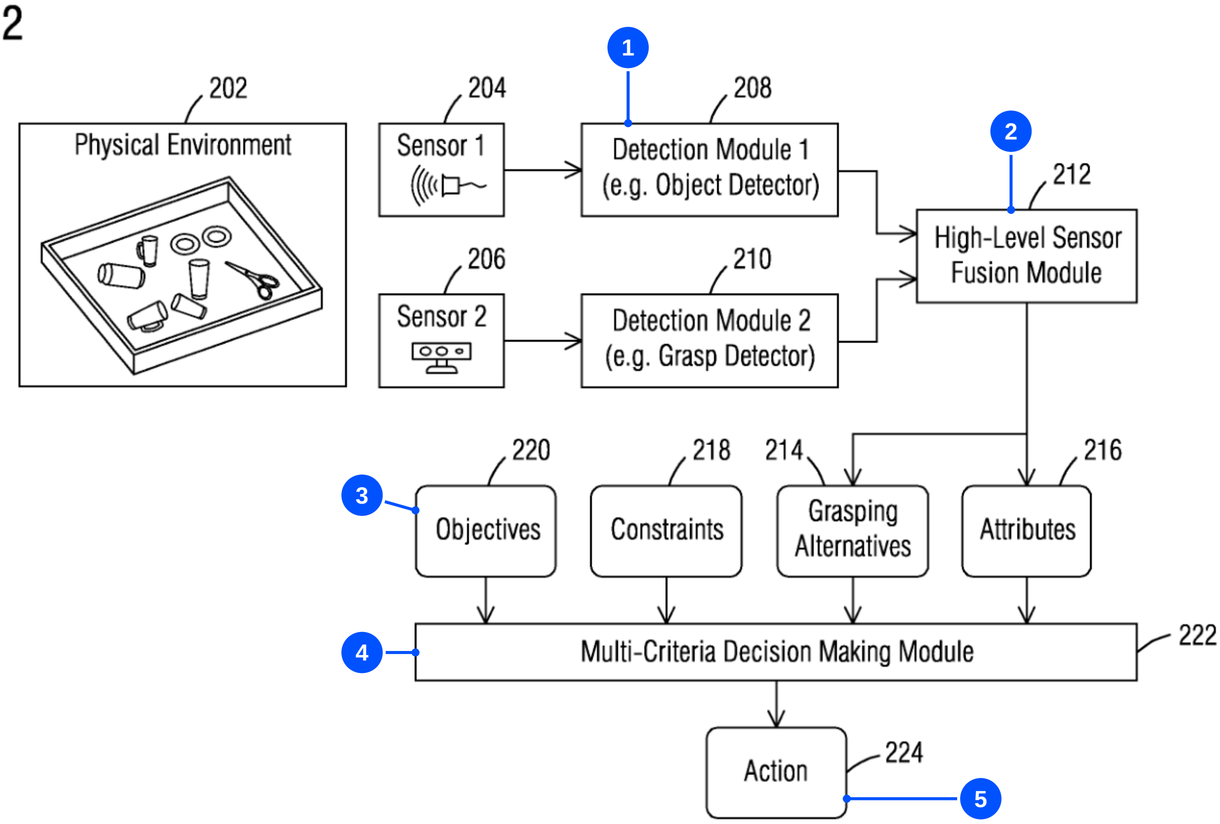
Seeing an object is only the beginning.

How detection outputs become a grasping decision. Read the mechanism on its original drawing.

FIG. 2



Inspect untouched FIG. 2 / original PDF page 3

Rotated upright and cropped. Original geometry and labels are preserved. Blue leaders identify parts; they are not additional process connections.

Detection supplies possibilities. Criteria guide the choice.

The disclosed system separates detection from the choice of action. It combines what the detectors report, attaches attributes to candidate grasps, then ranks the alternatives using the application's objectives and constraints. That distinction makes the decision criteria visible rather than treating a predicted grasp score as the whole decision.

Read the supporting description / paragraph [0027] / original page 8

Follow the decision.

01 Find objects and possible grasps.

The example object detector locates objects of interest; the grasp detector supplies candidate grasp locations. The description also allows both inputs to come from one sensor.

208 / 210 / Detection modules / [0027] / source p. 8

02 Relate the outputs.

The fusion module combines detection outputs and computes attributes for the alternatives, including relationships between candidate grasps and detected objects.

