(12) **United States Patent**
Northcutt

(10) **Patent No.:** **US 11,752,637 B2**
(45) **Date of Patent:** *Sep. 12, 2023

(54) **OBJECT LOCATION ANALYSIS**

(71) Applicant: **TOYOTA RESEARCH INSTITUTE, INC.**, Los Altos, CA (US)

(72) Inventor: **Brandon Northcutt**, Arlington, MA (US)

(73) Assignee: **TOYOTA RESEARCH INSTITUTE, INC.**, Los Altos, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **17/734,906**

(22) Filed: **May 2, 2022**

(65) **Prior Publication Data**

US 2022/0262109 A1 Aug. 18, 2022

Related U.S. Application Data

(63) Continuation of application No. 16/582,716, filed on Sep. 25, 2019, now Pat. No. 11,348,332.

(51) **Int. Cl.**
G06V 20/10	(2022.01)
G06V 20/64	(2022.01)
G06V 10/764	(2022.01)
B25J 11/00	(2006.01)
B25J 9/16	(2006.01)
G06F 17/18	(2006.01)
B25J 9/00	(2006.01)

(52) **U.S. Cl.**
CPC *B25J 11/008* (2013.01); *B25J 9/0003* (2013.01); *B25J 9/1666* (2013.01); *G06F*

17/18 (2013.01); *G06V 10/764* (2022.01); *G06V 20/10* (2022.01); *G06V 20/653* (2022.01)

(58) **Field of Classification Search**
CPC G06K 9/00664; G06K 9/00214; B25J 9/0003; B25J 11/008; B25J 9/1666; G06V 10/764; G06V 20/10; G06V 20/653
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

10,289,910	B1 *	5/2019	Chen	G06N 5/046
10,937,178	B1 *	3/2021	Srinivasan	G06T 7/521
11,348,332	B2	5/2022	Northcutt	
2007/0192910	A1 *	8/2007	Vu	B25J 19/06
				901/17
2007/0269077	A1 *	11/2007	Neff	G05D 1/107
				382/103

(Continued)

Primary Examiner — Ming Y Hon
(74) *Attorney, Agent, or Firm* — SEYFARTH SHAW LLP

(57) **ABSTRACT**

A method for controlling a robotic device based on observed object locations includes obtaining a group of observations of an object type in an environment based on identifying one or more first objects associated with the object type over a period of time in the environment. The method also includes localizing the object type in the environment based on a location cluster comprising a group of nodes. Each node in the group of nodes associated with one observation of the group of observations. The method further includes generating a cost map indicating a probability distribution of the object type in relation to the localized object type in the environment. The method still further includes controlling the robotic device to perform an action in the environment based on the cost map and a map of the environment.

20 Claims, 8 Drawing Sheets
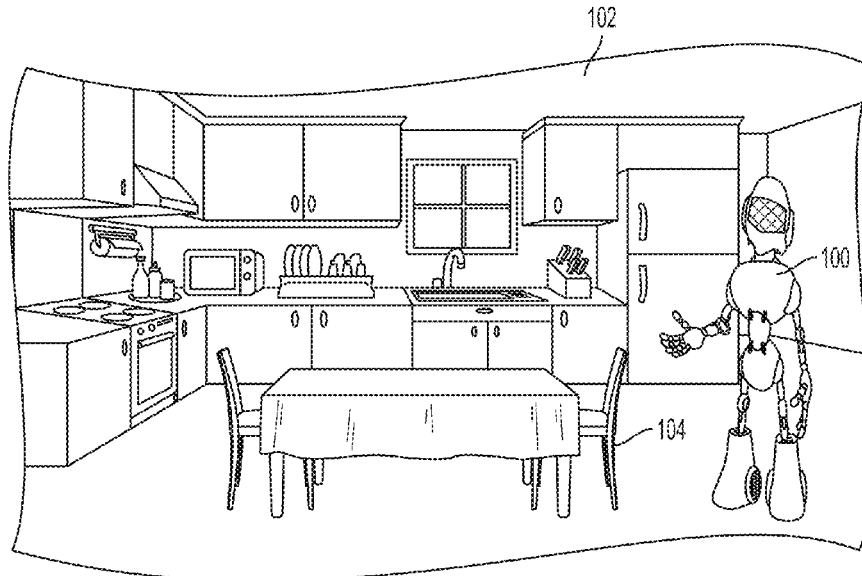
(56)   **References Cited**

U.S. PATENT DOCUMENTS

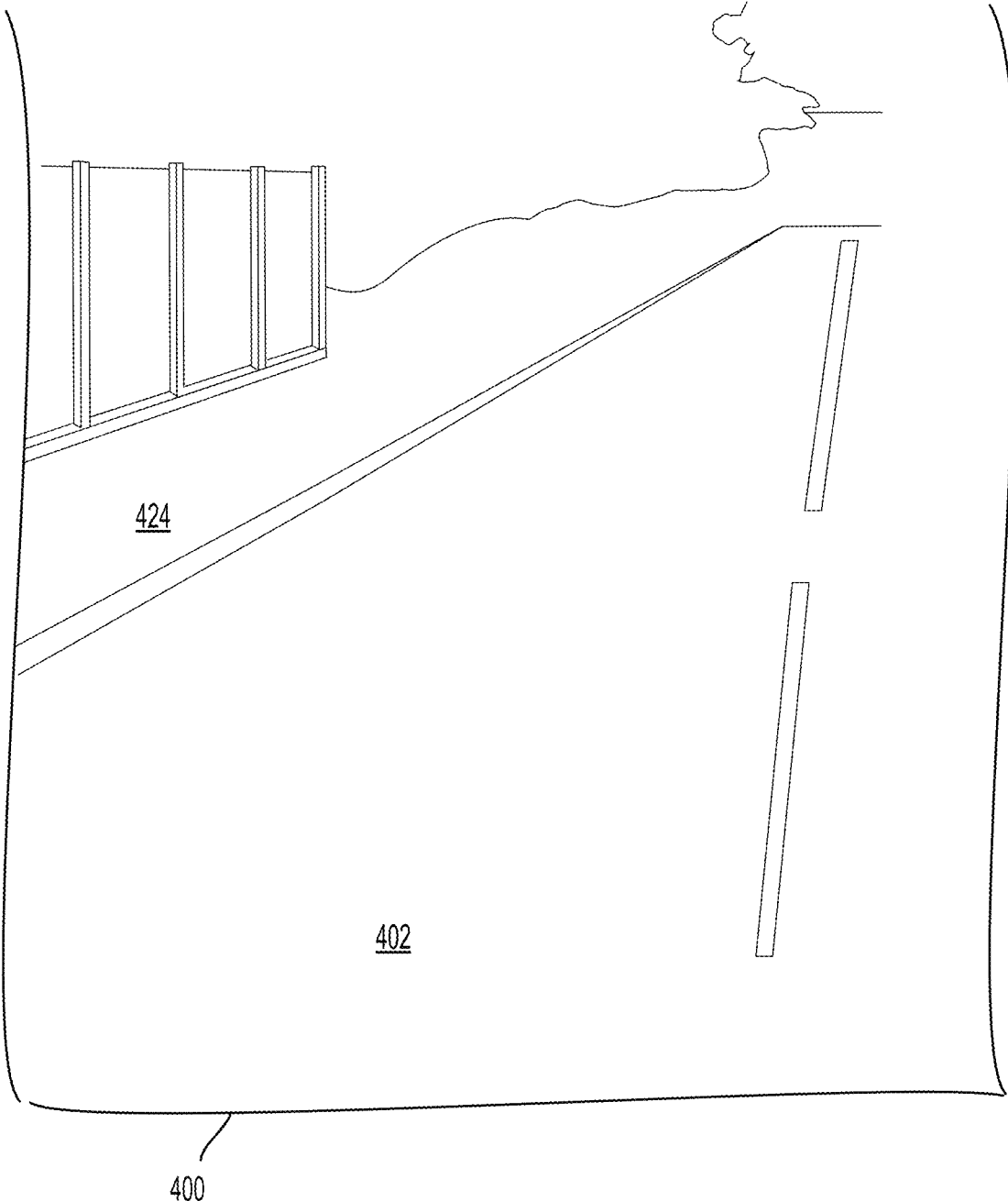|				
2012/0173018	A1*	7/2012	Allen	G05D 1/0248
				700/245
2014/0050455	A1*	2/2014	Ni	H04N 7/181
				386/E5.028
2014/0214255	A1*	7/2014	Dolgov	G05D 1/0274
				701/23
2014/0288705	A1*	9/2014	Yun	B62D 57/032
				700/246
2017/0121113	A1*	5/2017	Wagner	G05B 19/124
2017/0157769	A1*	6/2017	Aghamohammadi	
				B25J 9/1664
2018/0012370	A1*	1/2018	Aghamohammadi	G06T 7/50
2018/0107226	A1*	4/2018	Yang	G05D 1/042
2018/0326982	A1*	11/2018	Paris	B60W 30/18154
2019/0197861	A1*	6/2019	Tunnell	G08B 21/043
2019/0340775	A1*	11/2019	Lee	G05D 1/024
2020/0017317	A1*	1/2020	Yap	B65G 1/1376
2020/0018613	A1*	1/2020	Stenneth	G08G 1/048
2020/0053325	A1*	2/2020	Deyle	G05D 1/0088
2020/0086482	A1*	3/2020	Johnson	B25J 9/0009
2020/0174481	A1*	6/2020	Van Heukelom	G06N 3/044
2020/0207375	A1*	7/2020	Mehta	G06V 20/56
2020/0333142	A1*	10/2020	Wang	G05D 1/0221
2020/0410063	A1*	12/2020	O'Malley	G07C 5/02
2020/0410259	A1*	12/2020	Srinivasan	G06V 20/588
2021/0094538	A1*	4/2021	Beller	B60W 30/09
2021/0096571	A1*	4/2021	Modalavalasa	G05D 1/0027
2021/0139026	A1*	5/2021	Phan	G01S 13/867
2021/0347382	A1*	11/2021	Huang	B60W 60/0017
2021/0357377	A1*	11/2021	Stephens	G06F 16/221

