BRIEF DESCRIPTION OF THE DRAWINGS

[0007] The accompanying drawings, where like reference numerals refer to identical or functionally similar elements throughout the separate views, together with the detailed description below, are incorporated in and form part of the specification, and serve to further illustrate embodiments of concepts that include the claimed disclosure, and explain various principles and advantages of those embodiments.

[0008] The methods and systems disclosed herein have been represented where appropriate by conventional symbols in the drawings, showing only those specific details that are pertinent to understanding the embodiments of the present disclosure so as not to obscure the disclosure with details that will be readily apparent to those of ordinary skill in the art having the benefit of the description herein.

[0009] **FIG. 1** illustrates a diagram of an exemplary system configured to provide comprehensive injury risk assessment in a 3D environment, in accordance with various exemplary embodiments.

[0010] **FIG. 2** is a schematic of the simulation process implemented to build the 3D environment by an agent using Reinforced Learning (RL).

[0011] **FIG. 3** illustrates an exemplary method for providing comprehensive injury risk assessment in a 3D environment, in accordance with various exemplary embodiments.

[0012] **FIGS. 4a – 4c** illustrates the agent using RL in the simulation building process by moving throughout an environment.

[0013] **FIGS. 5a – 5c** have the associated risk outputs represented in the environment for low-to-no risk by check marks (✓) and for medium-to-high risks by cross marks (✗).

[0014] **FIG. 6** illustrates a diagrammatic representation of an exemplary machine in the form of a computing system within which a set of instructions for causing the

accident prevention, all possible risks can be detected without exposing any people to the risks, and the aging population will be able to live independently for a longer period of time.

[0020] **FIG. 1** illustrates an environment 100 within which systems and methods for identifying environmental risks in a simulated environment using machine learning such as reinforced learning and deep learning can be implemented, in accordance with some embodiments. The environment 100 may include an agent 110, a communications network 130 (e.g., the Internet or a computing cloud), a system for identifying environmental risks 140 (also referred to herein as system 140), user(s) 170, user device(s) 180, and optionally a database 150. The agent 110, the system 140, user device(s) 180, and the database 150 may be connected via the communications network 130.

[0021] The agent 110 may include or be coupled to one or more cameras 120, one or more onboard depth sensors, wheels, one or more electric motors, power supply, additional integrated devices, and wireless connection (i.e., Bluetooth, Wi-Fi, etc.). In some embodiments, agent 110 may be deployed on the floor of a space, and agent 110 travels around the space collecting data about items positioned within the space in order for system 140 to create a 3D map of the space. In another embodiment, agent 110 may be manually controlled by user device 180 operated by user 170 to collect data about objects positioned around the space. User device 180 may include a mobile telephone, a personal computer (PC), a laptop, a smartphone, a tablet PC, and so forth. In some embodiments, agent 110 may be replaced with, or take the form of, any suitable computing device, such as those described with respect to FIG. 6.

[0022] System 140 may be cloud-based, or comprise one or more remote servers. Through an assessment using system 140, a 3D model risk assessment 160 is provided to user 170 through user device 180 that may include a 3D map depicting areas of injury risk as depicted in the exemplary embodiments of FIGS. 5a – 5c.

using RL‡ to continue object detection. In **FIG. 2**, the RL that is implemented during the simulation process is represented in the schematic by boxes outside of the "simulation process" box. The dashed lines represent input and the solid lines represent output. As the agent moves around the environment detecting objects, as shown **FIGS. 4a – 4c**, it builds the 3D map and establishes the potential risk for accidents by mapping objects in the home, as shown in **FIGS. 5a – 5c** pathways, their relative locations, and simulating the user in that environment, respectively.

[0026] The agent determines an action based on its current state by employing a policy, π, that depends on an observation, made by the agent, of the state. It is a distribution over actions for each state, $\pi(a|s) = \underline{P}\,[A_i = a\,,\, S_i = s]$. For example, in **FIG. 2**, initial sequence, $i = 0$, begins with $S_{i+1=1}$ (environmental state sensed by agent). In this case for $i = 0$, the policy may depend on state s being an observation of more pieces of furniture, for example, $\pi(s)$ = if cannot determine if input is one or multiple objects, a = then the agent drives closer and gets a better look.

