Self-Confidence in Autonomous Systems
Papers from the AAAI 2015 Fall Symposium

Toward Estimating Task Execution Confidence for Robotic Bin-Picking Applications

Krishnanand N. Kaipa, Akshaya S. Kankanhalli-Nagendra, and **Satyandra K. Gupta**
Maryland Robotics Center, University of Maryland, College Park, MD.

Abstract

We present an approach geared toward estimating task execution confidence for robotic bin-picking applications. This requires estimating execution confidence for all constituent subtasks including part recognition and pose estimation, singulation, transport, and fine positioning. This paper is focussed on computing associated confidence parameters for the part recognition and pose estimation subtask. In particular, our approach allows a robot to evaluate how good the part recognition and pose estimation is, based on a confidence-measure, and thereby determine whether to proceed with the task execution (part singulation) or to request help from a human in order to resolve the associated failure. The value of a mean-square distance metric at a local minimum where the part matching solution is found is used as a surrogate for the confidence parameter. Experiments with a Baxter robot are used illustrate our approach.

Introduction

Currently deploying robots in industrial applications requires the reliability of robotic task execution to be high. This is accomplished by designing specialized hardware and software. Extensive system testing is needed to ensure all failure modes are well understood and contingency plans are developed to handle them. Task execution failures typically require the line to be shut down and human intervention to clear the fault and restart the line. This type of intervention is expensive and hence robots are not used on a task until high level reliability can be achieved. Customized hardware and software costs can only be justified if the production volume is sufficiently high (e.g., automotive assembly lines).

Currently robots have no way of assessing their own capability to complete a task. Consider the following case. A robot is capable of picking a part if it is presented to it at a certain location. However, if the part has shifted from its nominal location, the robot might not be able to pick it. The robot does not simply know where the transition boundary between task execution success and failure lies. As it attempts to pick the part, it might bump into it and push it further and jam the material handling system. This can in turn trigger a system fault and shut down the system.

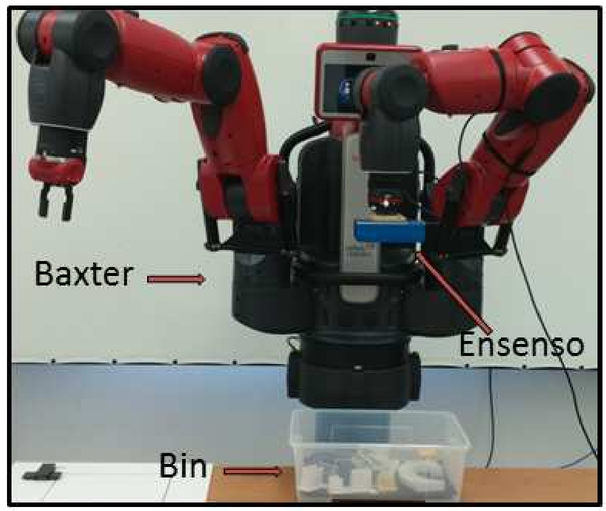

Figure 1: Experimental setup used for the bin-picking task

In order to use robots in small production batch operations, we will need robots that are able to estimate the probability of task completion before beginning the task. This will enable robots to assess their confidence in doing a task. If the robot does not have a high confidence in completing a task, then it can call for help. This will enable human operators to provide the robot the needed assistance (e.g., better part pose estimation, invoke a different grasping strategy) and prevent major system faults that result from task execution failure. Providing task assistance help to robots is cheaper than recovering from a system shutdown.

We will illustrate these concepts using robotic bin picking example in this paper. Robotic bin-picking is an important operation in many manufacturing and warehousing applications. Many research groups have addressed the problem of enabling robots, guided by machine-vision and other sensor modalities, to carry out bin-picking tasks (Kaipa et al. 2015b; Buchholz, Winkelbach, and Wahl 2010; Balakirsky et al. 2012; Schyja, Hypki, and Kuhlenkotter 2012). We are mainly interested in a class of bin-picking problems that manifest in the form of a part-order specifying multiple quantities of different parts to be singulated from a bin of