* cited by examiner

FIG. 1

FIG. 2

FIG. 3

424

402

400

FIG. 4A

FIG. 4B

Map y cell length = 5 (cm)

Map x (m) length = 5 (cm)

452

FIG. 4C

FIG. 5

600

| OBSERVE OBJECTS IN AN ENVIRONMENT | 602 |

| GENERATE A PROBABILITY DISTRIBUTION FOR LOCATIONS OF THE OBSERVED OBJECTS | 604 |

| PERFORM AN ACTION IN THE ENVIRONMENT BASED ON THE GENERATED PROBABILITY DISTRIBUTION | 606 |
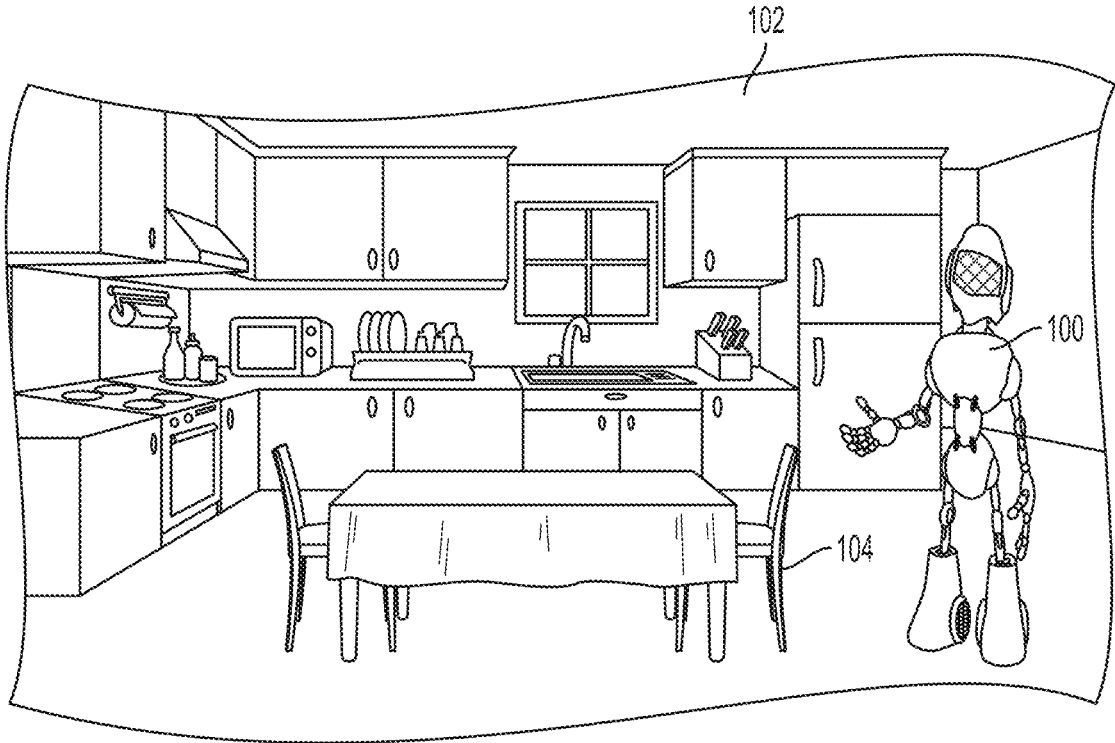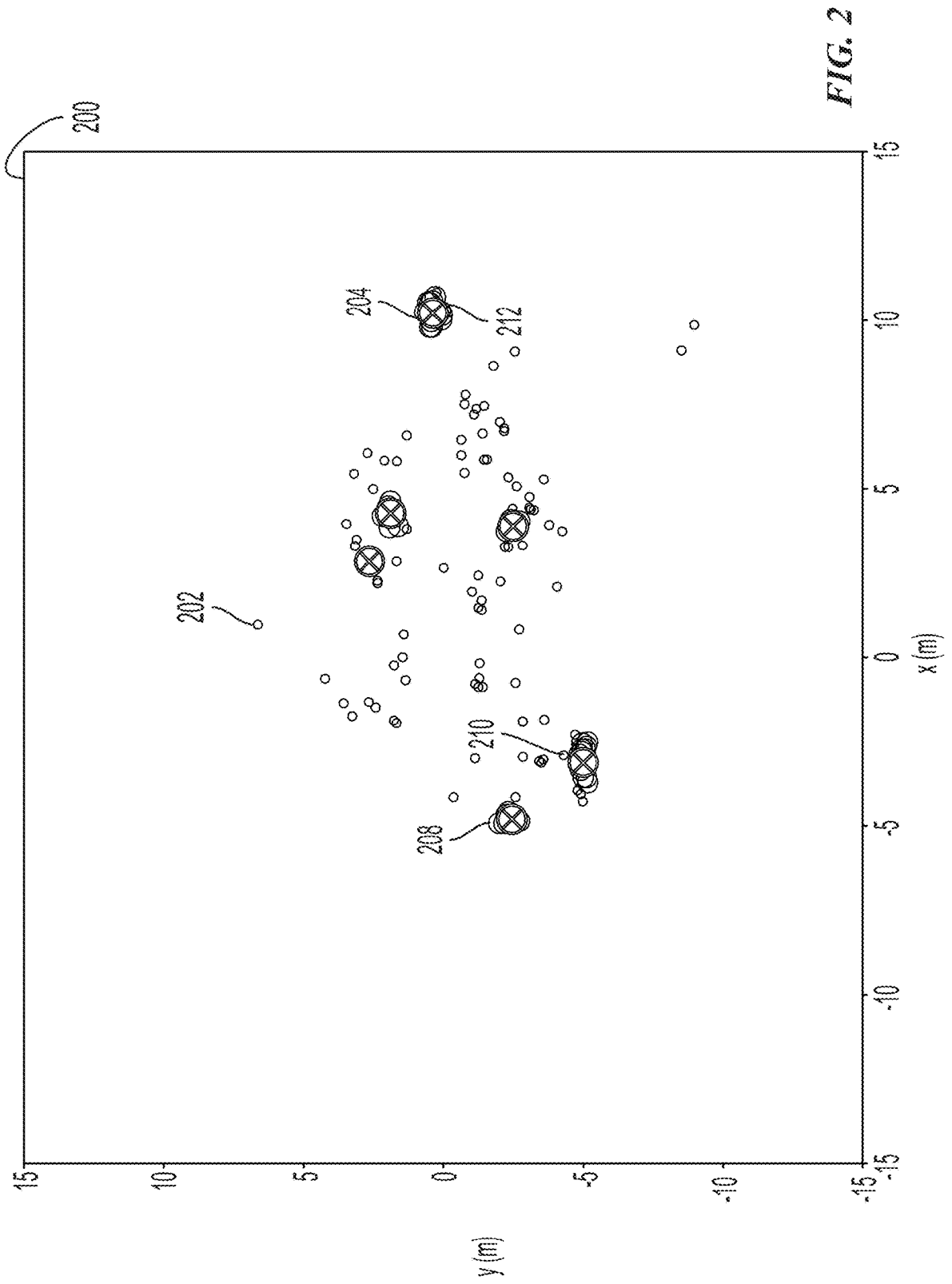