Towards Coordinated Precision Assembly with Robot Teams

Mehmet Dogar, Ross A. Knepper, Andrew Spielberg, Changhyun Choi, Henrik I. Christensen and Daniela Rus

Abstract We present a system in which a flexible team of robots coordinates to assemble large, complex, and diverse structures autonomously. Our system operates across a wide range of spatial scales and tolerances, using a hierarchical perception architecture. For the successful execution of very precise assembly operations under initial uncertainty, our system starts with high-field of view but low accuracy sensors, and gradually uses low field-of-view but high accuracy sensors. Our system also uses a failure detection and recovery system, integrated with this hierarchical perception architecture: upon losing track of a feature, our system retracts to using high-field of view systems to re-localize. Additionally, we contribute manipulation skills and tools necessary to assemble large structures with high precision. First, the team of robots coordinates to transport large assembly parts which are too heavy for a single robot to carry. Second, we develop a new tool which is capable of co-localizing holes and fasteners for robust insertion and fastening. We present real robot experiments where we measure the contribution of the hierarchical perception and failure recovery approach to the robustness of our system. We also present

M. Dogar, R.A. Knepper and A. Spielberg contributed equally to this paper.

M. Dogar (✉) · R.A. Knepper · A. Spielberg · D. Rus
Computer Science and Artificial Intelligence Lab, MIT, Cambridge, USA
e-mail: mdogar@csail.mit.edu

R.A. Knepper
e-mail: rak@csail.mit.edu

A. Spielberg
e-mail: aespielberg@csail.mit.edu

D. Rus
e-mail: rus@csail.mit.edu

C. Choi · H.I. Christensen
Institute for Robotics and Intelligent Machines, College of Computing,
Georgia Tech, Atlanta, Georgia
e-mail: cchoi@cc.gatech.edu

H.I. Christensen
e-mail: hic@cc.gatech.edu

© Springer International Publishing Switzerland 2016

M.A. Hsieh et al. (eds.), *Experimental Robotics*, Springer Tracts
in Advanced Robotics 109, DOI 10.1007/978-3-319-23778-7_43

an extensive set of experiments where our robots successfully insert all 80 of the attempted fastener insertion operations.

Keywords Robotic assembly · Robotic manufacturing · Robot teams · Distributed control · Precision assembly · Perception

1 Introduction

Manufacturing systems of today have very limited flexibility, often requiring months of fine-tuning before an industrial assembly line is ready for production. We envision the manufacturing systems of the future, in which agile, flexible teams of mobile robots coordinate to assemble complex and diverse structures autonomously. Here, we define flexibility as the ability for robots to change tasks, factory floors to be reconfigured, and similar parts to be interchanged without reprogramming the system.

This approach has the potential to meet the demands of modern production: ever-shortening product life-cycles, customized production, and efficiency [2]. In this paper we present a significant step in this direction through an exemplar task which requires a heterogeneous team of four robots with different skills to align and fasten a panel to a corresponding box. We outline the task below (see Fig. 1 for details):

- A robot specializing in fine perception and manipulation localizes a hole on the box.
- A fleet of two robots lift and rotate the panel which would be too heavy for a single robot to manipulate single-handedly.
- Using a robot specializing in coarse perception for guidance, the fleet aligns the panel to the box.
- Using the fine perception/manipulation robot for guidance, the fleet aligns a hole on the panel with a hole on the box.
- The fine perception/manipulation robot inserts the first fastener.
- Using the bounding geometry of the box, one of the fleet robots aligns the panel with the box.
- Finally, with all four holes aligned, the fine perception/manipulation robot inserts the three remaining fasteners into the remaining holes.

In particular, we present the following contributions:

1. A hierarchical perception system formalized in the context of integrated perception and manipulation over changing task scales and scopes.
2. A failure recovery approach which allows us to re-seed search and tracking procedures.
3. A simple but robust controller for the collaborative manipulation of objects whose manipulations are outside the physical limits of single individual robots.
4. The demonstration of a new rigid LIDAR and fastener tool which allows for simultaneous localization and fastener insertion within the same coordinate frame.