[0027] The agent now has input of this first state, S_1, and due to its policy it takes the action, a, arrives at $i = 1$ and gets feedback from reward function, R (s, a). It is a function of state-action pairs; $R_s^a = \underline{E}\,([R_{i+1} \mid S_i = s,\, A_i = a]$; where s, $a \in \mathbb{R}$ & $\underline{E}$ is expectation. Then, transition probability function is probability distribution over the next possible state given the current state, which yields next likely state, s_{i+1}., is $P_{ss'}{}^a = \underline{P}[\, S_{i+1} = s' \mid S_i = s,\, A_i = a]$; where S_i = particular state, s = current state, and $S_{i+1} = s'$ is setting the next sequential state to s', which is the predicted next state. The agent learns a Q-value based on the action taken from another policy, which is updated by calculating it according to a new Q-value function (i.e. update the policy function according to Q-value function). The agent now has a refined understanding of S_1 and input for S_2. In other words:

[0028] S_1 = input of state, then policy and action, $\pi\,(s) = a$ implemented and performed. Reward function received, R (s, a), transition probability over next

$\gamma \max_{a'} Q(s', a')] \rightarrow Q(s', a)$; Where s' and a' are future state and action (to use to update functions). Notice upcoming action a' is tuned to maximize next Q-value function, this is how it keeps the Q-value function updated with updated policy (i.e. instead of using current policy).

[0034] **FIG. 3** illustrates an exemplary method (and summary) for providing assessment of injury risk in a 3D environment, in accordance with various exemplary embodiments.

[0035] The method 300 includes:

[0036] At step 301, deploying an agent into an environment. For example, see **FIG. 2.**

[0037] At step 302, detecting a first object at a first state by the agent. For example, see **FIG. 2.**

[0038] At step 303, taking an action to interact with the first object using reinforced learning by the agent, the action taken in order to maximize a reward. For example, see **FIG. 2.**

[0039] At step 304, mapping the first object to a three-dimensional environment. For example, see **FIGS. 4a – 4c.**

[0040] At step 305 identifying potential risks within the three-dimensional environment. For example, see **FIGS. 5a – 5c.**

[0041] **FIGS. 4a - 4c** illustrates the agent using RL in the simulation building process by moving throughout the environment. In this image the agent starts on the left by the front door **[FIG. 4a]**, goes towards the couch **[FIG. 4b]**, then between the tables **[FIG. 4c]**, and onward throughout the home.

[0042] Refering to **FIG. 5a** for the full path, after being between the tables (depicted by **FIG. 4c**), the agent goes through the hall into the bedroom, then around the bedroom, and back through the hall into the kitchen, and ends where it started by the front door.

[0043] As it is moving and interacting, it improves the 3D model with each sequence. The solid (dotted) lines represent a finished (unfinished) model of an object. In this figure the agent recognizes chairs (white lines) and tables (black lines) as objects for illustrative purposes – it is not limited to chairs and tables.

[0044] Consider the starting point with dotted white lines to the chair and couch as the initial sequence (*i=0*). These objects are input of the state $S_{i+1} = S_{0+1} = S_1$ with R_1 and A_1, reward and action, to start the simulation process. At the next sequential point (*i=1*) the agent has interacted with these objects, detected them to be included in the next environmental state S_2, suitably mapped them in the environment (represented by the solid white lines), and identified a potential risk (**FIGS. 5a – 5c**). Note the dotted lines also present in this sequence (*i=1*) and how they solidify by the next sequence (*i=2*) similarly.

[0045] **FIGS. 5a – 5c** have the associated risk outputs represented for low-to-no risk by check marks (✓) and for medium-to-high risks by cross marks (✗). In **FIG. 5b** it is evident that as each sequence progresses, the 3D map and accident risks in it become better defined, optimizing the capabilities of the method here for the 3D model simulation. In **FIG. 5b** the house fades out to depict how the certainty of the 3D map and simulation, with corresponding risks, are improved with updated functions with each sequence. Hence, in **FIG. 5c**, the whole house is solid because the agent has finished and ended back by the front door.

[0046] **FIG. 6** illustrates an exemplary computing system 600 that may be used to implement embodiments described herein. The computing system 600 of **FIG. 6** may include one or more processors 610 and memory 620. Memory 620 stores, in part, instructions and data for execution by the one or more processors 610. Memory 620 can store the executable code when the computing system 600 is in operation. The computing system 600 of **FIG. 6** may further include a mass storage 630, portable storage 640, one or more output devices 650, one or more input devices 660, a network interface 670, and one or more peripheral devices 680.