fig. 1.

[0009] Fig. 3 shows a diagram of how a graph can be provided for a road network having a plurality of road segments according to an embodiment of the method in Fig. 2.

[0010] Fig. 4 shows a diagram of how a road network can be represented as an undirected graph according to an embodiment of the method in Fig. 2.

[0011] Fig 5a shows a diagram of how a road network including directions can be represented in a graph according to an embodiment of the method in Fig. 2.

[0012] Fig. 5b shows a diagram of how a road network including how each individual detection attends to each road segment according to an embodiment of the method in Fig. 2.

[0013] Fig 5c shows a diagram of how a road network including how each individual detection attends to direction of each road segment according to an embodiment of the method in Fig. 2.

[0014] Fig. 6 shows a graph according to an embodiment of the method in Fig. 2.

[0015] Figs. 7 and 8 form a schematic block diagram of a computer system upon which a graph generating server in the system of Fig. 1 can be practiced.

[0016] Fig. 9 shows an example of a computing device to realize the graph generating server shown in Fig. 1.

Detailed Description

Terms Description

[0017] The graph generating server is usually managed by a provider that may be an entity (e.g. a company or organization) which operates to process requests, manage data and provide/ display graphical representations that are related to a road network or a plurality of road segments and detections. The graph generating server centralizes the detections from the images collected in each area, and for all users and uses the input to generate the graph. The server may include one or more computing devices that are used for processing data and providing customisable services depending on situations. The graph generating server is also one that is configured to adaptively predict road segment attribute e.g. turn restrictions, speed limits based on a graph indicative of a relationship between a road segment and a detection.

[0018] The graph generating server manages graph generating information of users and the interactions between users and other external servers, along with the data that is exchanged.

[0019] Example Implementations

[0020] Where reference is made in any one or more of the accompanying drawings to steps and/or features, which have the same reference numerals, those steps and/or features have for

the purposes of this description the same function(s) or operation(s), unless the contrary intention appears.

[0021] It is to be noted that the discussions contained in the "Background" section and that above relating to prior art arrangements relate to discussions of devices which form public knowledge through their use. Such should not be interpreted as a representation by the present inventor(s) or the patent applicant that such devices in any way form part of the common general knowledge in the art.

[0022] The system 100

Fig. 1 illustrates a block diagram of a system 100 for providing a graph for a road network. The system 100 comprises a requestor device 102, a graph generating server 108, a remote assistance server 140, remote assistance hosts 150A to 150N, and sensors 142A to 142N. In various embodiments, the disclosure can be applied for graph generating at a server when a road attribute is detected after detecting detections from a plurality of image data from multiple anonymized users.

[0023] The requestor device 102 is in communication with a graph generating server 108 and/or a remote assistance server 140 via a connection 116 and 121, respectively. The connection 116 and 121 may be wireless (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet). The connection 116 and 121 may also be that of a network (e.g., the Internet).

[0024] The graph generating server 108 is further in communication with the remote assistance server 140 via a connection 120. The connection 120 may be over a network (e.g., a local area network, a wide area network, the Internet, etc.). In one arrangement, the graph generating server 108 and the remote assistance server 140 are combined and the connection 120 may be an interconnected bus.

[0025] The remote assistance server 140, in turn, is in communication with the remote assistance hosts 150A to 150N via respective connections 122A to 122N. The connections 122A to 122N may be a network (e.g., the Internet).

[0026] The remote assistance hosts 150A to 150N are servers. The term host is used herein to differentiate between the remote assistance hosts 150A to 150N and the remote assistance server 140. The remote assistance hosts 150A to 150N are collectively referred to herein as the remote assistance hosts 150, while the remote assistance host 150 refers to one of the remote

assistance hosts 150. The remote assistance hosts 150 may be combined with the remote assistance server 140.

[0027] In an example, the remote assistance host 150 may be one managed by an entity and the remote assistance server 140 is a central server that provides graphs at an organization level and decides which of the remote assistance hosts 150 to forward data or retrieve data like image inputs.