212 / High-level sensor fusion / [0027] / source p. 8

03 Specify what matters.

The ranking can reflect which objects to pick and constraints imposed by the environment. In the described embodiments these can be predefined or supplied by a user.

220 / 218 / Objectives and constraints / [0027] / source p. 8

04 Rank the alternatives.

Each criterion in the described decision matrix has a weight. A candidate is evaluated across those criteria before a grasp is selected. The patent gives several possible decision methods.

222 / Multi-criteria decision making / [0045] / source p. 10

05 Turn the selection into instructions.

The selected alternative defines the output action. Executable code can then be sent to the robot controller to carry out the selective grasp.

224 / Action / [0047] / source p. 11

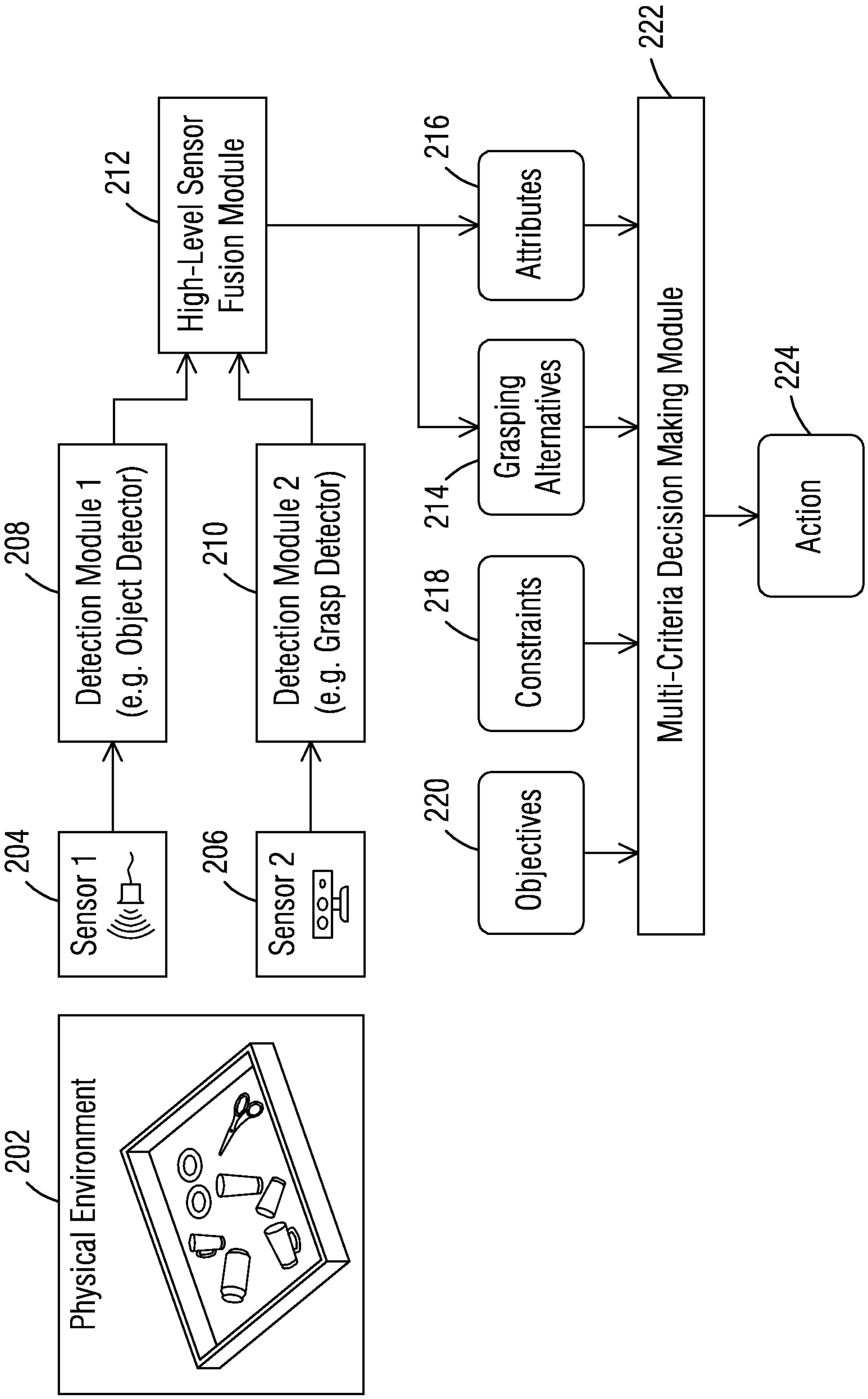
What changes after an attempt?