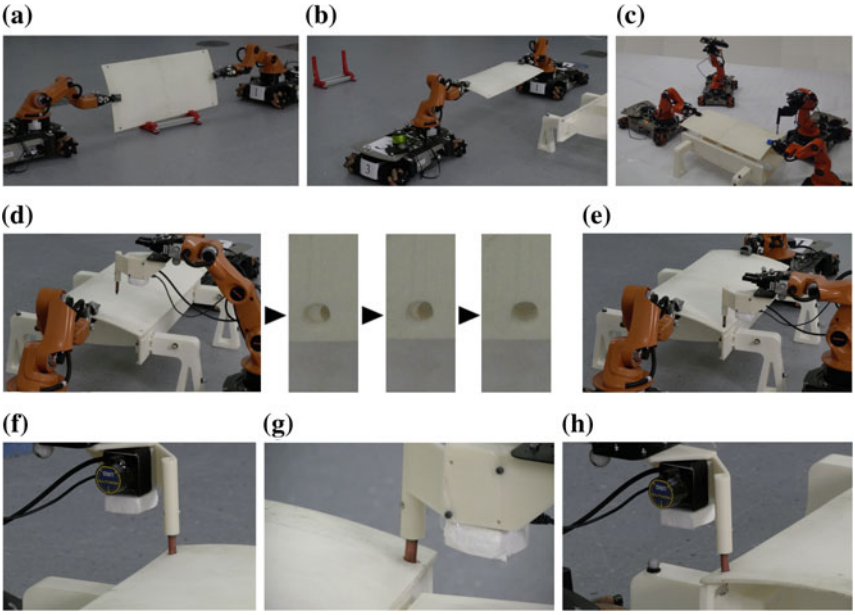

Fig. 1 Assembly tasks involve large-scale operations such as transport and fine manipulation operations such as hole alignment and fastener insertion. **a** Locate/grasp parts, **b** transport of parts, **c** part alignment, **d** hole alignment, **e** Fastener insertion, **f** Fastener 2, **g** Fastener 3, **h** Fastener 4

Hierarchical approach Autonomous manufacturing systems must operate across a wide range of spatial scales and tolerances. Consider the task of attaching a panel to a complex assembly. First, the robots move the panel from a storage rack to the assembly site (Fig. 1a–c). Second, the robots insert fasteners to attach the panel to the assembly (Fig. 1d–h). The first task requires perception and control at a spatial scale which captures the parts and sometimes the whole factory floor and tolerates relatively large errors in positioning. The second task requires fine perception and control with much tighter tolerances. With existing technologies, no monolithic perception and control approach solves both problems. In this work, we contribute a hierarchical approach in which different layers of localization and control systems interact to satisfy the continuously changing scale and precision requirements. See Table 1 for an example flow of control across the levels of the hierarchy.

Failure recovery By exploiting our system's hierarchical perception formalization we also introduce a failure recovery system. We present systems which can determine when insufficient precision has been obtained. Our system allows us to move freely between adjacent levels in the perception hierarchy, allowing us to re-seed failed searches and tracking procedures with better initial guesses. This allows us to avoid lengthy searches in the absence of useful feature information by falling back to estimates which are coarser but larger in scope. Such a system is applied to hole alignment but could also be applied to a number of other manipulation tasks in

Table 1 Flow of actions among four robots during attachment of a panel to a box

Robot			
R1	R2	R3	R4
			Move to hole 1 neighborhood
Navigate to and move gripper to panel		Localize box	Find hole 1 in box
Close grippers and form fleet			Find hole 1 in box
Pick up panel			
Orient panel to horizontal			
Transport panel into neighborhood of box			
Servo panel into alignment with box		Localize panel	
Servo panel hole 1 into alignment with box hole 1			Localize panel hole 1
End fleet formation and open grippers			Insert fastener 1
Move out of the way	Align panel hole 2 to box hole 2	Move out of the way	Navigate to panel hole 2
	Move out of the way		Localize hole 2
			Insert fastener 2
			Navigate to hole 3
			Localize hole 3
			Insert fastener 3
			Navigate to hole 4
			Localize hole 4
			Insert fastener 4

Time flows from top to bottom. Box colors indicate the type of localization used in each action. Blue boxes indicate fiducial-based localization. Green boxes denote object-shape-based tracking. Pink boxes indicate functional-feature level localization. White boxes indicate sensorless operations

other systems which involve active perception and estimation, including precision grasping and collision-free navigation of cluttered factory environments.

Collaborative manipulation A team of robots working in a factory requires coordination and collaboration. The coordination can be loosely coupled, as in collision-free navigation, or tightly-coupled, as when carrying a large part (Fig. 1b) as a team. Our system displays coordination between robots at these various levels.

Specialized tools for robotic manipulation An important challenge in flexible factory automation is enabling fine manipulation skills, e.g. inserting a fastener or screwing a nut. Much like human workers, robots need specialized tools and skills (control algorithms) to perform these operations to specifications. We've developed such a tool (Fig. 3) to unify sensing and actuation in the tool frame, thus delivering high precision, as suggested in our second listed contribution.