[0028] Sensors 142A to 142N are connected to the remote assistance server 140 or the graph generating server 108 via respective connections 144A to 144N or 144A to 144N. The sensors 142A to 142N are collectively referred to herein as the sensors 146A to 146N. The connections 144A to 144N are collectively referred to herein as the connections 144, while the connection 144 refers to one of the connections 144. Similarly, the connections 146A to 146N are collectively referred to herein as the connections 146, while the connection 146 refers to one of the connections 146. The connections 144 and 146 may be wireless (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet). The sensor 146 may be one of an image capturing device, video capturing device, and motion sensor and may be configured to send an input depending its type, to at least one of the graph generating server 108. The sensors 146 captures inputs including images of a road segment or a road segment attribute and sends the captured inputs to the server 108 which will then provide a graph.

[0029] In the illustrative embodiment, each of the devices 102 and 142; and the servers 108, 140, and 150 provides an interface to enable communication with other connected devices 102 and 142 and/or servers 108, 140, and 150. Such communication is facilitated by an application programming interface ("API"). Such APIs may be part of a user interface that may include graphical user interfaces (GUIs), Web-based interfaces, programmatic interfaces such as application programming interfaces (APIs) and/or sets of remote procedure calls (RPCs) corresponding to interface elements, messaging interfaces in which the interface elements correspond to messages of a communication protocol, and/or suitable combinations thereof.

[0030] Use of the term 'server' herein can mean a single computing device or a plurality of interconnected computing devices which operate together to perform a particular function. That is, the server may be contained within a single hardware unit or be distributed among several or many different hardware units.

[0031] The remote assistance server 140

[0032] The remote assistance server 140 is associated with an entity (e.g. a company or organization or moderator of the service). In one arrangement, the remote assistance server 140 is owned and operated by the entity operating the server 108. In such an arrangement, the remote assistance server 140 may be implemented as a part (e.g., a computer program module, a computing device, etc.) of server 108.

[0033] The requestor device 102

[0034] The requestor device 102 is associated with a subject (or requestor) who is a party to a request that starts at the requestor device 102. The requestor may be a concerned member of the public who is assisting to get data necessary to obtain a graphical representation of a network graph. The requestor device 102 may be a computing device such as a desktop computer, an interactive voice response (IVR) system, a smartphone, a laptop computer, a personal digital assistant computer (PDA), a mobile computer, a tablet computer, and the like.

[0035] In one example arrangement, the requestor device 102 is a computing device in a watch or similar wearable and is fitted with a wireless communications interface.

[0036] The graph generating server 108

[0037] The graph generating server 108 is as described above in the description section.

[0038] The graph generating server 108 is configured to adaptively provide a graph by performing at least identifying the plurality of road segments in a road network, and identifying the plurality of features. The graph generating server is also one that is configured to adaptively predict a road segment attribute e.g. turn restrictions, speed limits based on a graph indicative of a relationship between a road segment and a detection.

[0039] The remote access hosts 150

[0040] The remote access host 150 is a server associated with an entity (e.g. a company or organization) which manages (e.g. establishes, administers) information regarding information relating to a subject or a member of an organisation. In one embodiment, the server stores information relating to images relating to road segments.

[0041] In one arrangement, the entity is an organisation. Therefore, each entity operates a remote access host 150 to manage the resources by that entity. In one arrangement, a remote access host 150 receives a request that a graph for a remote network is requested. The remote

access host 150 may then arrange to send resources to graph generating server in response to the plurality of road networks identified in the request. For example, the host may be one that is configured to obtain relevant video or image input for processing.

[0042] Advantageously, such information is valuable to predict a road segment attribute e.g. turn restrictions, speed limits based on a graph indicative of a relationship between a road segment and a detection.