Paragraph [0048] describes an optional way to adjust criterion weights using simulation or real-world results, within ranges initially defined by an expert. Improved settings can become the starting point for the next optimization step. This is a specified tuning procedure, not evidence of unrestricted learning or guaranteed safe performance.

Inspect optional feedback / [0048] / original page 11

One disclosed embodiment. The drawing and description establish what is proposed; they do not establish deployment, measured picking performance or a safety guarantee. This is a source study, not the completed 28-publication robotics report. Source review: Codex; no independent-model approval.

FIG. 2


202
Physical Environment
204
Sensor 1
206
Sensor 2
208
Detection Module 1
(e.g. Object Detector)
210
Detection Module 2
(e.g. Grasp Detector)
212
High-Level Sensor
Fusion Module
220
Objectives
218
Constraints
214
Grasping
Alternatives
216
Attributes
222
Multi-Criteria Decision Making Module
224
Action

in addition to determining an optimal grasp, it is necessary to perform a semantic recognition of the objects **118** in the scene.

[0022] Bin picking of assorted or unknown objects may involve a combination an object detection algorithm, to localize an object of interest among the assorted pile, and a grasp detection algorithm to compute grasps given a 3D map of the scene. The object detection and grasp detection algorithms may comprise AI solutions, e.g., neural networks. The state-of-the-art lacks a systematic approach that tackles decision making as a combination of the output of said algorithms.

[0023] In current practice, new robotic grasping motions are typically sampled from all possible alternatives via a series of mostly disconnected conditional statements scattered throughout the codebase. These conditional statements check for possible workspace violations, affiliation of grasps to detected objects, combined object detection and grasping accuracy, etc. Overall, this approach lacks the flexibility and scalability required when, for example, another AI solution is added to solve the problem, more constraints are imposed, or more sensorial input is introduced.

[0024] Another approach is to combine the grasping and object detection in a single AI solution, e.g. a single neural network. While this approach tackles some of the decision-making uncertainty (e.g. affiliation of grasps to detected objects and combined expected accuracy), it does not allow inclusion of constraints imposed by the environment (e.g., workspace violations). Additionally, training such specific neural networks may not be straight-forward as abundant training data may be required but not available to the extent needed; this is unlike well-vetted generic object and grasp detection algorithms, which use mainstream datasets available through the AI community.

[0025] Embodiments of the present disclosure address at least some of the aforementioned technical challenges. The described embodiments utilize high-level sensor fusion (HLSF) and multi-criteria decision making (MCDM) methodologies to select an optimal alternative grasping action based on outputs from multiple detection algorithms in a bin picking application.

[0026] FIG. **2** is a block diagram illustrating functional blocks for executing autonomous bin picking according to described embodiments. The functional blocks may be implemented by an autonomous system such as that shown in FIG. **1**. At least some of the functional blocks are represented as modules. The term "module", as used herein, refers to a software component or part of a computer program that contains one or more routines. In some cases, a module can comprise an AI algorithm, such as a neural network. The modules that make up a computer program can be independent and interchangeable and are each configured to execute one aspect of a desired functionality. In embodiments, the computer program, of which the described modules are a part, includes code for autonomously executing a skill function (i.e., pick up or grasp an object) by a controllable device, such as a robot.

[0027] Referring to FIG. **2**, the described system includes multiple sensors, such as a first sensor **204** and a second sensor **206**, that are configured to capture images of a physical environment **202** comprising objects placed in a bin. In the presently described embodiment, the objects in the bin are of mixed types. The sensors **204**, **206** may provide multi-modal sensorial inputs, and/or may be positioned at different locations to capture different views of the physical environment **202** including the bin. The system utilizes multiple detection modules, such as one or more object detection modules **208** and one or more grasp detection modules **210**, which feed from different sensorial inputs. In the shown example, a first image captured by the first sensor **204** is sent to an object detection module **208** and a second image captured by the second sensor **206** is sent to a grasp detection module **210**. Based on the first image, the object detection module **208** generates a first output locating one or more objects of interest in the first image. Based on the second image, the grasp detection module **210** generates a second output defining a plurality of grasping alternatives that correspond to a plurality of locations in the second image. The terms "first image" and "second image" do not necessarily imply that the first image and the second image are different, and indeed in some embodiments (described later) refer to the same image captured by a single sensor. An HLSF module **212** combines the multiple outputs from the multiple detection modules, such as the above-described first and second outputs, to compute attributes **216** for each of the grasping alternatives **214**. The attributes **216** include functional relationships between the grasping alternatives and the located objects. An MCDM module **222** ranks the grasping alternatives **214** based on the computed attributes **216** to select one of the grasping alternatives for execution. The ranking may be generated based on the objectives **220** of the bin picking application (e.g., specific type or types of object to be picked) and constraints **218** that may be imposed by the physical environment (e.g., dimension and location of the bin). In various embodiments, the objectives **220** and/or constraints **218** may be predefined, or may be specified by a user, for example via a Human Machine Interface (HMI) panel. The MCDM module **222** outputs an action **224** defined by the selected grasping alternative, based on which executable instructions are generated to operate the controllable device or robot to selectively grasp an object from the bin.