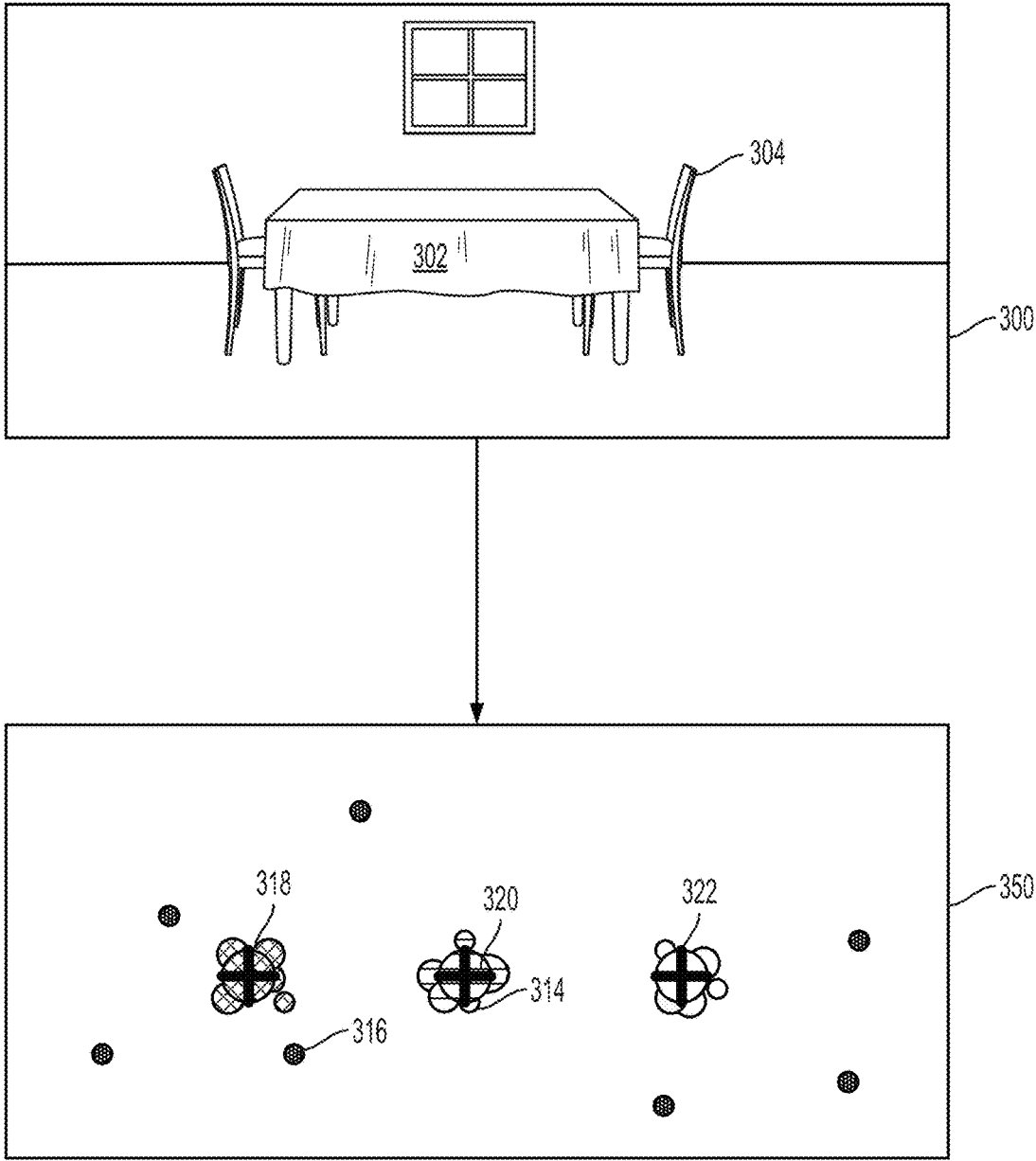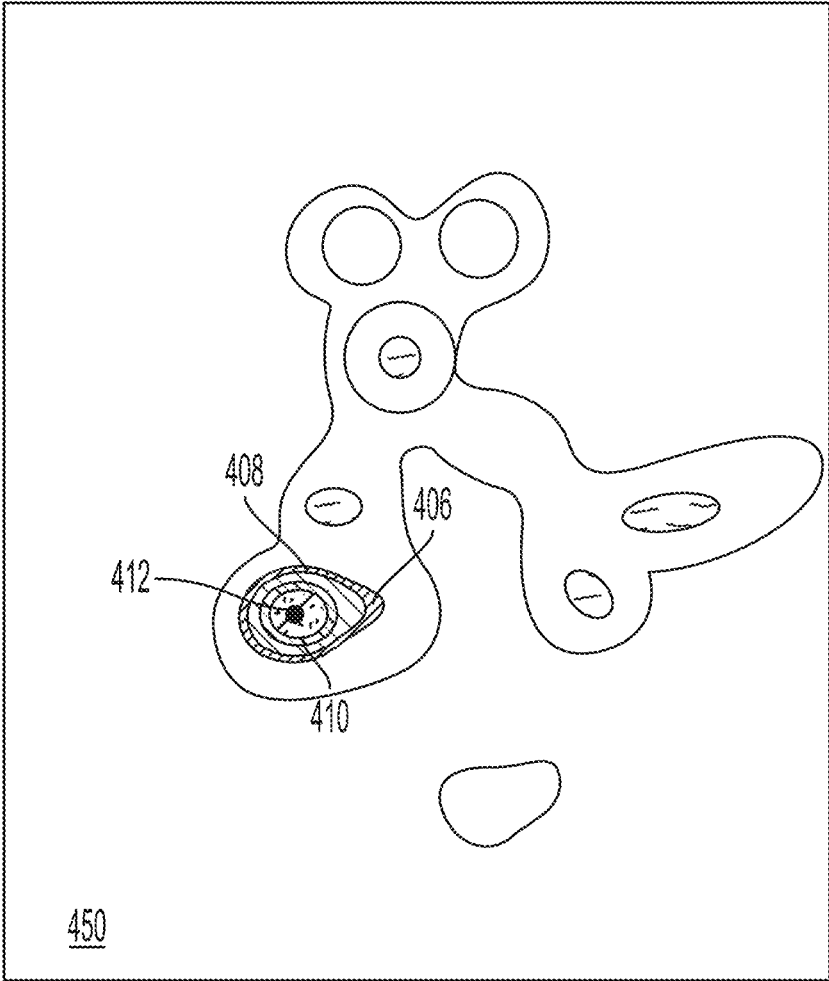
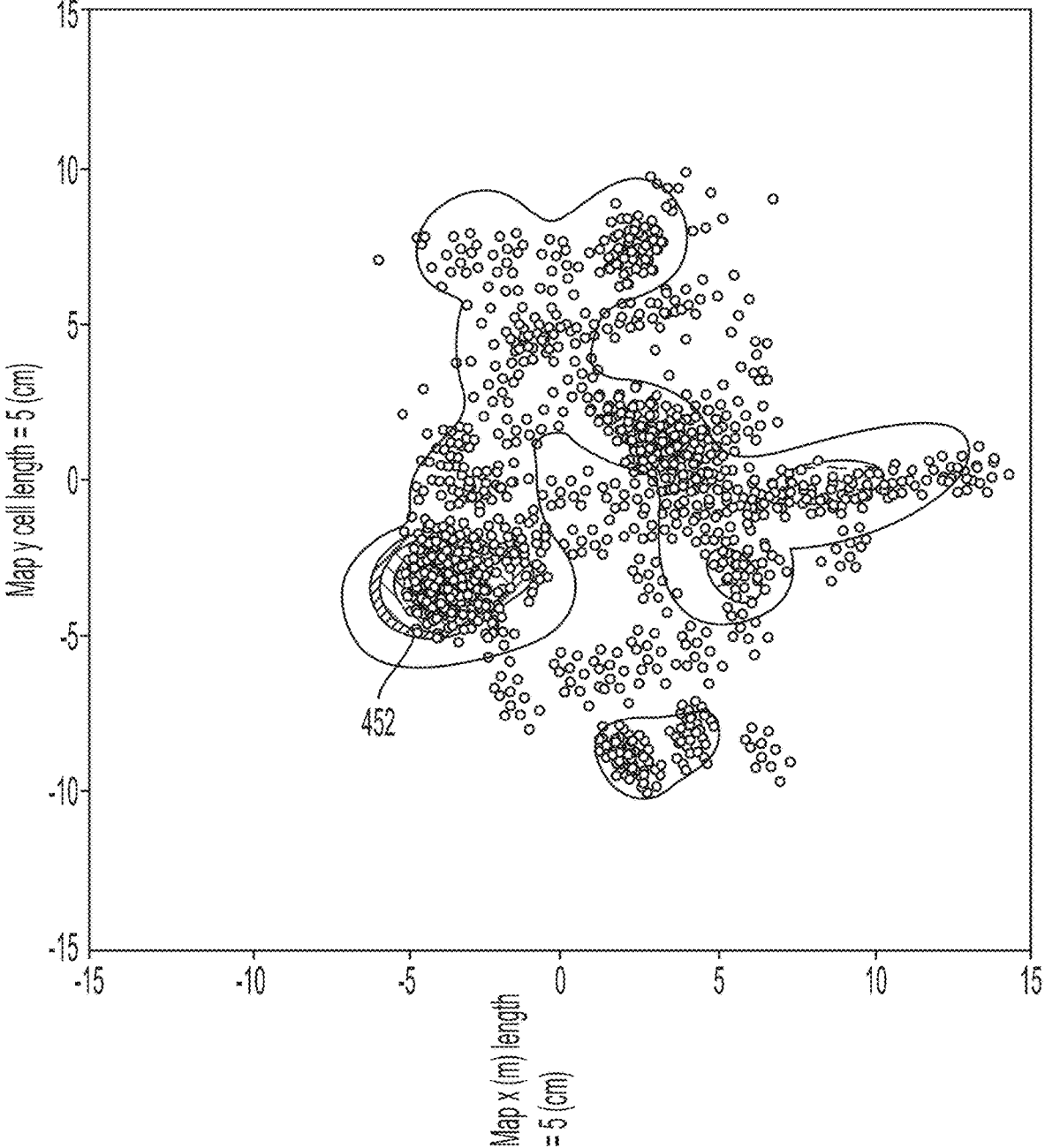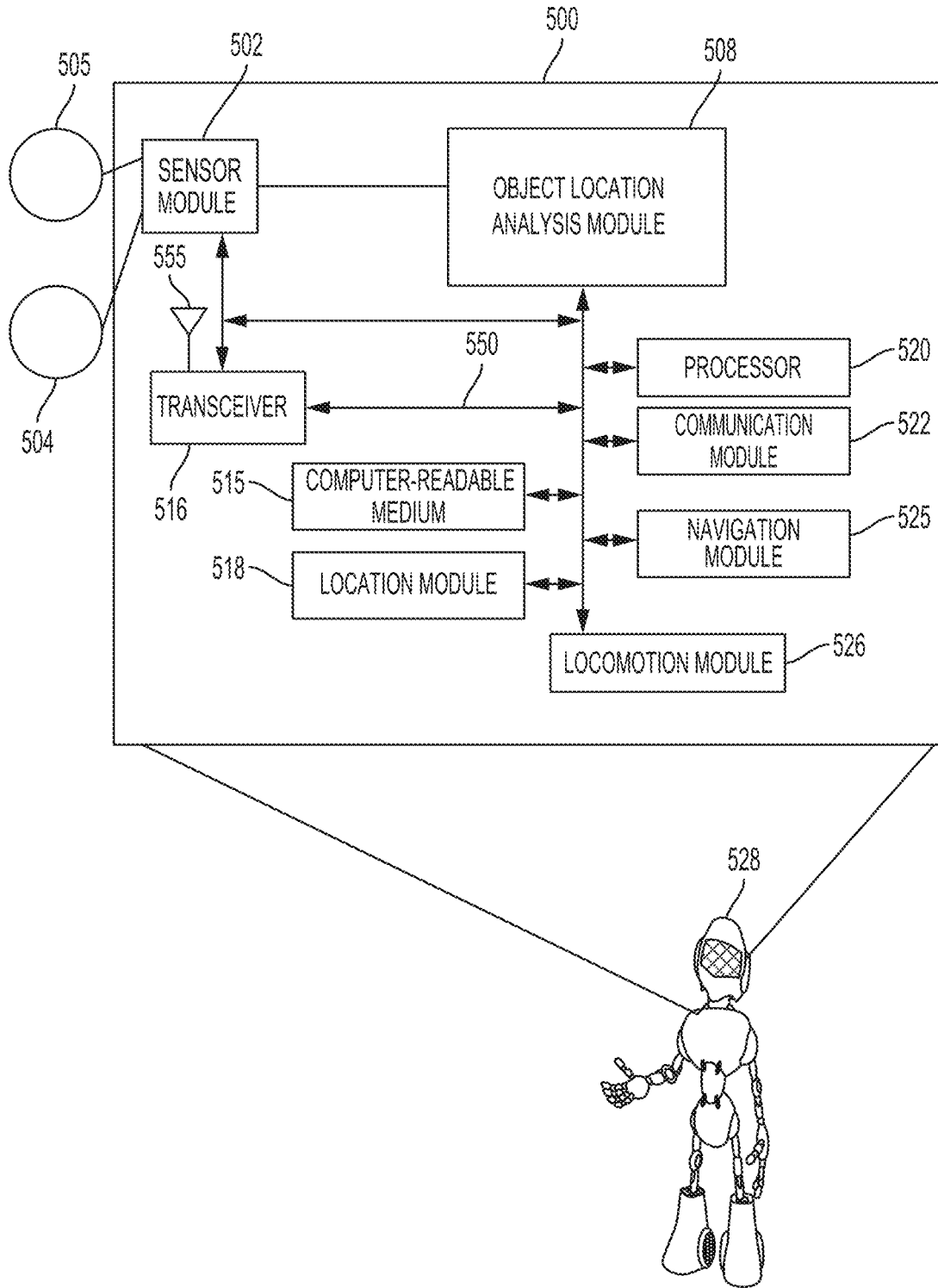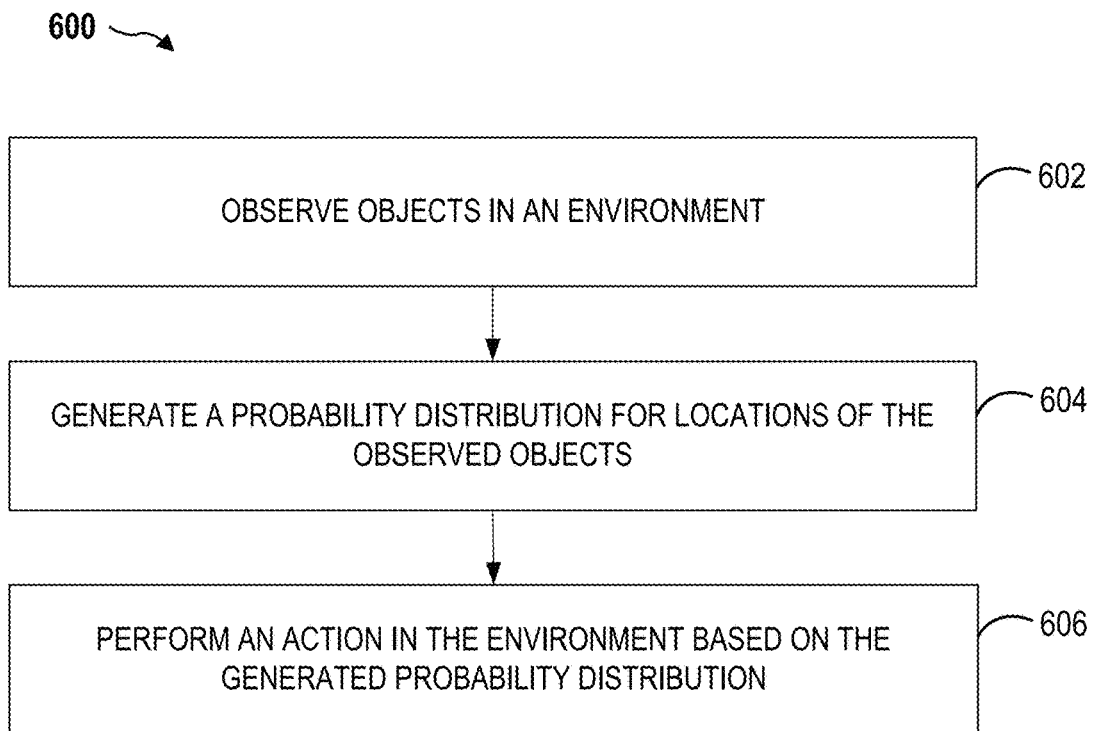*FIG. 6*