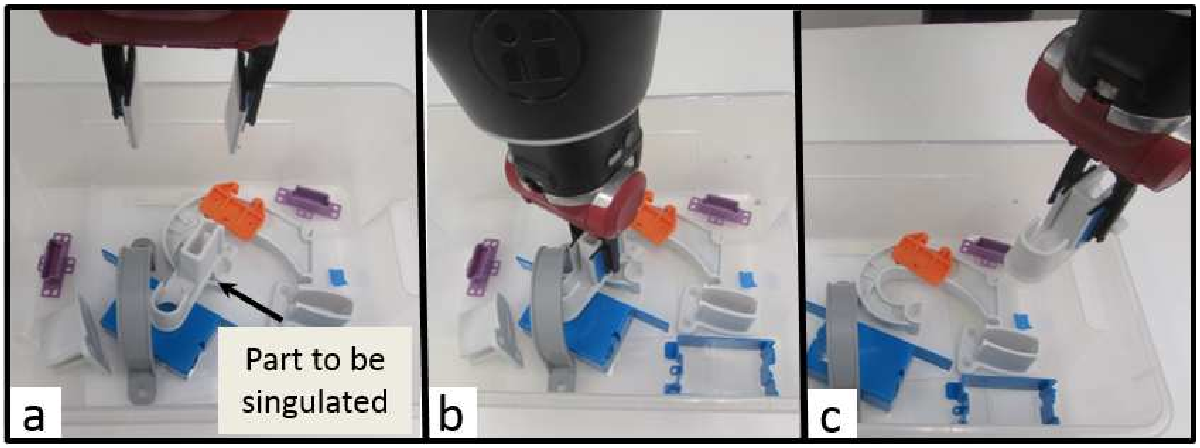

Figure 2: Illustration of the singulation task: (a) Robot gripper in the initial approach posture. (b) Part grasped. (c) Part successfully singulated.

randomly scattered pile of parts and transported to a destination location as rapidly as possible. Achieving this overall goal entails overcoming important challenges at various stages of task execution including part recognition and pose estimation, singulation, transport, and fine positioning. Accordingly it is important to enable the robot to estimate a level of confidence with which it can execute the overall task before actually proceeding to perform it. The singulation task, which involves picking only one part at a time is illustrated in Fig. 7.

This paper is focussed on part recognition and pose estimation. This problem is challenging and still not fully solved due to severe conditions commonly found in factory environments (Liu et al. 2012; Marvel et al. 2012). In particular, unstructured bins present diverse scenarios affording varying degrees of part recognition accuracies: 1) Parts may assume widely different postures, 2) parts may overlap with other parts, and 3) parts may be either partially or completely occluded. The problem is compounded due to factors like sensor noise, background clutter, shadows, complex reflectance properties of parts made of various materials, and poorly lit conditions. All these factors result in part recognition and pose estimation uncertainties.

In this paper, we present an approach that allows a robot to evaluate how good the part recognition and pose estimation is, based on a computed confidence-measure, and thereby determine whether to proceed with the task execution (part singulation) or to request help from a human in order to resolve the associated failure. For this purpose, we have developed a part matching algorithm that performs recognition and 6 DOF pose estimation of a part in a cluttered scene given a 3D point cloud of the scene and a CAD model of the part to be identified. A possible definition of confidence is a measure of how good a guess/decision is based on an estimate computed from the test data and the statistical mean of the training data (Pronobis and Caputo 2007). In this paper, the value of a mean-square distance metric (described later) at a local minimum where the part matching solution is found is used as a surrogate for the confidence parameter.

In another work (Kaipa et al. 2015b), we have developed a preliminary remote user interface that allows effective information exchange between the human and the robot that is geared toward solutions that minimize human operator time in resolving the detected part recognition failures. In this paper, we focus only on the automated part matching system and how to assess its pose estimation results.

Approach to Estimate Confidence During Part Recognition and Pose Estimation

Iterative closest point (ICP) (Besl and McKay 1992) is a well-established method used to match a pair of point clouds. ICP finds a match by performing transformations on one of the point clouds until an appropriately chosen mean-square distance metric is minimized. In our case, we deal with bins that contain multiple parts, that contribute to noise in the point cloud around the part we truly want to detect. In such a scenario, ICP algorithm could get stuck in a local minimum, thus returning a false match.