[0028] Exemplary and non-limiting embodiments of the functional blocks will now be described.

[0029] Object detection is a problem in computer vision that involves identifying the presence, location, and type of one or more objects in a given image. It is a problem that involves building upon methods for object localization and object classification. Object localization refers to identifying the location of one or more objects in an image and drawing a contour or a bounding box around their extent. Object classification involves predicting the class of an object in an image. Object detection combines these two tasks and localizes and classifies one or more objects in an image.

[0030] Many of the known object detection algorithms work in the RGB (red-green-blue) color space. Accordingly, the first image sent to the object detection module **208** may define an RGB color image. Alternately, the first image may comprise a point cloud with color information for each point in the point cloud (in addition to coordinates in 3D space).

[0031] In one embodiment, the object detection module **208** comprises a neural network, such as a segmentation neural network. An example of a neural network architecture suitable for the present purpose is a mask region-based convolutional neural network (Mask R-CNN). Segmentation neural networks provide pixel-wise object recognition outputs. The segmentation output may present contours of arbitrary shapes as the labeling granularity is done at a pixel

[0041] FIG. **3** illustrates a portion of a coherent representation of a physical environment obtained by combining outputs of multiple algorithms. Here, the outputs produced by multiple detection modules **208**, **210** are aligned to a common real-world coordinate system to produce a coherent representation **300**. The common coordinate system may be arbitrarily selected by the HLSF module **212** or may be specified by a user input. Each location (pixel or other defined region) of the representation **300** holds a probabilistic value or confidence level for the presence an object of interest and a quality of grasp. The above are computed based on a combination of confidence levels obtained from the one or more object detection modules **208** and one or more grasp detection modules **210**. Each location of the representation **300** represents a grasping alterative. For example, if multiple grasp detection modules **210** are employed, the quality of grasp for each location (representing a respective grasping alterative) in the coherent representation **300** is computed based on the grasp scores for the corresponding location predicted by the multiple grasp detection modules. As an example, the quality of grasp for a location (pixel or other defined region) in the coherent representation **300** may be determined as an average or weighted average of the grasp scores computed for that location by the individual grasp detection modules. In some embodiments, multiple grasp detection modules **210** may produce similar grasp scores (indicative of quality of grasp) for a particular grasping location (i.e., grasping alternative), but provide very different approach angles for that grasping alternative. This discrepancy in approach angle would result in lower overall score for that grasp. The HLSF module **212** can either lower the quality of that grasping alternative or provide an additional 'discrepancy' attribute associated to them. The latter approach may be leveraged by the MCDM module **222** to decide whether to penalize high discrepancy grasping alternatives or to accept them. When multiple object detection modules **208** are employed, the HLSF **212** can compute, for each location (pixel or other defined region) in the coherent representation **300**, the probability of the presence of an object of a given class label, for example, using Bayesian inference or similar probabilistic methods. This generalizes to the case where any number of algorithms may be used to produce outputs on the same feature, to achieve redundant information fusion.

[0042] Still referring to FIG. **3**, three grasping alternatives are illustrated, which correspond to locations or cells **302**, **304** and **306** of the coherent representation **300**. Each cell of the representation **300** may represent a single pixel or a larger region defined by multiple pixels. The shown portion of the representation includes an object A (i.e., an object of class label A) and an object B (i.e., an object of class label B). The probability of the presence of an object of class label A in a given cell is denoted as P(A). Likewise, probability of the presence of an object of class label B in a given cell is denoted as P(B). The grasping alternatives corresponding to cells **302** and **304** are determined to be affiliated to object A, based on the computed probability P(A). However, the grasping alternative corresponding to cell **304**, which is closer to the center of the object A, is associated with a higher quality of grasp than that associated with the grasping alternative corresponding to cell **302**. The grasping alternative corresponding to cell **306** has affiliation to multiple objects A and B, based on the computed probabilities P(A) and P(B). In examples, an affiliation of a grasping alterative to a particular object may be determined when the probability P of the presence of that object in the corresponding cell is higher than a threshold value.