[0043] This disclosure uses machine learning techniques for detection and geo-localisation. Additionally, the detection's map-matching is also done using machine learning techniques, for example, applying Graph Neural Networks (GNN) for expressing the relationship between the road segments and detections. In an example, not only the individual or just one type of detected traffic signs but all the detected traffic signs from a given area is utilised, regardless of the type that they are.

[0044] An extended set of detection's features is used to further augment the available input information (e.g. detections confidence, aspect ratio, distance from camera, etc).

[0045] In an embodiment, input data like the road classes and the spatial information about the road network is also used in providing the graph. Such data may be encoded as a graph.

[0046] The input data like road classes, spatial information about the road network is encoded or embedded, transformed and aggregated. GNN may be used for this purpose. This may be trained in a supervised manner, in which each node (road segment for one direction) is classified. The output classes are selected based on relevancy for the purpose of mapping a road segment attribute (e.g one way, give way, speed limit, etc).

[0047] Sensor 142

[0048] The sensor 142 is associated with a user associated with the requestor device 102. More details of how the sensor may be utilised will be provided below.

[0049] Fig. 2 is a flow chart diagram of a method for adaptively providing a graph indicative of a relationship between a road segment and a detection by the graph generating server in the system of Fig. 1. The method shown in Fig. 2 begins with step 202 by identifying identifying the road segment and its corresponding feature. Step 204 follows step 202 in which the method includes identifying the detection and its corresponding feature, the feature being one of detection confidence, geo-positioning confidence, distance from the camera. Step 206 follows

step 204 in which the method adaptively providing a graph on a relationship between the road segment and the detection based on the respective features of the road segment and the detection.

[0050] Fig. 3 shows a diagram of how a graph can be provided for on a road network having a plurality of road segments according to an embodiment of the method in Fig. 2.

[0051] In the example of Fig. 3, a sample road network 300, where each road segment has an identification number from 1 to 7 represented by 302, is shown. There are five individual traffic signs detections (or detection) 304 of type “turn restriction right”, each one identified with a letter from A to E. In an embodiment, there may be inherent geo-positioning error for at least one of the traffic signs detections 304. As such, the traffic signs detections 304 may not be located in the correct, exact position, each one having an inherent geo-positioning error. Also, each traffic light detection has a set of individual features of at least one of a detection confidence, geo-positioning confidence, distance from the camera. The following disclosure discloses the impact on the road segments attributes (or road segment features) based on the traffic signs detections. For example, how the method determines which attribute should be added to which road segment to express correctly detected turn restrictions.

[0052] Fig. 4 shows a diagram of how a road network can be represented as an undirected graph according to an embodiment of the method in Fig. 2. The road network is expressed as a graph in which each road segment is expressed as a node 402 and each edge 404 is expressed as a connectively to the road segment. The edge could be expressed in response to a traffic sign, the graph showing an attribute corresponding to a related one of the plurality of the road segments based on the detected traffic sign (or detection).

[0053] Fig. 5a shows a diagram of how a road network including directions can be represented in a graph according to an embodiment of the method in Fig. 2. Each road segment direction (or detection) affects each road segment in the road network. For example, a detection (e.g one way or turn restriction is an attribute applicable to only one direction for a road segment). As such one node for each road segment direction is created, each direction being marked with one of the letters “a” or “b” as shown as 501A and 501B.

[0054] Fig. 5b shows a diagram of how a road network including how each individual detection attends to each road segment according to an embodiment of the method in Fig. 2. In 550 as shown in Fig. 5b, distance between each road segment node and each road segment detection is measured and then determined if it is below a predetermined threshold (e.g., 15 metres). In

the event that it is determined to be below the predetermined threshold, a dotted line 555 is created. Advantageously, this helps to adaptively predict a road segment attribute e.g. turn restrictions, speed limits based on a graph indicative of a relationship between a road segment and a detection and more accurately inform the driver on when to respond appropriately to a road segment detection.