Our approach is holistic: we are interested in the challenges and questions that a complete system raises. The literature has approached the underlying problems sep-

arately. Many methods have been proposed for collaborative manipulation/transport of objects by a team of robots [9, 12, 15, 23, 25]. Particularly, Desai and Kumar [9] propose a motion planning approach for a team of robots transporting an object among obstacles; and Khatib et al. [12] present a decentralized control framework for the manipulation of an object with a system of multiple manipulators. Similar approaches have been applied to the factory floor [14, 20] where a team of robots transports an object with the help of human input. We present a system where the team of robots transports an object in the context of a complex task. To do this, they must form a fleet, and maintain specified relative arm configurations while making progress toward goal positions. We develop control algorithms which treat fleets (connected by manipulated objects) as rigid movable bodies and are able to correct for erroneous deviations. Our control/perception environment is not structured specifically for a transport task, but is generic enough to accommodate other assembly tasks.

One generic and important assembly operation is fastening multiple parts together. In our system this is achieved by inserting fasteners through holes on the parts. This operation, sometimes called *peg-in-hole* in the literature, has been studied extensively. One approach to this problem is to use a hybrid force-position control [17, 19], which, through force sensing and compliant motion [11], enables a manipulator to slide along surfaces. Combined with a principled approach to dealing with uncertainty [16], a high-precision operation such as peg-in-hole can be accomplished through a set of *guarded-moves*. This approach, however, may not be feasible if the assembly parts are very sensitive and prone to scratching. In our implementation we avoid making forceful interaction with the surfaces of assembly parts. Instead of a series of guarded moves, we use extensive and high-accuracy sensor readings to localize the hole, and a compliant shape for the fastener tip to account for any remaining inaccuracy in localization.

Robotic perception literature and technology provide a rich set of tools [4, 6, 21] which can be used for certain tasks in the factory setting. While these systems work best when the object is closer than a few meters, the accuracy drops as the object gets too far or too close. In addition, visual perception is highly challenged in many cases: occlusions, cluttered backgrounds, and image blurring because of fast motions either in objects or camera. To overcome these limitations of visual perception, it is often combined with motion estimation [13] or tactile sensing [1, 10]. Skotheim et al. [22] use functional feature detection for low-level industrial manipulation. Although the literature provides these powerful tools, any single one is insufficient to overcome the challenges of flexible factory environments.

2 Hierarchical Localization and Control Approach

Various objects and features of a flexible factory environment require various perception and control technologies and a smooth integration among them. We present a three-tiered perception and control structure, comprising fiducial-based, object-shape-based, and functional-feature-based approaches.

Fiducial-based technology tracks non-production parts, part sources, and robots, using a motion capture system like Vicon.[1] Motion capture provides highly accurate, sub-centimeter localization accuracy, but it is restricted to tracking parts to which external fiducials may be affixed. For many production parts, attaching fiducials is undesirable and impractical. Furthermore, occlusion can be a problem. Thus, complementary localization methods are needed.

Object-shape-based tracking is implemented as a particle filtering approach using an RGB-D camera [5]. 3D mesh models of production parts are known a priori, and three visual features—colors, depth points, and normals—are used to calculate the likelihood of each particle hypothesis with respect to the current RGB-D scene. Our system localizes the box and panel from a single RGB-D camera. The robot carrying the camera can be seen in Fig. 1c. The system may exploit the freedom of the camera's point of view to avoid occlusion.

Functional-feature-based tracking for hole alignment and insertion is the most demanding part of our task as it requires very high-precision coordination among multiple robots. We use a coordinated control procedure along with a specialized tool, explained in the next section.

We hypothesize that without the use of all three levels in the sensing and control hierarchy, the system cannot achieve robust fastener insertion. In the rest of this section, we discuss the levels of the hierarchy and how the robots may smoothly transition up and down through them.

2.1 Sequential Composition of Sensors

The *funnel* analogy has long served in robotics literature to represent the act of reducing uncertainty or error in the configuration of an object. Mason [18] first introduced the concept in the context of performing sensorless manipulation actions that employ passive mechanics to reduce part uncertainty. Later, Burridge et al. [3] applied the funnel analogy to feedback control in the form of sequential composition of controllers, spawning much follow-on work [7, 8, 24]. This body of work is sensor-agnostic in that the type and quality of sensor data is assumed to be homogeneous throughout the configuration space.