OBJECT LOCATION ANALYSIS

CROSS-REFERENCE TO RELATED APPLICATION

The present application is a continuation of U.S. patent application Ser. No. 16/582,716, filed on Sep. 25, 2019, and titled "OBJECT LOCATION ANALYSIS," the disclosure of which is expressly incorporated by reference in its entirety.

BACKGROUND

Field

Certain aspects of the present disclosure generally relate to object detection and, more particularly, to a system and method for detecting an object based on a spatial and temporal knowledge of objects in an environment.

Background

A robotic device may use one or more sensors, such as a camera, to identify objects in an environment. A location of each identified object may be estimated. Additionally, a label may be assigned to each localized object. In conventional object localization systems, each estimated object location may be stored in a database along with an estimation time stamp.

In most systems, images are generated at a number of frames per second, such as ten frames per second. Each frame may include multiple objects. The number of frames and the number of objects in each frame increase an amount of storage resources used by conventional object detection systems. Additionally, conventional object localization system may use an increased amount of processing resources for processing the images to localize the objects.

Conventional object detection systems may generate false positive and false negative location estimations. That is, for each estimated object location, some estimates may be valid, while others some may be invalid.

It is desirable to improve object detection systems to reduce an amount of used resources. It is also desirable to improve object detection systems to reduce a number of false positives and false negatives.

SUMMARY

In one aspect of the present disclosure, a method for controlling a robotic device based on observed object locations is disclosed. The method includes observing objects in an environment. The method also includes generating a probability distribution for locations of the observed objects. The method further includes controlling the robotic device to perform an action in the environment based on the generated probability distribution.

In another aspect of the present disclosure, a non-transitory computer-readable medium with non-transitory program code recorded thereon is disclosed. The program code is for controlling a robotic device based observed object locations. The program code is executed by a processor and includes program code to observe objects in an environment. The program code also includes program code to generate a probability distribution for locations of the observed objects. The program code further includes program code to control the robotic device to perform an action in the environment based on the generated probability distribution.

Another aspect of the present disclosure is directed to an apparatus for controlling a robotic device based on observed object locations. The apparatus having a memory and one or more processors coupled to the memory. The processor(s) is configured to observe objects in an environment. The processor(s) is also configured to generate a probability distribution for locations of the observed objects. The processor (s) is further configured to control the robotic device to perform an action in the environment based on the generated probability distribution.

This has outlined, rather broadly, the features and technical advantages of the present disclosure in order that the detailed description that follows may be better understood. Additional features and advantages of the present disclosure will be described below. It should be appreciated by those skilled in the art that this present disclosure may be readily utilized as a basis for modifying or designing other structures for carrying out the same purposes of the present disclosure. It should also be realized by those skilled in the art that such equivalent constructions do not depart from the teachings of the present disclosure as set forth in the appended claims. The novel features, which are believed to be characteristic of the present disclosure, both as to its organization and method of operation, together with further objects and advantages, will be better understood from the following description when considered in connection with the accompanying figures. It is to be expressly understood, however, that each of the figures is provided for the purpose of illustration and description only and is not intended as a definition of the limits of the present disclosure.

BRIEF DESCRIPTION OF THE DRAWINGS

The features, nature, and advantages of the present disclosure will become more apparent from the detailed description set forth below when taken in conjunction with the drawings in which like reference characters identify correspondingly throughout.

FIG. 1 illustrates an example of a robot in an environment according to aspects of the present disclosure.

FIG. 2 illustrates an example of a clustering analysis map according to aspects of the present disclosure.

FIG. 3 illustrates an example of generating a clustering analysis map of object locations in an environment by observing object locations over time.

FIG. 4A illustrates an example of an environment according to aspects of the present disclosure.

FIG. 4B illustrate an example of a cost map according to aspects of the present disclosure.

FIG. 4C illustrates an example of a cost map with object locations integrated over a time period according to aspects of the present disclosure.

FIG. 5 is a diagram illustrating an example of a hardware implementation for an object location system according to aspects of the present disclosure.

FIG. 6 illustrates a flow diagram for a method for controlling a robotic device based on observed object locations according to aspects of the present disclosure.

DETAILED DESCRIPTION

The detailed description set forth below, in connection with the appended drawings, is intended as a description of various configurations and is not intended to represent the only configurations in which the concepts described herein may be practiced. The detailed description includes specific details for the purpose of providing a thorough understand-

ing of the various concepts. It will be apparent to those skilled in the art, however, that these concepts may be practiced without these specific details. In some instances, well-known structures and components are shown in block diagram form in order to avoid obscuring such concepts.

Conventional systems (e.g., object detection networks) that perform visual object classification generate false positive and false negative detections. The false positive and false negative detections reduce an accuracy of the detections. A system or method for visual object detection may be combined with a system or method of estimating the detected object location to analyze the object locations. A detected visual object may be identified by a bounding box. The analysis of the object locations may be referred to as a clustering analysis.

Errors may be detected from the clustering analysis. Additionally, true object locations may be estimated based on the clustering analysis. The clustering analysis assumes a static environment and exploits this assumption to count the number of objects in the environment, improve the estimated true location of each object, and reject detections that are not part of a spatial cluster (e.g., false positive detection).

Additionally, or alternatively, objection location estimations may be analyzed using a probability density function. The probability density function (e.g., probability density analysis) does not assume a static environment. The probability density analysis does not use the clustering analysis, object counting, or erroneous detection estimation. In one configuration, a spatial and temporal search for an object may be performed based on the probability density analysis.

A robotic device may use one or more sensors to identify objects in an environment. The sensors may include a red-green-blue (RGB) camera, radio detection and ranging (RADAR) sensor, light detection and ranging (LiDAR) sensor, or another type of sensor. In an image captured by a sensor, a location of one or more identified object may be estimated. Each object's location estimate may be stored along with an estimation time stamp. The estimation time stamp identifies a time when the object was at an estimated location. In one configuration, three-dimensional (3D) object locations are determined by registering detection bounding box centroids with a point cloud produced by a depth camera.

In most systems, images are generated at a number of frames per second, such as ten frames per second. Additionally, each frame may include multiple objects. The number of frames and the number of objects in each frame increase an amount of storage resources used by conventional object detection systems. Additionally, conventional object detection systems may use an increased amount of processing resources for processing the images.

Conventional object detection systems may also generate false positive and false negative location estimations. That is, for each object detected in a frame, some detections may be valid, while others some may be invalid.

Aspects of the present disclosure are directed to using a density based clustering function to estimate the number of objects in the environment. The estimate is based on an assumption that the observation density for true object locations will be greater than a high density threshold. Additionally, the observation density for false positive object locations will be less than a low density threshold.

In the current disclosure, for simplicity, a robotic device may be referred to as a robot. Additionally, the objects may include stationary and dynamic objects in an environment.

The objects may include man-made objects (e.g., chairs, desks, cars, books, etc.), natural objects (e.g., rocks, trees, animals, etc.), and humans.

FIG. 1 illustrates an example of a robot 100 in an environment 102, according to aspects of the present disclosure. In the example of FIG. 1, the robot 100 is a humanoid robot and the environment 102 is a kitchen. Aspects of the present disclosure are not limited to the humanoid robot. The robot 100 may be any type of autonomous or semi-autonomous device, such as a drone or a vehicle. Additionally, the robot 100 may be in any type of environment.

In one configuration, the robot 100 obtains one or more images of the environment 102 via one or more sensors of the robot 100. The robot 100 may localize one or more objects in each image. Localization refers to determining the location (e.g., coordinates) of the object within the image. In conventional object detection systems, a bounding box may be used to indicate the location of the object in the image. The localized objects may be one or more specific classes of objects, such as chairs 104, or all objects in the image. The objects may be localized via an object detection system, such as a pre-trained object detection neural network.

As discussed, one or more identified objects in an image (e.g., frame) may be localized. In one configuration, object location estimators are observed over time in an environment. Specifically, a clustering analysis may be applied to the object location estimators over various temporal windows (e.g., different time periods).

FIG. 2 illustrates an example of a clustering analysis map 200, according to aspects of the present disclosure. In the example of FIG. 2, object location estimators were observed over time in an environment. Each node 202, 208 identifies a location of an estimated location of an object in the environment. As shown in FIG. 2, some nodes 202 (e.g., non-clustered nodes 202) may not be associated with a cluster. Other nodes 208 (e.g., in-clustered nodes 208) may be clustered around a centroid 204. Non-clustered nodes 202 may be considered false positives. For brevity, labels are not provided for each non-clustered node 202 and in-clustered node 208 of FIG. 2.

In the example of FIG. 2, in-clustered nodes 208 may be edge nodes 210 or core nodes 212. A size (e.g., diameter) of an edge in-clustered node 210 is less than a size of core in-clustered node 212. Each edge in-clustered node 210 is associated with one neighbor node, where the distance of the neighbor node is less than a distance threshold. Each core in-clustered nodes 212 is associated with one neighbor node, where the distance of the neighbor node is less than a distance threshold. In one configuration, because the environment is a 3D environment, the centroid 204 is a 3D centroid. The centroid 204 may be associated with the object's true location.

In general, the clustered nodes 202 are location estimates of a same object or a same class of objects. For example, in an office, a location of office furniture may be less dynamic than a location of office supplies, such as pens or paper. The nodes 202 within an area of a location may be clustered. For example, the nodes 202 corresponding to a location of an office chair may form a cluster if the location is within a specific area over the course of the observations. The 3D centroid 204 (e.g., mean centroid) may be determined for each cluster.

For density based clustering, accuracy of the object location estimates may be determined based on the 3D centroid 204 of each cluster. Estimating an object's location based on the 3D centroid 204 may improve the accuracy of the

estimation. Additionally, the object localization functions accuracy may be determined based on a 3D standard deviation of each cluster. The 3D mean of each cluster may be used to estimate an object's true location. The standard deviation may be used to determine the variance of a cluster. When the standard deviation is greater than a threshold, there may be uncertainty in an object's true location. The uncertainty in the object's location decreases when a variance in the standard deviation decreases. As such, the standard deviation correlates to an accuracy of the object's location.