[0055] Fig 5c shows a diagram of how a road network including how each individual detection attends to direction of each road segment according to an embodiment of the method in Fig. 2. In 580 of Fig. 5C, the directions are considered. In an example, there may be two different nodes relating to the same road segment but for different directions (e.g 582A shown as 2a and 582B shown as 2b).

[0056] Fig. 6 shows a graph according to an embodiment of the method in Fig. 2. As shown in 600 of Fig. 6, in an example, the structure is obtained as a Graph Neural Network model, in which each node has its own computational graph based on the neighbouring nodes.

[0057] For each node 602A shown as 1A, it relates to a road segment's direction and there is a feature vector (or road segment vector) X encoding data like its type (road segment Vs detection). Alternatively, or additionally, the feature vector is encoded in the same feature vector road segment direction, road classes, number of lanes, segment length, etc

[0058] For each edge between a detection node and a road-segment node, a feature vector is encoded, including at least one of the following data: detection type, minimum distance between detection and road-segment, distance from camera to detection, detection's confidence, detection's geopositioning confidence, etc;

[0059] In an embodiment, the encoding may be done using a feature vector of the same size, each feature type having its own distinct position in the vector space. One position may be preserved for encoding the type (road segment or detection's edge). It is to be appreciated that not all edges and nodes have associated feature vectors.

[0060] Alternatively, or additionally, the model is trained in a supervised manner for node classification using GNN techniques defining computational graphs only for road-segments nodes. In an example of 600 shown in Fig 6, an example is shown using a GNN with a depth of 2 layers. Arrows 622A and 622B indicate how the message (node/edge embedding) is propagated or/and transformed. Rectangles 632A, 632B and 632C indicate that messages are aggregated and a nonlinear function is applied on the aggregated result. For layer 0, input

vector x embeds a node or an edge is the input vector x_i . Similarly, Layer k which embeds collects information from nodes and edges that are k hops away.

Structural Context of the graph generating server 108

[081] Figs. 7 and 8 depict the graph generating server 108, upon which the transaction management system described above can be practiced.

[082] As seen in Fig. 7, the graph generating server 108 includes: a computer module 201; input devices such as a keyboard 202, a mouse pointer device 203, a scanner 226, a camera 227, and a microphone 280; and output devices including a printer 215, a display device 214 and loudspeakers 217. An external Modulator-Demodulator (Modem) transceiver device 216 may be used by the computer module 201 for communicating to and from a communications network 220 via a connection 221. The communications network 220 may be a wide-area network (WAN), such as the Internet, a cellular telecommunications network, or a private WAN. Where the connection 221 is a telephone line, the modem 216 may be a traditional “dial-up” modem. Alternatively, where the connection 221 is a high capacity (e.g., cable) connection, the modem 216 may be a broadband modem. A wireless modem may also be used for wireless connection to the communications network 220.

[083] The input and output devices may be used by an operator who is interacting with the graph generating server 108. For example, the printer 215 may be used to print reports relating to the status of the graph generating server 108.

[084] The graph generating server 108 uses the communications network 220 to communicate with the requestor devices 102 to receive commands and data. The graph generating server 108 also uses the communications network 220 to communicate with the requestor devices 102 to send notification messages or broadcast transaction records.

[085] The computer module 201 typically includes at least one processor unit 205, and at least one memory unit 206. For example, the memory unit 206 may have semiconductor random access memory (RAM) and semiconductor read only memory (ROM). The computer module 201 also includes a number of input/output (I/O) interfaces including: an audio-video interface 207 that couples to the video display 214, loudspeakers 217 and microphone 280; an I/O interface 213 that couples to the keyboard 202, mouse 203, scanner 226, camera 227 and optionally a joystick or other human interface device (not illustrated); and an interface 208 for the external modem 216 and printer 215. In some implementations, the modem 216 may be

incorporated within the computer module 201, for example within the interface 208. The computer module 201 also has a local network interface 211, which permits coupling of the computer system 110 via a connection 223 to a local-area communications network 222, known as a Local Area Network (LAN). As illustrated in Fig. 7, the local communications network 222 may also couple to the wide network 220 via a connection 224, which would typically include a so-called "firewall" device or device of similar functionality. The local network interface 211 may comprise an Ethernet circuit card, a Bluetooth® wireless arrangement or an IEEE 802.11 wireless arrangement; however, numerous other types of interfaces may be practiced for the interface 211.