A contribution of this paper is sequential composition of sensors used for localization. Each sensor operates over some capture volume, or scope, which is the top of the funnel. Within the scope, it delivers to the robot a pose estimate that reduces uncertainty with some accuracy, which is the bottom of the funnel. Each of the localization technologies we employ imposes errors that limit accuracy in three categories: (1) *sensor error*, (2) *indirection error* and (3) *semantic calibration error*. Sensor error, the accuracy claimed by the sensor manufacturer, is typically the smallest contribution to overall error in performing localization.

[1] http://www.vicon.com/.

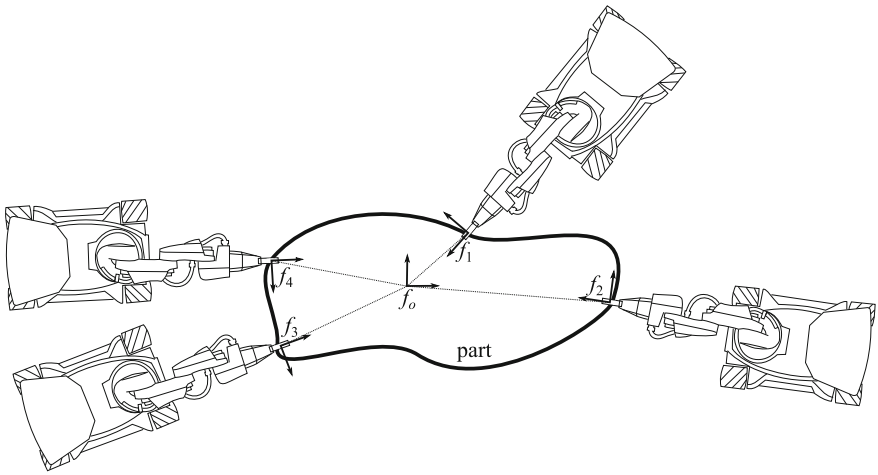

Fig. 2 Through fleet control, an arbitrary number of robots collaboratively carry a part in an arbitrary shape formation. Individual robot motions are computed with respect to a commanded twist at the fleet origin, o. Each robot n maintains the pose of the fleet origin in its own local coordinate frame, f_n, so there is no need for a global reference. The algorithm is fully distributed

Indirection error stems from the fact that sensors rarely localize the desired coordinate frame directly. Instead, they sense some set of features, each with some transform to the desired frame. This indirection leads to two sources of error: (1) small errors in orientation are magnified by translation, and (2) the feature poses may not be well calibrated to the desired frame. All three localization technologies exhibit indirection error. Since fiducials cannot be applied directly to the part being assembled, the robot hands must be tracked instead with fiducial-based technology. The position of each hand on the object may not be well known (as in Fig. 2). Since the hands grasp the perimeter of the object, the indirection error of fiducial-based methods is proportional to the size of the part, making them the coarsest level of the hierarchy. In the case of object-shape-based tracking, a point cloud over a smooth panel surface (as in Fig. 1) produces substantial ambiguity about the location of each point on the object and hence of the the object origin. It is the indirection error that the particle filter strives to minimize. Finally, the functional-feature-based hole detector tracks the hole's circumference, whereas the center of the hole is desired. In the case of a circular hole, the resulting indirection error is minimal due to symmetry.

Finally, semantic calibration error originates from the fact that a perception model used for localization must be calibrated against the semantic model used for manipulation. For example, fiducials placed on the robot for motion capture must be manually calibrated to the robot's pose. Similarly, for object-shape-based tracking, the origin and shape of the CAD model of the tracked object may not match the origin and shape of the physical object. The functional-feature-based hole tracker has no semantic calibration error because the sensor directly tracks a semantic feature.

Given a position estimate of the object with uncertainty, it may be within scope of several sensors, giving the system some flexibility in which technology to use (see

Table 2 Order of magnitude of sensor capabilities and of errors induced by the usage model

Approach	Sensor	Scope (m^3)	Error (m)			Net accuracy (m^3)
			Sensor	Indirection	Semantic calib.	
Fiducial-based	Vicon	10^2	10^{-3}	10^{-1}	10^{-2}	10^{-3}
Object-shape-based	Kinect	10^0	10^{-2}	10^{-2}	10^{-2}	10^{-6}
Functional-feature-based	Hokuyo	10^{-2}	10^{-3}	10^{-3}	0	10^{-9}

See the text for a description of error sources. Net accuracy is the volume resulting from the sum of the three distance errors

Table 2 for a summary of sensor capabilities). This flexibility allows the system to be tolerant to effects such as occlusion or communication drop-outs. The typical progression of the localized feedback control system is to servo the object into position at increasingly detailed scales.

2.2 Failure Recovery

Failures in execution can happen at any step of the assembly operation. To make sure that the assembly operation completes successfully, our system detects and tries to recover from failures.