FIG. 3 illustrates an example of generating a clustering analysis map 350, according to aspects of the present disclosure. As shown in FIG. 3, the clustering analysis map 350 is based on observations of objects 302, 304 in an environment 300. In FIG. 3, a robot may observe the location of the objects (e.g., chairs 304 and table 302) in the environment 300 over a period of time. The robot may localize the locations of the chairs 304 and the table 302 for each observation.

After a number of observations, the robot may perform a clustering analysis to generate a clustering analysis map 350. The clustering analysis map 350 includes in-clustered nodes 314 and non-clustered nodes 316. The non-clustered nodes 316 are isolated observations from the environment 300. The in-clustered nodes 314 may be considered repeated observations of the same object 304, 302. Each centroid 318, 320, 322 is considered a true object location. For example, centroids 318, 322 correspond to the estimated true location of the chairs 304. Centroid 320 corresponds to the estimated true location of the table 302. The centroids 318, 320, 322 may be 3D centroids.

False positives are common in conventional convolutional neural network detectors. The clustering analysis (e.g., density based clustering) improves an estimate of a number of objects in a spatial environment, thereby reducing false positives. Density based clustering may be used to set density criteria for multiple detections in a similar, or same, location. As such, repeated observations of the same object may be considered. That is, in one configuration, density based clustering is used to consider multiple observations and discard isolated observations. Additionally, density based clustering may fuse repeated observations into unified object instances. Each cluster represents an instance of an object, such as a bottle or a plant.

In the example of FIG. 2, each cluster corresponds to one object based on multiple observations of the object. For example, hundreds or thousands of individual detections of the objects may be used for each cluster. Additionally, the clustering analysis may be used to determine a quality of the location estimations. That is, the clustering analysis may determine how well the location estimators are performing.

In one configuration, by observing object locations over time, a probability distribution of object locations in an environment may be generated. That is, the probability density function (e.g., probability density analysis) estimates a continuous distribution using the set of observations to represent the probability of finding an object at any given location on a map. Accordingly, the probability density function may be used to represent spatial and temporal knowledge about objects in the spatial environment.

A memory footprint of the data corresponding to the estimated probability density function is less than a memory footprint for data corresponding to conventional object localization systems. For example, conventional object localization systems may generate a 1 GB object observation log. In contrast, the estimated probability density function

may be 50 KB. Due to the reduced memory footprint, the kernel density estimation may be transferred between robotic devices. As such, robotic devices may share knowledge about object locations in an environment.

As discussed, the probability density function analysis estimates a continuous distribution using all of the observations configurations. The probability distribution of object locations in an environment is generated by observing object locations over time. Random samples may be extracted from the probability distribution. A random sample from the distribution may have a higher likelihood of being in an area of high probability as opposed to an area of low probability. Additionally, evaluating the distribution over the map grid may provide a heat map (e.g., cost map) showing areas of the map that are more likely to contain objects.

A robotic device may sample areas of high probability to reduce a search time for an object. If the object is not in the high probability area (e.g., a most likely location), the object search may then sample areas of low probability. The probability distribution may reduce search times by distinguishing high probability and low probability areas. In one configuration, if an object is found in an unlikely location (e.g., low probability area), the robotic device may place the object back in a most likely location (e.g., high probability area).

As another example, specific objects may correspond to specific high probability and low probability areas. For example, a first set of chairs are for a kitchen dining table, and a second set of chairs are for a formal dining table. In this example, the kitchen is a high probability area for the first set of chairs and an area outside of the kitchen is a low probability area. For example, the formal dining room may be a low probability area for the first set of chairs. Additionally, the formal dining room is a high probability area for the second set of chairs and the kitchen is a low probability area for the second set of chairs.

FIG. 4A illustrates an image of an environment 400, according to aspects of the present disclosure. In the example of FIG. 4A, the environment includes a road 402 and a sidewalk 424. In one configuration, a robot observes the environment 400 over time. For example, the robot may observe the environment 400 over a twenty-four hour period. Based on the observations, the robot generates a probability distribution of object locations in the environment 400. The objects may be humans or another type of object. The probability distributions may be used to generate a cost map.

As shown in FIG. 4B, a cost map 450 may be generated based on a probability density analysis of objects in an environment. The cost map 450 may be based on a 2D or 3D probability density function. As such, any type of map, such as an ego view or top-down view, may be generated.

In the cost map 450, low probability areas may be distinguished from high probability areas. Each area may be assigned a probability value. The probability value, or probability value range, may be visualized by color or other visual representation. For example, a color or shading of low probability areas may be different from high probability areas. Different granularity levels may be assigned to an occupancy probability. For example, as shown in FIG. 4B, the probability areas may be low 404, mild 406, medium 408, high 410, or very high 412. The probability of an object being present at a location may be determined from a value assigned to the location in the cost map. The probability of an object being present at a location may be referred to as an object probability.

FIG. 4C illustrates an example of the cost map 450 with object locations 452 integrated over a time period. In the example of FIG. 4C, the time period may be a twenty-four hour period. Other time periods, such as hours, days, weeks, months, etc., are also contemplated.

In the example of FIG. 4C, each object location 452 represents the location of an object, such as a human. The cost map 450 is generated based on a probability density analysis. As discussed, the probability density analysis estimates a continuous distribution using all of the observations in order to represent the probability of finding an object at any given location on the cost map 450. The cost map may be considered a Gaussian mixture model with some number of 2D Gaussian functions fit to the data.

Each object or object type may correspond to specific high probability and low probability areas. For example, for kitchenware in a home environment, a high probability may be the kitchen and a low probability area may be a bathroom. As another example, for clothes in a home environment, a high probability area may be one or more closets and a low probability area may be the kitchen. According to aspects of the present disclosure, a robotic device may return objects in the environment to their "home" location (e.g., high probability area) without user instruction or user intervention.

As discussed, cost maps may be generated from the probability distributions. The cost maps may be overlaid over a map of an environment. In one configuration, a path planning function may use an overlaid cost map to minimize or maximize the probability of encountering specific objects. As shown in FIGS. 4B and 4C, low probability areas may be distinguished from high probability areas.

In one configuration, the robot uses the cost map overlay for route planning. For example, the robot's route may be planned such that the robot may detour regions where it is likely to encounter an object, such as a human. The robot may also use the inverse of the cost map to paths with an increased likelihood of encountering an object, such as the human. That is, probability density functions may be explored to maximize or minimize the chance of encountering one or more objects.

In one example, to avoid one or more objects, the robot may plan a route that avoids probability areas that are greater than a threshold. For example, the robot may avoid medium 408, high 410, and very high 412 probability areas (see FIGS. 4B-C). The aforementioned probability areas are used for an example, the threshold may be determined based on the robot's task, fuel level, costs, and/or other factors.

In another example, to encounter one or more objects, the robot may plan a route that includes probability areas that are greater than a threshold. For example, the robot may plan a route that includes medium 408, high 410, and very high 412 probability areas (see FIGS. 4B-C). Additionally, or alternatively, in this example, the robot may plan a route that avoids probability areas that are less than a threshold. For example, the robot may plan a route that avoids low 404 and/or mild 406 probability areas (see FIGS. 4B-C). The aforementioned probability areas are used for an example, the threshold may be determined based on the robot's task, fuel level, costs, and/or other factors.

According to aspects of the present disclosure stationary objects, such as sofas, tables, and potted plants, have stationary location probability density functions. Dynamic objects, such as people, coffee mugs, pets, and chairs, are dynamic and have non-stationary probability density functions. For objects with non-stationary distributions, the probability density function may be estimated at different frequencies. For example, the probability density function

may be estimated during one or more different times of the day, different days of the week, different months, etc.

In one configuration, the probability of finding non-dynamic objects in this environment may vary based on time. In one configuration, a robot may search for dynamic objects based on a temporally relevant distribution. The search may be limited to specific objects or any other things that a robot can recognize and estimate the location of.

In some embodiments, based on the probability distribution of people in a space, algorithms may determine furniture placement to minimize impact on flow of foot traffic, or to identify unused furniture in a space. Based on the probability distribution of people in a space, optimal advertisement placement in an environment may be determined. In addition, based on the probability distribution of people in a space, it may be determined how long people are engaged with advertisements at different locations.

FIG. 5 is a diagram illustrating an example of a hardware implementation for an object location system 500, according to aspects of the present disclosure. The object location system 500 may be a component of a vehicle, a robotic device, or another device. For example, as shown in FIG. 5, the object location system 500 is a component of a robot 528 (e.g., robotic device).

Aspects of the present disclosure are not limited to the object location system 500 being a component of the robot 528. Other devices, such as a bus, boat, drone, or vehicle, are also contemplated for using the object location system 500. The robot 528 may operate in at least an autonomous operating mode and a manual operating mode.