We address these issues by extracting features (e.g., edges) available in the sensed data and exploiting these features in a manner that collapses the problem from a search in 6D space to a 1D search along a finite number of lines oriented in a 3D space. Feature extraction (Weber, Hahmann, and Hagen 2010; Demarsin et al. 2007; Gumhold, Wang, and Macleod 2001) is one of the preprocessing procedures used in many scene reconstruction tasks. We use it for our problem of part recognition and pose estimation in cluttered scenes. In particular, our approach consists of first performing a neighborhood based normal estimation of each point in the point cloud. The normals are then binned to recognize planes in the point cloud. Iterating over these planes, we find intersection of planes to pick edges in the point cloud. The CAD model is rotated based on the orientation of the edge found and also the normals of the planes forming the edge. The CAD model is then filtered to contain only the points perceivable from the camera for that orientation of the CAD model. This filtered CAD model is iteratively moved over every edge found in the point cloud and the translated version that minimizes the mean square distance metric is declared as the desired part. Let $\mathcal{P} = \{p_i : p_i \in \mathbb{R}^3, i = 1, 2, \ldots, N\}$ be the point cloud of the bin of parts captured from 3D sensor. Let $\mathcal{Q} = \{q_i : q_i \in \mathbb{R}^3, i = 1, 2, \ldots, M\}$ be the point cloud obtained by uniform surface sampling of the CAD model of the part to be indentified. We first filter the $\mathcal{P}$ using z-thresholding to get rid of points that belong to the base bin. Next, we find 50-nearest neighbors of every point in $\mathcal{P}$ using a kd-Tree implementation, fit a plane to these points, and estimate the normal to the plane as the normal at that point.

The planes in the scene are estimated using the normals estimated in the previous step. For this purpose, the normals at all points are projected onto a unit circle and collected into a set of bins $\mathcal{B} = \{b_i\}$ on a 2D mesh with a bin resolution of (0.02, 0.02). From this set, we select a subset $\mathcal{B}'$ of bins with a value greater than a threshold τ_b and identify the normals in these bins as potential plane normals in the point cloud (τ_b was set to 17 in order to capture the smallest surface features in the bin scenarios considered in this paper).

$$\mathcal{B}' = \{b_i \in \mathcal{B} : |b_i| > \tau_b\} \tag{1}$$

Now, for each $b_i \in \mathcal{B}'$, we consider an infinite line L_i along the normal corresponding to the bin-center and project all the points in the point cloud $\mathcal{P}$ onto L_i. We bin these projections and select all bins that have a cardinality of 90 % of that of the bin with maximum cardinality and group them into $\mathcal{B}_p$. This second binning procedure bifurcates all the parallel planes in the point cloud. Each $b_p \in \mathcal{B}_p$ represents a set of projections whose corresponding points in the original point cloud belong to a plane. Therefore, we iterate over each point in $\mathcal{P}$ and compute the distance from the point to the plane fit for each corresponding $b_p \in \mathcal{B}_p$. If the distance is below a threshold value of 3mm, the point is considered to belong to that plane.

Having all the planes in the previous step, we iterate over all the planes and find planes that intersect at angles greater than some threshold τ_θ. The line of intersection of these planes is identified as an edge. As the plane estimation method is bound to have multiple planes representing the same plane, varying by small angles with respect to each other, the intersection method might have edges that are redundant. Therefore, averaging is carried out to obtain an edge estimate. Having found the edges in the point cloud, we first rotate the point cloud of the CAD model $\mathcal{Q}$ to orient it along each edge. We then use normal information of one of the planes forming the edge to get the $\mathcal{Q}$ to orient exactly to the object in the point cloud. We then filter the CAD model in its current orientation to remove all the points that are not visible from the point of view of the camera. This is done in two stages. First, we use the concept of back-face culling where we remove all the points whose normals make a negative or zero dot product with the camera to point vector. Second, we project a ray from every point in $\mathcal{Q}$ toward the camera. As we have information of faces while reading the CAD model, we check if the ray hits any of these faces. If it does hit a face, then it is an occluded point and we remove it from $\mathcal{Q}$.

At this point we have a filtered version of the point cloud of the CAD model $\mathcal{Q}_f$ oriented to the object in the point cloud and docked on the edge that was detected. We now move $\mathcal{Q}_f$ along the edge it is docked on as a function of a translation parameter s, and find the edge along which we get the minimum point-to-point mean distance ρ from the filtered CAD model to the point cloud from the sensor.

$$\rho = \min_s \frac{1}{|\mathcal{Q}_f|} \sqrt{\sum_{i=1}^{|\mathcal{Q}_f|} d(q_i, \mathcal{P})^2} \tag{2}$$

where, $d(q_i, \mathcal{P}) = \min_j ||q_i - p_j||, q_i \in \mathcal{Q}_f, p_j \in \mathcal{P}$

The parameter ρ acts as a surrogate measure for estimating confidence during the part recognition and pose estimation subtask. A low value of ρ implies high confidence and increasing values of ρ indicate a decline in confidence.