The hierarchical perception/control structure provides the backbone of our failure recovery approach. During successful execution, the control is handed-off from higher levels to the lower levels: higher levels perform coarse localization and lower levels perform precise tasks. Failure recovery is implemented as the inverse process, where the control is handed off from lower levels to higher levels: lower levels of perception are precise in tracking objects/features but have limited scope, which may result in the tracked objects/features getting lost. In such a case the control is handed-off to the higher level for a coarse but larger scope localization.

A crucial example of the failure recovery process occurs during alignment of the panel-hole with the box-hole. To accomplish this task, the panel is first aligned with the box using the object-shape-based perception system, which has a large scope but low accuracy. Once the panel is coarsely aligned with the box, the functional-feature-based localizer takes over to track the panel-hole and align it with the box-hole. This localizer has high accuracy but a small scope. The scanner occasionally loses track of the hole due to the small scope and the noise in the arm and base motions of the robots during alignment. In such a case, the system reverts back to the previous level, the object-shape-based alignment. The larger scope re-aligns the panel with the box

and hands over the control to the functional-feature-based tracker once more. This process continues until this sensor successfully tracks the panel-hole and aligns it with the box-hole.

This approach to detecting and recovering from failure provides significant robustness to our system. Even if the individual layers permit failure, the overall architecture displays very high robustness as long as failures are detected and the system is started from a recoverable state.

2.3 Fleet Control

For collaborative transport of large parts, the robots perform a distributed, collective behavior inspired by human group behavior using force feedback and observation of others. In fleet control mode, the robots maintain a fixed formation of arbitrary shape while holding an object, as in Fig. 2.

Initially, each robot separately moves into formation by grasping the object at an appropriate location. Robot n's pose, f_n is measured at this grasp point because the other robots can readily localize its hand. Formation control initializes via a synchronization broadcast message. Upon initialization, the robots compute a common reference origin f_o for the object. Robot n represents the fleet origin in its own frame as $T_{f_n}^{f_o}$. The position of the origin defaults to the mean of all robot hand positions, and its orientation initializes to that of the global coordinate frame (i.e. Vicon frame). Henceforth, the global frame is not needed as all coordinates are given in f_o or f_n. If desired, f_o can be moved with respect to the fleet.

Group motions are commanded as a twist w specified in frame f_o. Each robot computes its own hand motion in order to comply with w in six degrees of freedom (DoFs). Hand motions are achieved through base motion when possible (X, Y, yaw) and arm motion otherwise (Z, roll, pitch). It should be noted, however, that the KUKA youBot cannot achieve full six DoF motion due to their arm kinematics. Therefore, the task presented in this paper involves only five DoF object manipulation.

An important function of the fleet controller is to maintain a stable fleet formation. Any position error introduced by group motion will cause the fleet origin to drift away from its target pose in the frame of the robots. A PD controller introduces correction terms to the body and arm motions in order to maintain the correct fleet formation.

Similarly, force exchange among the robots through the object can indicate an error in desired position. The robots' arm joints perform PD velocity control on joint angle. In the steady state, an error derived from the joint torques can be attributed to a combination of gravity and an error in the fleet formation. Thus, the robot has detected a resultant force from the combined motion of the rest of the fleet. In response to this force, the fleet controller applies a correction term to $T_{f_n}^{f_o}$.

Since each robot computes a motion consistent with the fleet twist command, any residual force results from an error in the formation, which may have two causes. First, the robot may drift slightly out of formation while carrying a rigid object. Second, the object may be somewhat deformable. Although the fleet cannot deliberately exploit

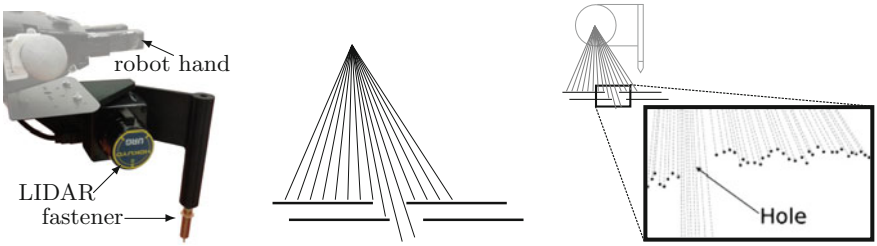

Fig. 3 *Left* Hole alignment and insertion tool. *Center* Alignment of two holes is achieved by estimating the width of the opening. *Right* Example real data used to estimate the width of the opening

deformability of material, it will accommodate deformations induced by the carrying operation by slightly varying the formation in response to these joint torques.