[086] The I/O interfaces 208 and 213 may afford either or both of serial and parallel connectivity, the former typically being implemented according to the Universal Serial Bus (USB) standards and having corresponding USB connectors (not illustrated). Storage devices 209 are provided and typically include a hard disk drive (HDD) 210. Other storage devices such as a floppy disk drive and a magnetic tape drive (not illustrated) may also be used. An optical disk drive 212 is typically provided to act as a non-volatile source of data. Portable memory devices, such optical disks (e.g., CD-ROM, DVD, Blu-ray Disc™), USB-RAM, portable, external hard drives, and floppy disks, for example, may be used as appropriate sources of data to the graph generating server 108.

[087] The components 205 to 213 of the computer module 201 typically communicate via an interconnected bus 204 and in a manner that results in a conventional mode of operation of a computer system known to those in the relevant art. For example, the processor 205 is coupled to the system bus 204 using a connection 218. Likewise, the memory 206 and optical disk drive 212 are coupled to the system bus 204 by connections 219. Examples of computers on which the described arrangements can be practised include IBM-PC's and compatibles, Sun Sparcstations, Apple Mac™ or like computer systems.

[088] The methods of operating the transaction processing server 110, as shown in the processes of Figs. 2, 3, 4, and 6 to be described, may be implemented as one or more software application programs 233 executable within the graph generating server 108. In particular, the steps of the methods shown in Figs. 2, 3, 4, and 6 are effected by instructions 231 (see Fig. 8) in the software (e.g., computer program codes) 233 that are carried out within the transaction processing server 110. The software instructions 231 may be formed as one or more code modules, each for performing one or more particular tasks. The software may also be divided into two separate parts, in which a first part and the corresponding code modules performs the operation of the graph generating server 108 and a second part and the corresponding code

modules manages the API and corresponding user interfaces in the requestor devices 102, and on the display 214. In other words, the second part of the software manages the interaction between (a) the first part and (b) any one of the requestor devices 102, and the operator of the server 108.

[089] The software may be stored in a computer readable medium, including the storage devices described below, for example. The software is loaded into the graph generating server 108 from the computer readable medium, and then executed by graph generating server 108. A computer readable medium having such software or computer program recorded on the computer readable medium is a computer program product.

[090] The software (e.g., computer program codes) 233 is typically stored in the HDD 210 or the memory 206. The software 233 is loaded into the transaction processing server 110 from a computer readable medium (e.g., the memory 206), and executed by the processor 205. Thus, for example, the software 233 may be stored on an optically readable disk storage medium (e.g., CD-ROM) 225 that is read by the optical disk drive 212. A computer readable medium having such software or computer program recorded on it is a computer program product. The use of the computer program product in the server 108 preferably effects an apparatus for processing transaction requests for functioning as a transaction management system.

[091] In some instances, the application programs 233 may be supplied to the user encoded on one or more CD-ROMs 225 and read via the corresponding drive 212, or alternatively may be read by the user from the networks 220 or 222. Still further, the software can also be loaded into the transaction processing server 110 from other computer readable media. Computer readable storage media refers to any non-transitory tangible storage medium that provides recorded instructions and/or data to the graph generating server 108 for execution and/or processing by the processor 205. Examples of such storage media include floppy disks, magnetic tape, CD-ROM, DVD, Blu-ray™ Disc, a hard disk drive, a ROM or integrated circuit, USB memory, a magneto-optical disk, or a computer readable card such as a PCMCIA card and the like, whether or not such devices are internal or external of the computer module 201. Examples of transitory or non-tangible computer readable transmission media that may also participate in the provision of software, application programs, instructions and/or data to the computer module 201 include radio or infra-red transmission channels as well as a network connection to another computer or networked device, and the Internet or Intranets including e-mail transmissions and information recorded on Websites and the like.