Experiments

The robotic set up used in the experiments is shown in Fig. 1. First, we consider a simple bin scenario as shown in Fig. 4(a). Figure 4(b) shows the corresponding point cloud obtained from an Ensenso 3D camera. Figures 4(c) - 4(h) illustrate the working of the part matching algorithm. The confidence level in the pose estimation of the target part is very high as indicated by a very low value of ρ (3 mm). Therefore, the robot decides to go ahead and execute the part singulation as shown in Fig. 7.

For calibration purpose, we considered a cuboid as shown in Fig. 5(a). We placed the cuboid in such a way that three planes of it were exposed to the 3D camera. We then registered a filtered CAD model of the cuboid with the acquired point cloud. The corresponding match is shown in Fig. 5(b). The ρ value for this match was found to be 1.66 mm. We computed the angle between planes that we had captured in the point cloud and found offsets from the ideal orthogonal value as all the planes are perpendicular to each other in the cuboid. The offsets were 0.62^o (between P_1 and P_2), 0.086 ^{o} (between P_2 and P_3), and 0.65^o (between P_1 and P_3), respectively. These offsets imply that if we were to match one plane from the CAD to its corresponding plane in the point cloud, we end up having the other plane off by some angle. Thereby these offsets contribute to a small non-zero point-to-point distance in the match. Also, by matching a single plane we found ρ to be 0.5 mm and with the multiple plane case as we expect the ρ for each plane to accumulate, leading to a larger value of ρ (1.66 mm). Owing to all these possible noise in a match, we kept a threshold of 3 mm in our experiments.

Figure 6 shows the matching results by running the algorithm on some representative bin scenarios. In particular, this experiment reveals how the matching performance (ρ value) changes as a function of bin complexity−parts of same type not touching with each other (Fig. 6(a, b)), parts of same type overlapping with each other (Fig. 6(c, d)), and parts of

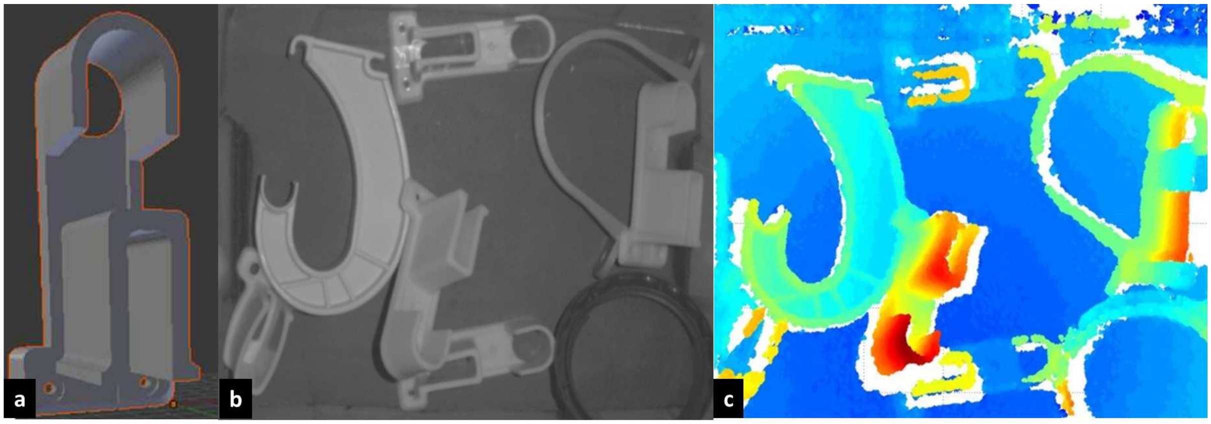

Figure 3: Inputs to the pose estimation algorithm: (a) CAD model of the part to be singulated. (b, c) Raw image and the corresponding 3D point cloud obtained from the Ensenso 3D camera.