[0043] The HLSF module thus computes, for each grasping alterative, attributes that include functional relationships between the grasping alternatives and the detected objects. The attributes for each grasping alternative may comprise, for example, quality of grasp, affiliation to object A, affiliation to object B, discrepancy in approach angles, and so on.

[0044] Referring again to FIG. **2**, based on the grasping alternatives **214** and the respective attributes **216** computed by the HLSF module **212**, a decision is made by the MCDM module (**222**) to select one of the grasping alternatives for execution. As per described embodiments, the MCDM module **222** may rank the grasping alternatives **214** based on multiple criteria that are mapped to the attributes and a respective weight assigned to each criterion. The weights may be determined based on a specified bin picking application objective and one or more specified application constraints.

[0045] The MCDM module **222** may start by setting up a decision matrix as shown in FIG. **4**. In the shown decision matrix **400**, the rows represent possible grasping alternatives A**1**, A**2**, A**3**, etc., while the columns represent criteria C**1**, C**2**, C**3**, C**4**, etc. The criteria are given by attributes of the grasps. Each criterion C**1**, C**2**, C**3**, C**4**, etc. is associated with a respective weight W**1**, W**2**, W**3**, W**4** etc. Criteria examples include 'affiliation to object A', 'affiliation to object B', predicted grasp quality, robotic path distance, etc. For each grasping alternative, the weighted score pertaining to multiple criteria is computed by the MCDM module **222**. In FIG. **4**, the scores for grasping alternative A**1** that pertain to criteria C**1**, C**2**, C**3** and C**4** are respectively indicated as a**11**, a**12**, a**13** and a**14** respectively; the scores for grasping alternative A**2** that pertain to criteria C**1**, C**2**, C**3** and C**4** are respectively indicated as a**21**, a**22**, a**23** and a**24**; and so on. The MCDM module **222** then ranks the grasping alternatives based on the weighted scores on multiple criteria and selects the optimal grasping alternative given a current application objective (e.g., grasp objects of class A and class C only, preference to pick objects with the smallest robotic path distance, preference to pick objects with high quality of grasp even in spite of long travel distances, etc.) and application constraints (e.g., workspace boundaries, hardware of the robot, grasping modality such as suction, pinching, etc.). For this purpose, starting from the decision matrix shown in FIG. **4**, one or more of several MCDM techniques may be used to arrive at the final decision. Examples of known MCDM techniques suitable for the present purpose include simple techniques such as Weight Sum Model (WSM) and Weighted Product Model (WPM), or sophisticated techniques such as Analytic Hierarchy Process (AHP) as well as the ELECTRE and TOPSIS methods.

[0046] In some embodiments, infeasible grasping alternatives as per the bin picking application may be removed from the decision matrix prior to the implementation of the MCDM solution in order to improve computational efficiency. Examples of infeasible grasping alternatives include grasps whose execution can lead to collision, grasps having multiple object affiliations, among others. In different instances, this constraint-based elimination procedure of candidate grasps may be performed in an automated manner at different stages of the process flow in FIG. **2**, such as by the individual detection modules, the HLSF module or at the MCDM solution stage.

[0047] Continuing with reference to FIG. **2**, the MCDM module **222** outputs an action **224** defined by the selected grasping alternative arrived at by any of the techniques mentioned above. Based on the output action **224**, executable code is generated, which may be sent to a robot controller to operate the robot to selectively grasp an object from the bin. The selective grasping can include grasping an object of a specified type from a pile of assorted objects in the bin.