The object location system 500 may be implemented with a bus architecture, represented generally by a bus 550. The bus 550 may include any number of interconnecting buses and bridges depending on the specific application of the object location system 500 and the overall design constraints. The bus 550 links together various circuits including one or more processors and/or hardware modules, represented by a processor 520, a communication module 522, a location module 518, a sensor module 502, a locomotion module 526, a navigation module 525, and a computer-readable medium 515. The bus 550 may also link various other circuits such as timing sources, peripherals, voltage regulators, and power management circuits, which are well known in the art, and therefore, will not be described any further.

The object location system 500 includes a transceiver 516 coupled to the processor 520, the sensor module 502, an object location analysis module 508, the communication module 522, the location module 518, the locomotion module 526, the navigation module 525, and the computer-readable medium 515. The transceiver 516 is coupled to an antenna 555. The transceiver 516 communicates with various other devices over a transmission medium. For example, the transceiver 516 may receive commands via transmissions from a user or a remote device. As another example, the transceiver 516 may transmit statistics and other information from the object location analysis module 508 to a server (not shown).

The object location system 500 includes the processor 520 coupled to the computer-readable medium 515. The processor 520 performs processing, including the execution of software stored on the computer-readable medium 515 providing functionality according to the disclosure. The software, when executed by the processor 520, causes the object location system 500 to perform the various functions described for a particular device, such as the robot 528, or any of the modules 502, 508, 515, 516, 518, 520, 522, 525,

526. The computer-readable medium 515 may also be used for storing data that is manipulated by the processor 520 when executing the software.

The sensor module 502 may be used to obtain measurements via different sensors, such as a first sensor 506 and a second sensor 505. The first sensor 506 may be a vision sensor, such as a stereoscopic camera or a red-green-blue (RGB) camera, for capturing 2D images. The second sensor 505 may be a ranging sensor, such as a light detection and ranging (LIDAR) sensor or a radio detection and ranging (RADAR) sensor. Of course, aspects of the present disclosure are not limited to the aforementioned sensors as other types of sensors, such as, for example, thermal, sonar, and/or lasers are also contemplated for either of the sensors 505, 506.

The measurements of the first sensor 506 and the second sensor 505 may be processed by one or more of the processor 520, the sensor module 502, the object location analysis module 508, the communication module 522, the location module 518, the locomotion module 526, the navigation module 525, in conjunction with the computer-readable medium 515 to implement the functionality described herein. In one configuration, the data captured by the first sensor 506 and the second sensor 505 may be transmitted to an external device via the transceiver 516. The first sensor 506 and the second sensor 505 may be coupled to the robot 528 or may be in communication with the robot 528.

The location module 518 may be used to determine a location of the robot 528. For example, the location module 518 may use a global positioning system (GPS) to determine the location of the robot 528. The communication module 522 may be used to facilitate communications via the transceiver 516. For example, the communication module 522 may be configured to provide communication capabilities via different wireless protocols, such as WiFi, long term evolution (LTE), 5G, etc. The communication module 522 may also be used to communicate with other components of the robot 528 that are not modules of the object location system 500.

The locomotion module 526 may be used to facilitate locomotion of the robot 528. As another example, the locomotion module 526 may be in communication with one or more power sources of the robot 528, such as a motor and/or batteries. The locomotion may be proved via wheels, moveable limbs, propellers, treads, fins, jet engines, and/or other sources of locomotion.

The object location system 500 also includes the navigation module 525 for planning a route or controlling the locomotion of the robot 528, via the locomotion module 526. A route may be planned based on data provided via the object location analysis module 508. The modules may be software modules running in the processor 520, resident/stored in the computer-readable medium 515, one or more hardware modules coupled to the processor 520, or some combination thereof.

The object location analysis module 508 may be in communication with the sensor module 502, the transceiver 516, the processor 520, the communication module 522, the location module 518, the locomotion module 526, the navigation module 525, and the computer-readable medium 515. In one configuration, the object location analysis module 508 receives sensor data from the sensor module 502. The sensor module 502 may receive the sensor data from the first sensor 506 and the second sensor 505. According to aspects of the present disclosure, the sensor module 502 may filter the data to remove noise, encode the data, decode the data,

merge the data, extract frames, or perform other functions. In an alternate configuration, the object location analysis module 508 may receive sensor data directly from the first sensor 506 and the second sensor 505.

In one configuration, the object location analysis module 508 determines a probability area for objects based on information from the processor 520, the location module 518, the computer-readable medium 515, the first sensor 506, and/or the second sensor 505. The probability areas may be determined by generating a probability distribution of the object locations for the observed objects. Based on the probability areas, the object location analysis module 508 may control one or more actions of the robot 528.

For example, the action may be planning a route for the robot 528. The route may be determined based on a given task, such as encountering or avoiding one or more objects. The object location analysis module 508 may plan the route via the processor 520, the location module 518, the computer-readable medium 515, the locomotion module 526, and/or the navigation module 525.

FIG. 6 illustrates a flow diagram 600 for controlling a robotic device based on observed object locations according to aspects of the present disclosure. For simplicity, the robotic device will be referred to as a robot. As shown in FIG. 6, at block 602 the robot observes objects in an environment. The objects may be observed via one or more sensors of the robot, such as LiDAR, RADAR, and/or an RGB camera. The objects may be observed over a period of time, such as hours, days, etc.

At block 604, the robot generates a probability distribution for locations of the observed objects. In one configuration, the robot estimates a continuous distribution using the observations of the objects over the period of time. In this configuration, the probability distribution is based on the continuous distribution.

A cost map may be generated from the probability distribution. The cost map may also be referred to as a heat map. The cost map may be overlaid on a map of the environment. The cost map may be a 2D or 3D cost map.

At block 606, the robot performs an action in the environment based on the generated probability distribution. The robot may be controlled based on the cost map. As an example, the robot may avoid a first area in the environment with a first object probability that is greater than a first threshold. Additionally, or alternatively, the robot may navigate to a second area in the environment with a second object probability that is greater than a second threshold.

In another configuration, the robot may identify an out of place object based on the probability distribution. For example, the robot may identify an object in a low probability area. The robot may place the mis-placed object back to an intended (e.g., correct) location.

Based on the teachings, one skilled in the art should appreciate that the scope of the present disclosure is intended to cover any aspect of the present disclosure, whether implemented independently of or combined with any other aspect of the present disclosure. For example, an apparatus may be implemented or a method may be practiced using any number of the aspects set forth. In addition, the scope of the present disclosure is intended to cover such an apparatus or method practiced using other structure, functionality, or structure and functionality in addition to, or other than the various aspects of the present disclosure set forth. It should be understood that any aspect of the present disclosure may be embodied by one or more elements of a claim.

The word "exemplary" is used herein to mean "serving as an example, instance, or illustration." Any aspect described herein as "exemplary" is not necessarily to be construed as preferred or advantageous over other aspects.

Although particular aspects are described herein, many variations and permutations of these aspects fall within the scope of the present disclosure. Although some benefits and advantages of the preferred aspects are mentioned, the scope of the present disclosure is not intended to be limited to particular benefits, uses or objectives. Rather, aspects of the present disclosure are intended to be broadly applicable to different technologies, system configurations, networks and protocols, some of which are illustrated by way of example in the figures and in the following description of the preferred aspects. The detailed description and drawings are merely illustrative of the present disclosure rather than limiting, the scope of the present disclosure being defined by the appended claims and equivalents thereof.

As used herein, the term "determining" encompasses a wide variety of actions. For example, "determining" may include calculating, computing, processing, deriving, investigating, looking up (e.g., looking up in a table, a database or another data structure), ascertaining and the like. Additionally, "determining" may include receiving (e.g., receiving information), accessing (e.g., accessing data in a memory) and the like. Furthermore, "determining" may include resolving, selecting, choosing, establishing, and the like.

As used herein, a phrase referring to "at least one of" a list of items refers to any combination of those items, including single members. As an example, "at least one of: a, b, or c" is intended to cover: a, b, c, a-b, a-c, b-c, and a-b-c.

The various illustrative logical blocks, modules and circuits described in connection with the present disclosure may be implemented or performed with a processor specially configured to perform the functions discussed in the present disclosure. The processor may be a neural network processor, a digital signal processor (DSP), an application specific integrated circuit (ASIC), a field programmable gate array signal (FPGA) or other programmable logic device (PLD), discrete gate or transistor logic, discrete hardware components or any combination thereof designed to perform the functions described herein. Alternatively, the processing system may comprise one or more neuromorphic processors for implementing the neuron models and models of neural systems described herein. The processor may be a microprocessor, controller, microcontroller, or state machine specially configured as described herein. A processor may also be implemented as a combination of computing devices, e.g., a combination of a DSP and a microprocessor, a plurality of microprocessors, one or more microprocessors in conjunction with a DSP core, or such other special configuration, as described herein.