2.4 Coordinated Mating of Holes and Fastener Insertion

To achieve millimeter-scale accuracy, we employ a custom-built end-effector tool on which both a Hokuyo LIDAR and a fastener are rigidly affixed (Fig. 3-left). This sensor fulfills the functional-feature-based localization in the hierarchy.

Algorithm 1 Coordinated alignment of holes

```
1: function AlignHoles
2:     while hole-width < threshold do
3:         twist ← FastenerRobot.DesiredPartMotion(history)
4:         Fleet.MovePart(twist)
5:         hole-width ← FastenerRobot.EstimateHoleWidth()
6:         history.Add(hole-width)
7: function Fleet.MovePart(twist)
8:     for each robot in fleet do
9:         pose ← robot.ComputePoseRelativeToPart()
10:         robot-twist ← Transform(twist,pose)
11:         Robot.Move(robot-twist)
```

We present the collaborative procedure by which our system aligns the holes of two different parts in Algorithm 1. This procedure is executed after the robot with the fastener locates the hole on one of the parts (the box, in our example) and the fleet of robots brings the panel into the vicinity using the object-level tracking.

The goal in Algorithm 1 is to achieve an alignment within the tolerance required by the fastener. At each step the robot with the tool estimates (line 5) the alignment of the two holes (Fig. 3-center) by measuring the width of the opening (Fig. 3-right). If the opening is not large enough (line 2), the fastener robot commands a new velocity twist for the moving part (lines 3–4). In computing this, the fastener robot can use the

history of readings to maximize the alignment using gradient ascent. We implement this by making the fleet follow a series of waypoints.

A twist for the moving part commands the robots in the fleet to move (lines 7–11) using decentralized fleet control. After the holes are aligned, the fastener can be inserted. The fastener is placed directly in line with the LIDAR's laser scan, thus allowing the robot to know exactly where the fastener is with respect to a detected hole at all times, and to bring the fastener over the hole.

3 Experiments

We use a team of four KUKA Youbots for our experiments. These robots are tasked with assembling a panel (Fig. 4a) on a box (Fig. 4b) using fasteners (Fig. 4c). The panel and box are initially placed on supporting racks, which have markers for the fiducial-based Vicon tracking system. Two of the robots, R1 and R2, are responsible for the manipulation of the panel. Robot R3 carries a Kinect RGB-D camera which performs the object-shape-based tracking of the panel and the box. Robot R4 carries the insertion tool (Fig. 3-left). The insertion tool has an integrated Hokuyo laser scanner which performs the functional-feature-based alignment with the holes on the box and the panel.

We measure the effectiveness of different components of our perception and control hierarchy by running experiments with three different configurations of this system:

1. *Fiducial-based + Object-shape-based (FO):* In this case, the panel and box are aligned only using the object-shape-based tracking and control. The functional-feature-based tracking, i.e. the Hokuyo laser scanner is *not* used.

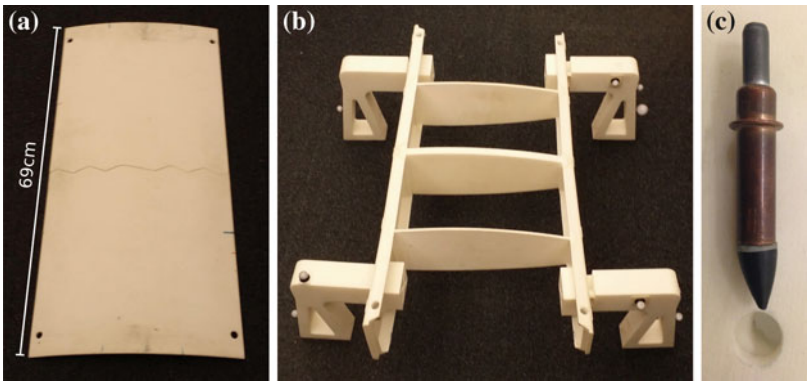

Fig. 4 Assembly parts used in our experiments. **a** Panel. **b** Box. **c** A fastener and hole (misaligned) as used in this task. The fastener is an adapted cleco. The holes were drilled to permit a cleco to fit up to the flange with a tolerance of 1.5 mm

2. *Fiducial-based + Functional-feature-based (FF):* In this case, the object-shape-based tracking of the panel and box is left out, i.e. the Kinect RGB-D sensor is *not* used. Instead, the robots remember their grasping configuration of the panel and assume it does not change relative to the robot hands during the course of the task.
3. *Fiducial-based + Object-shape-based + Functional-feature-based (FOF):* Our system as described in Sect. 2 where the objects are tracked using the Kinect RGB-D camera and the hole is aligned using the Hokuyo laser scanner.