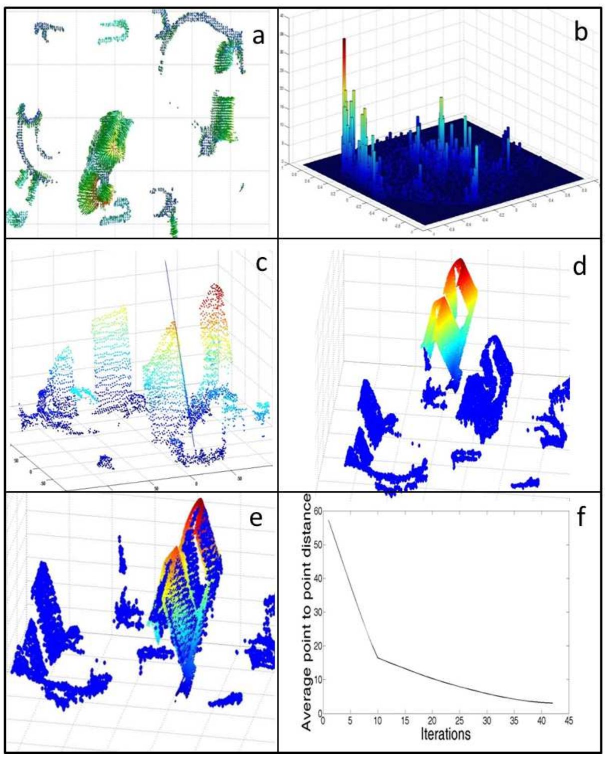

Figure 4: Example illustrating the various stages of the automated perception algorithm: (a) 3D plot showing estimated normals of a down-sampled point cloud. (b) Gauss map used to detect planes. (c) Edge extracted from intersection of two planes. (d) Initial docking of the CAD model along an oriented edge. (e) Final match obtained by translation of the CAD model along the oriented edge. (f) Plot of mean square distance as a function of number of iterations.

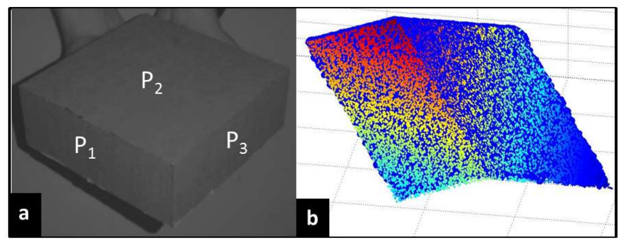

Figure 5: (a) Object used for calibration. (b) Match obtained between the point cloud of the scene and the filtered CAD model

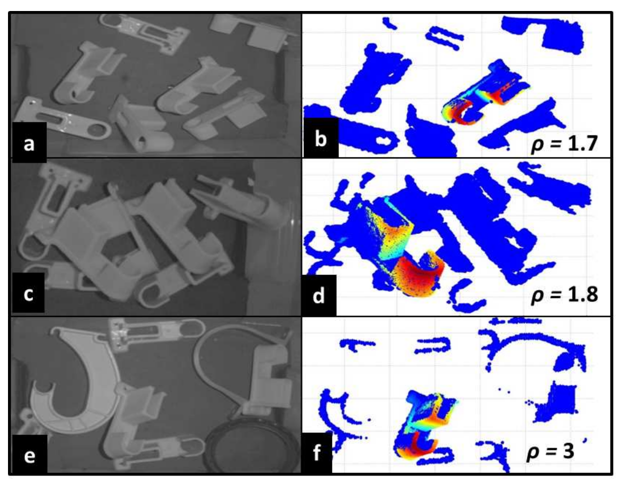

Figure 6: Representative bin scenarios and corresponding matches: (a, b) Multiple parts of same type not touching with each other. (c, d) Multiple parts of same type overlapping with each other. (e, f) Multiple parts of different type overlapping with each other.

Figure 7: Robot using the pose estimated by the system to proceed with the part singulation task.