[092] The second part of the application programs 233 and the corresponding code modules mentioned above may be executed to implement one or more API of the transaction processing server 110 with associated graphical user interfaces (GUIs) to be rendered or otherwise represented upon the display 214 or the display of the merchant device 120 and the user device 130. Through manipulation of typically the keyboard 202 and the mouse 203, an operator of the server 110 and the application may manipulate the interface in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Similarly, on the merchant devices 120 and the user devices 130, a user of those devices 120 and 130 manipulate the input devices (e.g., touch screen, keyboard, mouse, etc.) of those devices 120, 130 in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Other forms of functionally adaptable user interfaces may also be implemented, such as an audio interface utilizing speech prompts output via the loudspeakers 217 and user voice commands input via the microphone 280. These other forms of functionally adaptable user interfaces may also be implemented on the devices 102.

[093] Fig. 8 is a detailed schematic block diagram of the processor 205 and a “memory” 234. The memory 234 represents a logical aggregation of all the memory modules (including the HDD 209 and semiconductor memory 206) that can be accessed by the computer module 201 in Fig. 7.

[094] When the computer module 201 is initially powered up, a power-on self-test (POST) program 250 executes. The POST program 250 is typically stored in a ROM 249 of the semiconductor memory 206 of Fig. 7. A hardware device such as the ROM 249 storing software is sometimes referred to as firmware. The POST program 250 examines hardware within the computer module 201 to ensure proper functioning and typically checks the processor 205, the memory 234 (209, 206), and a basic input-output systems software (BIOS) module 251, also typically stored in the ROM 249, for correct operation. Once the POST program 250 has run successfully, the BIOS 251 activates the hard disk drive 210 of Fig. 7. Activation of the hard disk drive 210 causes a bootstrap loader program 252 that is resident on the hard disk drive 210 to execute via the processor 205. This loads an operating system 253 into the RAM memory 206, upon which the operating system 253 commences operation. The operating system 253 is a system level application, executable by the processor 205, to fulfil various high level functions, including processor management, memory management, device management, storage management, software application interface, and generic user interface.

[095] The operating system 253 manages the memory 234 (209, 206) to ensure that each process or application running on the computer module 201 has sufficient memory in which to

execute without colliding with memory allocated to another process. Furthermore, the different types of memory available in the server 108 of Fig. 7 must be used properly so that each process can run effectively. Accordingly, the aggregated memory 234 is not intended to illustrate how particular segments of memory are allocated (unless otherwise stated), but rather to provide a general view of the memory accessible by the server 108 and how such is used.

[096] As shown in Fig. 8, the processor 205 includes a number of functional modules including a control unit 239, an arithmetic logic unit (ALU) 240, and a local or internal memory 248, sometimes called a cache memory. The cache memory 248 typically includes a number of storage registers 244 - 246 in a register section. One or more internal busses 241 functionally interconnect these functional modules. The processor 205 typically also has one or more interfaces 242 for communicating with external devices via the system bus 204, using a connection 218. The memory 234 is coupled to the bus 204 using a connection 219.

[097] The application program 233 includes a sequence of instructions 231 that may include conditional branch and loop instructions. The program 233 may also include data 232 which is used in execution of the program 233. The instructions 231 and the data 232 are stored in memory locations 228, 229, 230 and 235, 236, 237, respectively. Depending upon the relative size of the instructions 231 and the memory locations 228-230, a particular instruction may be stored in a single memory location as depicted by the instruction shown in the memory location 230. Alternately, an instruction may be segmented into a number of parts each of which is stored in a separate memory location, as depicted by the instruction segments shown in the memory locations 228 and 229.