With our system we performed two sets of experiments. First, we ran our system in the FOF configuration 22 times to measure the robustness, the contribution of our failure recovery system to the robustness, and the overall speed of our system. A video of one such run is available at: http://youtu.be/cmJTsyIgCRo.

Second, we performed experiments to measure the contribution of the hierarchical perception architecture to the robustness of our system. In this set of experiments we created perturbations to the pose of the panel as it was being carried. Under these perturbations we ran our system four times in each of the FO, FF, and FOF configurations, totaling to twelve more runs.

4 Results

We start with reporting the results of 22 experiments in the FOF configuration. Our system showed a remarkable robustness for such a complicated and long task. Aside from two hardware failures of unknown cause, the system succeeded 20 out of 20 times. Table 3 shows the average time of 20 successful runs along with the minimum and maximum durations. The first column shows the time spent for localizing the four holes on the assembly during each run. The second column shows the time spent during aligning the panel to the box using the object-based tracking system. The last column shows the execution time for the complete assembly operation.

The first set of experiments also showed the important contribution of failure recovery to the robustness of our system. In 20 % of panel alignment attempts the two holes were not aligned precisely, which resulted in failure recovery getting triggered. After failure recovery the holes were aligned and the fasteners were successfully inserted. During these experiments our system attempted 80 fastener insertions and succeeded in all of them.

Table 3 Execution times

	Hole localization	Ladder-panel alignment	Total
Mean time (s)	92	37	679
Min time (s)	27	17	569
Max time (s)	259	141	849

Table 4 Comparison of the performance of different configurations of our system

	Success	Notes
FO	1/4	Successful run scratched panel surface on 2 of the 4 holes
FF	2/4	Panel hole search timed out at 10 min
FOF	3/4	All succeeded for hole alignment but one failed during fastener insertion

We report the result of our second set of experiments in Table 4. Here we perturb the position of the grasped panel to measure the robustness of our system. The first two cases show the system running with certain layers of the hierarchical perception system removed. In these cases the system was not able to get precise alignment between the holes of the panel and the box. The full hierarchical perception system was able to get precise alignment between the holes in all four cases, but had trouble with the insertion of the fastener since the insertion routine was not adaptive to the changed height of the panel due to the perturbation. However our full system was robust in achieving the precise hole alignment.

5 Insights and Conclusion

The results show that intelligent use of a hierarchical perception system can greatly improve the robustness of a manufacturing system to be nearly perfect. The system is not only able to perform collaborative carrying, precise alignment, and collision-free insertion, but is also able to detect and fix the rare errors in alignment. Further, the only failures were in the cases of high-torque-driven arm failures, in which the system failed in the collaborative carry step. In addition, we have demonstrated that use of object-shape-based tracking makes the system robust to outside perturbations or other internal errors that could lead to poor grasps.

Robustness is a key attribute for maximizing productivity in manufacturing. Traditional factory robots are bolted to the floor, thus achieving sensorless high precision through kinematics. Modern factory automation processes eliminate uncertainty through careful, time-consuming human design. Product changes require re-engineering of the process, contributing to a lack of versatility. Instead, we present a flexible system which achieved alignment within tolerance in 100 % of trials, but the fastener motion caused a failure in one of the perturbed cases.

We have identified several avenues for improving fastener insertion. Most importantly, torque feedback at the fastener tip flags an insertion failure. Soft guard materials around the fastener might help to avoid damaging fragile parts.

Our experiments validated our hypothesis that a hierarchical sensing system improves robustness in assembly. Removal of either the functional-feature-based or the object-shape-based localization from the hierarchy substantially diminished the successful completion performance of the system. With all three sensing sys-

tems, any gaps in perception by one sensor can be filled by one of the other sensors, allowing for a smooth transition among operating scales.

As many assembly procedures are composed of successive individual steps each of which must succeed, identifying and recovering from failures is crucial. A single failed step either requires a method of recovery failure or requires a restart of the procedure. Potential failure modes of the system include: misalignment of the two holes and fastener, dropping the panel prematurely, and incorrectly tracking the panel or features. We implemented automatic failure detection and handling algorithms for many of these problems and have designed the system to minimize the incidence of failure.

Acknowledgments This work was supported by The Boeing Company. We are grateful for their support.