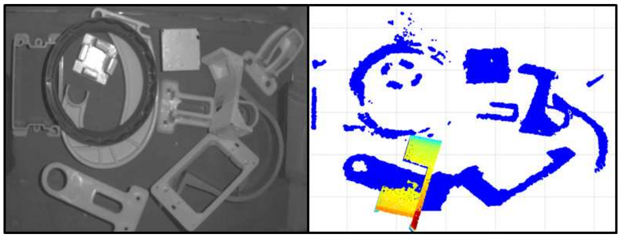

Figure 8: Bin scenario that results in a part matching failure. The high value of point-to-point distance (12 mm) captures this information and enables the robot to halt and request for human's help.

different type overlapping with each other (Fig. 6(e, f)). Figure 8 illustrates a bin scenario that results in a part matching failure. The high value of ρ (12 mm) captures this information and enables the robot to halt and request for human's help. Figure 9(a) shows an example where the part matching fails resulting in the robot invoking a Skype call to the human Fig. 9(b) requesting help. Figure 9(c) shows a snapshot of the user interface used by a remote human to resolve the part recognition and pose estimation failure. Figure 10 shows the robot performing part singulation using the pose estimation help provided by the human.

Conclusions

Object picking is a crucial capability to have for robots whether they are used for bin-picking tasks in a factory setting, packaging tasks in a warehouse, or handling utensils in a kitchen. Assessing their own capability of achieving this task enables them to either reliably perform the task or request a human for assistance rather than doing so after the damage is made. This requires estimating execution confidence for all constituent subtasks including part recognition and pose estimation, singulation, transport, and fine positioning. This paper presented a method geared toward estimating associated confidence parameters for the part recognition and pose estimation subtask. Integration between the automated perception system presented here, a singulation method (Kaipa et al. 2015a), a human-aided perception system (Kaipa et al. 2015b), and a fine-positioning method (Kaipa, Kumbla, and Gupta 2015) is currently under progress. In our previous work, we have developed other modules including ontology for task partitioning in human-robot collaboration for kitting operations (Banerjee et al. 2015), sequence planning for complex assemblies (Morato, Kaipa, and Gupta 2013), instruction generation for human operations (Kaipa et al. 2012), ensuring human safety (Morato et al. 2014b), and a framework for replanning to recover from errors (Morato et al. 2014a). Future work consists of investigating how to integrate them in order to realize hybrid work cells where humans and robots collaborate to carry out industrial tasks.

Acknowledgements

This work is supported in part by Office of Naval Research award N000141010886 and the Maryland Robotics Center.

References

Balakirsky, S.; Kootbally, Z.; Schlenoff, C.; Kramer, T.; and Gupta, S. K. 2012. An industrial robotic knowledge representation for kit building applications. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1365–1370.

Banerjee, A. G.; Barnes, A.; Kaipa, K. N.; Liu, J.; Shriyam, S.; Shah, N.; and Gupta, S. K. 2015. An ontology to enable optimized task partitioning in human-robot collaboration for warehouse kitting operations. In *Proc. SPIE, Next-Generation Robotics II; and Machine Intelligence and Bio-inspired Computation: Theory and Applications IX, 94940H*.

Besl, P. J., and McKay, N. D. 1992. Method for registration of 3-d shapes.

Buchholz, D.; Winkelbach, S.; and Wahl, F. M. 2010. Ransam for industrial bin-picking. In *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*, 1–6.

Demarsin, K.; Vanderstraeten, D.; Volodine, T.; and Roose, D. 2007. Detection of closed sharp edges in point clouds using normal estimation and graph theory. *Computer-Aided Design* 39(4):276 – 283.

Gumhold, S.; Wang, X.; and Macleod, R. 2001. Feature extraction from point clouds. In *In Proceedings of the 10 th International Meshing Roundtable*, 293–305.

Kaipa, K. N.; Morato, C.; Zhao, B.; and Gupta, S. K. 2012. Instruction generation for assembly operations performed by humans. In *ASME 2012 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 1121–1130. American Society of Mechanical Engineers.

Kaipa, K. N.; Shriyam, S.; Kumbla, N. B.; and Gupta, S. K. 2015a. Automated plan generation for robotic singulation from mixed bins. In *IROS Workshop on Task Planning for Intelligent Robots in Service and Manufacturing*.

Kaipa, K. N.; Thevendria-Karthic, S. S.; Shriyam, S.; Kabir, A. M.; Langsfeld, J. D.; and Gupta, S. K. 2015b. Resolving automated perception system failures in bin-picking tasks using assistance from remote human operators. In *Proc. of IEEE International Conference on Automation Science and Engineering*.