[0048] The importance weights of the MCDM module **222** can be set manually by an expert based on the bin picking application. For example, the robotic path distance may not be as important as the quality of grasp if the overall grasps per hour should be maximized. In some embodiments, an initial weight may be assigned to each of the criteria of MCDM module (e.g., by an expert), the weights being subsequently adjusted based on feedback from simulation or real-world execution of consecutive instances of the autonomous bin picking. This approach is particular suitable in many bin picking applications where, while some importance weights are clear or binary (e.g., solutions that can lead to collisions should be excluded), others are only known approximately (e.g., path distance ~0.2 and grasp quality ~0.3). Therefore, rather than fixing all importance weights a priori, the expert can define ranges and initial values where the parameters are permitted. The MCDM module **222** can then fine-tune the parameters using the experience from either simulation experiments or the real world itself. For example, based on the success criteria of the current action (e.g., overall grasps per hour), the system can probabilistically at random change the settings in the permitted ranges. More specifically, if the robotic path distance p is defined [0.1 0.3] and grasp quality q is defined [0.2 0.4] then after the first iteration with settings p=0.2 and q=0.3 the system may try again with p=0.21 and q=0.29. If the success criteria are more accurately fulfilled with the new settings than the original settings, the new settings are used as the origin for the next optimization step. If this is not the case, then the original setting remains as origin for the next instance of execution of bin picking. In this way, the MCDM module **222** can fine-tune the settings iteratively to optimize a criterion based on the real results from the application at hand.

[0049] The proposed methodology of combining HLSF and MCDM methodologies may also be applied to a scenario where semantic recognition of objects in the bin is not necessary. An example of such a scenario is a bin picking application involving only objects of the same type placed in a bin. In this case, there is no requirement for an object detection module. However, the method may utilize multiple grasp detection modules. The multiple grasp detection modules may comprise multiple different neural networks or may comprise multiple instances of the same neural network. The multiple grasp detection modules are each fed with a respective image captured by a different sensor. Each sensor may be configured to define a depth map of the physical environment. Example sensors include depth sensors, RGB-D cameras, point cloud sensors, among others. The multiple different sensors may be associated with different capabilities or accuracies, or different vendors, or different views of the scene, or any combinations of the above. The multiple grasp detection modules produce multiple outputs based on the respective input image, each output defining a plurality of grasping alternatives that correspond to a plurality of locations in the respective input image. The HLSF module, in this case, combines the outputs of the multiple grasp detection modules to compute attributes (e.g., quality of grasp) for the grasping alternatives. The MCDM module ranks the grasping alternatives based on the computed attributes to select one of the grasping alternatives for execution. The MCDM module outputs an action defined by the selected grasping alternative, based on which executable instructions are generated to operate a controllable device such as a robot to grasp an object from the bin.

[0050] Similar to the previously described embodiments, the grasp neural networks in the present embodiment may each be trained to produce an output vector that includes a plurality of predicted grasp scores associated with various locations in the respective input image, the grasp scores indicating a quality of grasp at the respective location. For example, the output of a grasp neural network may indicate, for each location (e.g., a pixel or other defined region) in the respective input image, a predicted grasp score. Each location or grasping point represents a grasping alternative which may be used to execute a grasp with a predicted confidence for success. The grasp neural network may define, for each grasping alternative, a grasp parametrization that may consist of the location or grasping point (e.g. x, y, and z coordinates) and an approach direction for the grasp, along with a grasp score. In some embodiments, the grasp neural networks may comprise off-the-shelf neural networks which have been validated and tested in similar applications.

[0051] Furthermore, similar to the previously described embodiments, the HLSF module may align the outputs of the multiple grasp detection modules to a common coordinate system to generate a coherent representation of the physical environment, and compute, for each location in the coherent representation, a probabilistic value for a quality of grasp. The quality of grasp for each location (representing a respective grasping alterative) in the coherent representation is computed based on the grasp scores for the corresponding location predicted by the multiple grasp detection modules. As an example, the quality of grasp for a location (pixel or other defined region) in the coherent representation may be determined as an average or weighted average of the grasp scores computed for that location by the individual grasp detection modules. In some embodiments, multiple grasp detection modules may produce similar grasp scores (indicative of quality of grasp) for a particular grasping location (i.e., grasping alternative), but provide very different approach angles for that grasping alternative. This discrepancy in approach angle would result in lower overall score for that grasp. The HLSF module can either lower the quality of that grasping alternative or provide an additional 'discrepancy' attribute associated to them. The latter approach may be leveraged by the MCDM module to decide whether to penalize high discrepancy grasping alternatives or to accept them.

[0052] The MCDM module may rank the grasping alternatives computed by the HLSF module based on multiple criteria that are mapped to the attributes and a respective weight assigned to each criterion, the weights being determined based on a specified bin picking objective and one or more specified constraints. To that end, the MCDM module