[098] In general, the processor 205 is given a set of instructions which are executed therein. The processor 205 waits for a subsequent input, to which the processor 205 reacts to by executing another set of instructions. Each input may be provided from one or more of a number of sources, including data generated by one or more of the input devices 202, 203, data received from an external source across one of the networks 220, 202, data retrieved from one of the storage devices 206, 209 or data retrieved from a storage medium 225 inserted into the corresponding reader 212, all depicted in Fig. 7. The execution of a set of the instructions may in some cases result in output of data. Execution may also involve storing data or variables to the memory 234.

[099] The disclosed transaction management arrangements use input variables 254, which are stored in the memory 234 in corresponding memory locations 255, 256, 257. The transaction management arrangements produce output variables 261, which are stored in the

memory 234 in corresponding memory locations 262, 263, 264. Intermediate variables 258 may be stored in memory locations 259, 260, 266 and 267.

[0100] Referring to the processor 205 of Fig. 8, the registers 244, 245, 246, the arithmetic logic unit (ALU) 240, and the control unit 239 work together to perform sequences of micro-operations needed to perform “fetch, decode, and execute” cycles for every instruction in the instruction set making up the program 233. Each fetch, decode, and execute cycle comprises:

[0101] a fetch operation, which fetches or reads an instruction 231 from a memory location 228, 229, 230;

[0102] a decode operation in which the control unit 239 determines which instruction has been fetched; and

[0103] an execute operation in which the control unit 239 and/or the ALU 240 execute the instruction.

[0104] Thereafter, a further fetch, decode, and execute cycle for the next instruction may be executed. Similarly, a store cycle may be performed by which the control unit 239 stores or writes a value to a memory location 232.

[0105] Each step or sub-process in the processes of Figs. 2, 3, 4, and 6 is associated with one or more segments of the program 233 and is performed by the register section 244, 245, 247, the ALU 240, and the control unit 239 in the processor 205 working together to perform the fetch, decode, and execute cycles for every instruction in the instruction set for the noted segments of the program 233.

[0106] It is to be understood that the structural context of the graph generating server 108 is presented merely by way of example. Therefore, in some arrangements, one or more features of the graph generating server 108 may be omitted. Also, in some arrangements, one or more features of the graph generating server 108 may be combined together. Additionally, in some arrangements, one or more features of the graph generating server 108 may be split into one or more component parts.

[0107] Fig. 9 shows an alternative implementation of the graph generating server 108. In the alternative implementation, the graph generating server 108 may be generally described as a physical device comprising at least one processor 902 and at least one memory 904 including

computer program code. The at least one memory 904 and the computer program code are configured to, with the at least one processor 902, cause the graph generating server 108 to perform the operations described in Figs. 2, 3, 4, and 6. The graph generating server 108 may also include a transaction processing module 906, payment monitoring module 908, a registered user module 910, a registered merchant module 912, and credit risk limit module 914. The memory 904 stores computer program code that the processor 902 compiles to have each of the transaction processing module 906, the payment monitoring module 908, the registered user detail module 910, the registered merchant detail module 912, and the credit risk limit module 912 performs their respective functions. It will be appreciated that the processor 902 may also be configured to perform the functions performed by each of the transaction processing module 906, the payment monitoring module 908, the registered user detail module 910, the registered merchant detail module 912, and the credit risk limit module 912. In this arrangement, the transaction processing server 110 may only have a single processor 902 for performing the above-mentioned functions.

[0108] With reference to the discussion above regarding the devices 102, the registered user module 910 manages the on-boarding (see the on-boarding discussion above) and storing of users who are consumers who wish to buy products from registered merchants. With reference to the discussion above regarding the devices 102, the registered merchant module 912 manages the on-boarding (see the on-boarding discussion above) and storing of merchants which offer products for sale.

[0109] The arrangements described are applicable to the computer and data processing industries and particularly for the payment technology.