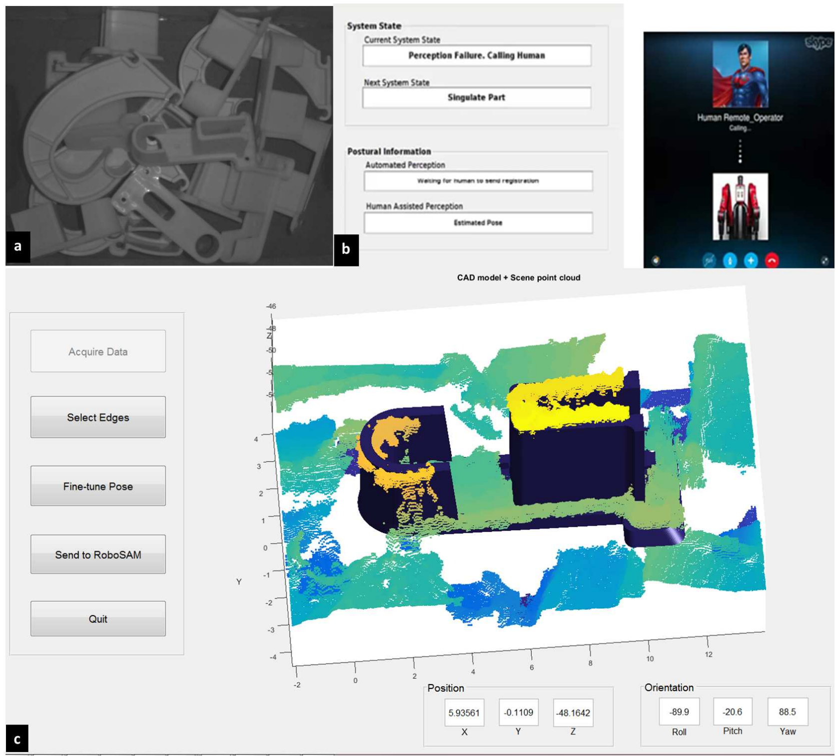

Figure 9: (a) Bin scenario that results in a part matching failure. (b) Robot sending Skype call to a remotely operating human requesting for help. (c) User interface used by a remote human to resolve part recognition and pose estimation failures

Figure 10: Robot using the pose estimated by the human to proceed with the part singulation task.

Kaipa, K. N.; Kumbla, N. B.; and Gupta, S. K. 2015. Characterizing performance of sensorless fine positioning moves in the presence of grasping position uncertainty. In *IROS Workshop on Task Planning for Intelligent Robots in Service and Manufacturing*.

Liu, M.-Y.; Tuzel, O.; Veeraraghavan, A.; Taguchi, Y.; Marks, T. K.; and Chellappa, R. 2012. Fast object localization and pose estimation in heavy clutter for robotic bin picking. *The International Journal of Robotics Research* 31(8):951–973.

Marvel, J. A.; Saidi, K.; Eastman, R.; Hong, T.; Cheok, G.; and Messina, E. 2012. Technology readiness levels for randomized bin picking. In *Proceedings of the Workshop on Performance Metrics for Intelligent Systems*, PerMIS '12, 109–113. New York, NY, USA: ACM.

Morato, C.; Kaipa, K. N.; Liu, J.; and Gupta, S. K. 2014a. A framework for hybrid cells that support safe and efficient human-robot collaboration in assembly operations. In *ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, V01AT02A078–V01AT02A078. American Society of Mechanical Engineers.

Morato, C.; Kaipa, K. N.; Zhao, B.; and Gupta, S. K. 2014b. Toward safe human robot collaboration by using multiple kinects based real-time human tracking. *Journal of Computing and Information Science in Engineering* 14(1):011006.

Morato, C.; Kaipa, K. N.; and Gupta, S. K. 2013. Improving assembly precedence constraint generation by utilizing motion planning and part interaction clusters. *Computer-Aided Design* 45(11):1349 – 1364.

Pronobis, A., and Caputo, B. 2007. Confidence-based cue integration for visual place recognition. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, 2394–2401.

Schyja, A.; Hypki, A.; and Kuhlenkotter, B. 2012. A modular and extensible framework for real and virtual bin-picking environments. In *Proc. of IEEE International Conference on Robotics and Automation*, 5246–5251.

Weber, C.; Hahmann, S.; and Hagen, H. 2010. Sharp feature detection in point clouds. In *Shape Modeling International Conference (SMI), 2010*, 175–186.