References

1. Allen, P.K.: Integrating vision and touch for object recognition tasks. Int. J. Robot. Res. **7**(6), 15–33 (1988)
2. Bourne, D.: My boss the robot. Sci. Am. **308**(5), 38–41 (2013)
3. Burridge, R.R., Rizzi, A.A., Koditschek, D.E.: Sequential composition of dynamically dexterous robot behaviors. Int. J. Robot. Res. **18**(6), 534–555 (1999)
4. Choi, C., Christensen, H.I.: Robust 3D visual tracking using particle filtering on the Special Euclidean group: a combined approach of keypoint and edge features. Int. J. Robot. Res. **31**(4), 498–519 (2012)
5. Choi, C., Christensen, H.I.: RGB-D object tracking: a particle filter approach on GPU. In: Proceedings of the IEEE International Conference on Intelligent Robots and Systems, pp. 1084–1091 (2013)
6. Collet, A., Martinez, M., Srinivasa, S.S.: The moped framework: object recognition and pose estimation for manipulation. Int. J. Robot. Res. **30**(10), 1284–1306 (2011)
7. Conner, D.C., Rizzi, A.A., Choset, H.: Composition of local potential functions for global robot control and navigation. In: Proceedings of the IEEE International Conference on Intelligent Robots and Systems, vol. 4, pp. 3546–3551. IEEE (2003)
8. Das, A.K., Fierro, R., Kumar, V., Ostrowski, J.P., Spletzer, J., Taylor, C.J.: A vision-based formation control framework. IEEE Trans. Robot. Autom. **18**(5), 813–825 (2002)
9. Desai, J.P., Kumar, V.: Motion planning for cooperating mobile manipulators. J. Robot. Syst. **16**(10), 557–579 (1999)
10. Ilonen, J., Bohg, J., Kyrki, V.: Fusing visual and tactile sensing for 3-D object reconstruction while grasping. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 3547–3554 (2013)
11. Inoue, H.: Force feedback in precise assembly tasks. Technical report, DTIC Document (1974)
12. Khatib, O., Yokoi, K., Chang, K., Ruspini, D., Holmberg, R., Casal, A.: Coordination and decentralized cooperation of multiple mobile manipulators. J. Robot. Syst. **13**(11), 755–764 (1996)
13. Klein, G., Drummond, T.: Tightly integrated sensor fusion for robust visual tracking. Image Vis. Comput. **22**(10), 769–776 (2004)
14. Lenz, C., Nair, S., Rickert, M., Knoll, A., Rosel, W., Gast, J., Bannat, A., Wallhoff, F.: Joint-action for humans and industrial robots for assembly tasks. In: RO-MAN, pp. 130–135 (2008)
15. Li, Z., Ge, S.S., Wang, Z.: Robust adaptive control of coordinated multiple mobile manipulators. Mechatronics **18**(5–6), 239–250 (2008)

16. Lozano-Perez, T., Mason, M.T., Taylor, R.H.: Automatic synthesis of fine-motion strategies for robots. Int. J. Robot. Res. **3**(1), 3–24 (1984)
17. Mason, M.T.: Compliance and force control for computer controlled manipulators. IEEE Trans. Syst. Man Cybern. **11**(6), 418–432 (1981)
18. Mason, M.T.: The mechanics of manipulation. In: Proceedings of the IEEE International Conference on Robotics and Automation, vol. 2, pp. 544–548. IEEE (1985)
19. Raibert, M.H., Craig, J.J.: Hybrid position/force control of manipulators. J. Dyn. Syst. Meas. Control **103**(2), 126–133 (1981)
20. Reinhart, G., Zaidan, S.: A generic framework for workpiece-based programming of cooperating industrial robots. In: ICMA, pp. 37–42 (2009)
21. Rusu, R.B., Bradski, G., Thibaux, R., Hsu, J.: Fast 3d recognition and pose using the viewpoint feature histogram. In: IROS, pp. 2155–2162 (2010)
22. Skotheim, Ø., Nygaard, J.O., Thielemann, J., Vollset, T.: A flexible 3d vision system based on structured light for in-line product inspection. Electr. Imaging **2008**, 680505–680505 (2008)
23. Sugar, T.G., Kumar, V.: Control of cooperating mobile manipulators. IEEE Trans. Robot. Autom. **18**(1), 94–103 (2002)
24. Tedrake, R., Manchester, I.R., Tobenkin, M., Roberts, J.W.: Lqr-trees: feedback motion planning via sums-of-squares verification. Int. J. Robot. Res. **29**(8), 1038–1052 (2010)
25. Yamashita, A., Arai, T., Ota, J., Asama, H.: Motion planning of multiple mobile robots for cooperative manipulation and transportation. IEEE Trans. Robot. Autom. **19**(2), 223–237 (2003)