The steps of a method or algorithm described in connection with the present disclosure may be embodied directly in hardware, in a software module executed by a processor, or in a combination of the two. A software module may reside in storage or machine readable medium, including random access memory (RAM), read only memory (ROM), flash memory, erasable programmable read-only memory (EPROM), electrically erasable programmable read-only memory (EEPROM), registers, a hard disk, a removable disk, a CD-ROM or other optical disk storage, magnetic disk storage or other magnetic storage devices, or any other medium that can be used to carry or store desired program code in the form of instructions or data structures and that can be accessed by a computer. A software module may

comprise a single instruction, or many instructions, and may be distributed over several different code segments, among different programs, and across multiple storage media. A storage medium may be coupled to a processor such that the processor can read information from, and write information to, the storage medium. In the alternative, the storage medium may be integral to the processor.

The methods disclosed herein comprise one or more steps or actions for achieving the described method. The method steps and/or actions may be interchanged with one another without departing from the scope of the claims. In other words, unless a specific order of steps or actions is specified, the order and/or use of specific steps and/or actions may be modified without departing from the scope of the claims.

The functions described may be implemented in hardware, software, firmware, or any combination thereof. If implemented in hardware, an example hardware configuration may comprise a processing system in a device. The processing system may be implemented with a bus architecture. The bus may include any number of interconnecting buses and bridges depending on the specific application of the processing system and the overall design constraints. The bus may link together various circuits including a processor, machine-readable media, and a bus interface. The bus interface may be used to connect a network adapter, among other things, to the processing system via the bus. The network adapter may be used to implement signal processing functions. For certain aspects, a user interface (e.g., keypad, display, mouse, joystick, etc.) may also be connected to the bus. The bus may also link various other circuits such as timing sources, peripherals, voltage regulators, power management circuits, and the like, which are well known in the art, and therefore, will not be described any further.

The processor may be responsible for managing the bus and processing, including the execution of software stored on the machine-readable media. Software shall be construed to mean instructions, data, or any combination thereof, whether referred to as software, firmware, middleware, microcode, hardware description language, or otherwise.

In a hardware implementation, the machine-readable media may be part of the processing system separate from the processor. However, as those skilled in the art will readily appreciate, the machine-readable media, or any portion thereof, may be external to the processing system. By way of example, the machine-readable media may include a transmission line, a carrier wave modulated by data, and/or a computer product separate from the device, all which may be accessed by the processor through the bus interface. Alternatively, or in addition, the machine-readable media, or any portion thereof, may be integrated into the processor, such as the case may be with cache and/or specialized register files. Although the various components discussed may be described as having a specific location, such as a local component, they may also be configured in various ways, such as certain components being configured as part of a distributed computing system.

The machine-readable media may comprise a number of software modules. The software modules may include a transmission module and a receiving module. Each software module may reside in a single storage device or be distributed across multiple storage devices. By way of example, a software module may be loaded into RAM from a hard drive when a triggering event occurs. During execution of the software module, the processor may load some of the instructions into cache to increase access speed. One or more cache lines may then be loaded into a special purpose

register file for execution by the processor. When referring to the functionality of a software module below, it will be understood that such functionality is implemented by the processor when executing instructions from that software module. Furthermore, it should be appreciated that aspects of the present disclosure result in improvements to the functioning of the processor, computer, machine, or other system implementing such aspects.

If implemented in software, the functions may be stored or transmitted over as one or more instructions or code on a computer-readable medium. Computer-readable media include both computer storage media and communication media including any storage medium that facilitates transfer of a computer program from one place to another.

Further, it should be appreciated that modules and/or other appropriate means for performing the methods and techniques described herein can be downloaded and/or otherwise obtained by a user terminal and/or base station as applicable. For example, such a device can be coupled to a server to facilitate the transfer of means for performing the methods described herein. Alternatively, various methods described herein can be provided via storage means, such that a user terminal and/or base station can obtain the various methods upon coupling or providing the storage means to the device. Moreover, any other suitable technique for providing the methods and techniques described herein to a device can be utilized.

It is to be understood that the claims are not limited to the precise configuration and components illustrated above. Various modifications, changes, and variations may be made in the arrangement, operation, and details of the methods and apparatus described above without departing from the scope of the claims.

What is claimed is:

1. A method for controlling a robotic device based on observed object locations, comprising:

obtaining a plurality of observations of an object type in an environment based on identifying one or more first objects associated with the object type over a period of time in the environment;

localizing the object type in the environment based on a location cluster comprising a plurality of nodes, each node in the plurality of nodes associated with one observation of the plurality of observations;

generating a cost map indicating a probability distribution of the object type in relation to the localized object type in the environment; and

controlling the robotic device to perform an action in the environment based on the cost map and a map of the environment.

2. The method of claim 1, further comprising generating the probability distribution based on the plurality of observations.

3. The method of claim 2, further comprising estimating a continuous distribution based on plurality of observations.

4. The method of claim 3, wherein the probability distribution is based on the continuous distribution.

5. The method of claim 4, further comprising controlling the robotic device to:

avoid a first area in the environment with a first object type probability that is greater than a first threshold; or

navigate to a second area in the environment with a second object type probability that is greater than a second threshold.

6. The method of claim 1, further comprising:

identifying a second object associated with the object type at a first location in the environment; and

determining, based on the cost map, that a probability of the second object being at the first location is less than a threshold.

7. The method of claim 6, further comprising:

identifying, based on the cost map, a second location in the environment, wherein the probability of the second object being at the second location is greater than the threshold; and

controlling the robotic device to place the second object at the second location.

8. An apparatus for controlling a robotic device based on observed object locations, the apparatus comprising:

a processor; and

a memory coupled with the processor and storing instructions operable, when executed by the processor, to cause the apparatus to:

obtain a plurality of observations of an object type in an environment based on identifying one or more first objects associated with the object type over a period of time in the environment;

localize the object type in the environment based on a location cluster comprising a plurality of nodes, each node in the plurality of nodes associated with one observation of the plurality of observations;

generate a cost map indicating a probability distribution of the object type in relation to the localized object type in the environment; and

control the robotic device to perform an action in the environment based on the cost map and a map of the environment.

9. The apparatus of claim 8, wherein execution of the instructions further cause the apparatus to generate the probability distribution based on the plurality of observations.

10. The apparatus of claim 9, wherein execution of the instructions further cause the apparatus to estimate a continuous distribution based on plurality of observations.

11. The apparatus of claim 10, wherein the probability distribution is based on the continuous distribution.

12. The apparatus of claim 11, wherein execution of the instructions further cause the apparatus to control the robotic device to:

avoid a first area in the environment with a first object type probability that is greater than a first threshold; or

navigate to a second area in the environment with a second object type probability that is greater than a second threshold.

13. The apparatus of claim 8, wherein execution of the instructions further cause the apparatus to:

identify a second object associated with the object type at a first location in the environment; and

determine, based on the cost map, that a probability of the second object being at the first location is less than a threshold.

14. The apparatus of claim 13, wherein execution of the instructions further cause the apparatus to:

identify, based on the cost map, a second location in the environment, wherein the probability of the second object being at the second location is greater than the threshold; and

control the robotic device to place the second object at the second location.

15. A non-transitory computer-readable medium having program code recorded thereon for controlling a robotic device based on observed object locations, the program code executed by a processor and comprising:

program code to obtain a plurality of observations of an object type in an environment based on identifying one or more first objects associated with the object type over a period of time in the environment;

program code to localize the object type in the environment based on a location cluster comprising a plurality of nodes, each node in the plurality of nodes associated with one observation of the plurality of observations;

program code to generate a cost map indicating a probability distribution of the object type in relation to the localized object type in the environment; and

program code to control the robotic device to perform an action in the environment based on the cost map and a map of the environment.

16. The non-transitory computer-readable medium of claim **15**, wherein the program code further comprises program code to generate the probability distribution based on the plurality of observations.

17. The non-transitory computer-readable medium of claim **16**, wherein the program code further comprises program code to estimate a continuous distribution based on plurality of observations.

18. The non-transitory computer-readable medium of claim **17**, wherein the probability distribution is based on the continuous distribution.

19. The non-transitory computer-readable medium of claim **18**, wherein the program code further comprises program code to control the robotic device to:

avoid a first area in the environment with a first object type probability that is greater than a first threshold; or

navigate to a second area in the environment with a second object type probability that is greater than a second threshold.

20. The non-transitory computer-readable medium of claim **15**, wherein the program code further comprises:

program code to identify a second object associated with the object type at a first location in the environment;

program code to determine, based on the cost map, that a probability of the second object being at the first location is less than a threshold;

program code to identify, based on the cost map, a second location in the environment, wherein the probability of the second object being at the second location is greater than the threshold; and

program code to control the robotic device to place the second object at the second location.

* * * * *