[0110] The foregoing describes only some embodiments of the present invention, and modifications and/or changes can be made thereto without departing from the scope and spirit of the invention, the embodiments being illustrative and not restrictive.

CLAIMS:

1. A computer-implemented method for adaptively predicting a road segment attribute based on a a graph that is indicative of a relationship between a road segment and a detection, the method comprising:
 - identifying the road segment and its corresponding feature;
 - identifying the detection and its corresponding feature, the feature being one of detection confidence, geo-positioning confidence, distance from the camera; and
 - adaptively providing a graph on a relationship between the road segment and the detection based on the respective features of the road segment and the detection so as to predict the road segment attribute.
2. The method according to claim 1, further comprising:
 - determining the relationship between the road segment and the detection, the corresponding feature of the road segment being one of length, road class, road width, the relationship representing at least one of a minimum distance, a maximum distance from the detection to the road segment.
3. The method according to claim 1, wherein the step of identifying the respective feature of the road segment, including identifying a direction of the road segment and a further road segment.
4. The method according to claim 2, further comprising:
 - representing each detection and each road segment as a detection node and a road segment node, respectively.
5. The method according to claim 4, further comprising:
 - determining a distance between the detection node and the road segment node; and
 - providing a link between the detection node and the road segment node when the distance between the detection node and the road segment node is below a threshold value.

6. The method according to claim 5, further comprising:
identifying an edge between each road segment node and each detection node.
7. The method according to claim 6, further comprising:
encoding a feature vector for the detection, the feature vector comprising at least one of a detection type, minimum distance between detection and road segment, distance from camera to detection, detection confidence, detection geo-positioning confidence.
8. The method according to any one of the preceding claims, further comprising:
encoding a road segment vector for the road segment node, the road segment vector comprising at least one of a road segment direction, road class, a number of lanes, segment length.
9. A server for adaptively predicting a road segment attribute based on a sgraph that is indicative of a relationship between a road segment and a detection, the server comprising: at least one processor; and at least one memory including computer program code; the at least one memory and the computer program code configured to, with the at least one processor, cause the server at least to:

identify the road segment and its corresponding feature,

identify the detection and its corresponding feature, the feature being one of detection confidence, geo-positioning confidence, distance from the camera; and

adaptively provide a graph on a relationship between the road segment and the detection based on the respective features of the road segment and the detection so as to predict the road segment attribute.
10. The server according to claim 9, wherein the at least one memory and the computer program code is further configured with the at least one processor to determine the relationship between the road segment and the detection, the corresponding feature of the road segment being one of length, road class, road width, the relationship representing at least one of a minimum distance, a maximum distance from the detection to the road segment.
11. The server according to claim 9, wherein the at least one memory and the computer program code is further configured with the at least one processor to:

identify the respective feature of the road segment, including identifying a direction of the road segment and a further road segment.

12. The server according to claim 10, wherein the at least one memory and the computer program code is further configured with the at least one processor to:

represent each detection and each road segment as a detection node and a road segment node, respectively.

13. The server according to claim 12, wherein the at least one memory and the computer program code is further configured with the at least one processor to determine a distance between the detection node and the road segment node; and

provide a link between the detection node and the road segment node when the distance between the detection node and the road segment node is below a threshold value.

14. The server according to any one of claims 13, wherein the at least one memory and the computer program code is further configured with the at least one processor to: identify an edge between each road segment node and each detection node.

15. The server according to any one of claims 14, wherein the at least one memory and the computer program code is further configured with the at least one processor to: encode a feature vector for the detection, the feature vector comprising at least one of a detection type, minimum distance between detection and road-segment, distance from camera to detection, detection confidence, detection geopositioning confidence.

16. The server according to any one of claims 9-15, wherein the at least one memory and the computer program code is further configured with the at least one processor

to: encode a road segment vector for the road segment node, the road segment vector comprising at least one of a road segment direction, road class, a number of lanes, segment length.