*Proceedings of the 35th European Safety and Reliability & the 33rd Society for Risk Analysis Europe Conference*
*Edited by* Eirik Bjorheim Abrahamsen, Terje Aven, Frederic Bouder, Roger Flage, Marja Ylönen

doi: 10.3850/978-981-94-3281-3_ESREL-SRA-E2025-P9797-cd

# Towards a framework for evaluation of spatial uncertainty for risk-based robotic decision-making

Klaus Ening[1], Ahmed Mohammed[2], Ingrid Bouwer Utne[1]

[1] *Department of Marine Technology, Norwegian University of Science and Technology, Trondheim, Norway*
*E-mail: { klaus.ening | ingrid.b.utne}@ntnu.no*
[2] *Smart Sensors and Microsystems, SINTEF Digital, Oslo, Norway E-mail: ahmed.mohammed@sintef.no*

Subsea Inspection, Maintenance and Repair (IMR) interactions on underwater Oil & Gas infrastructure can have severe consequences in case of failure. Currently, these interactions are mainly carried out using Remotely Operated Vehicles (ROVs) with attached robotic arms where operators assess the situation and make decisions. To allow for increased autonomy in operations on routine objects (valves, wires, hoses, tools), the ROV has to detect the objects and their pose before manipulation tasks can be performed. These tasks typically involve risks, and therefore it is desirable to estimate the probability of operation failure in order to provide decision-support to human operators during the mission. In this paper, we propose a framework using machine learning with a Gaussian Naive Bayes Classifier to estimate the failure-probability of robotic tasks based on the objects' spatial uncertainties. As the uncertainty input-feature we use the 6 DOF standard deviation of the object's pose-estimate. We show how prediction accuracy improves over time and how well the predictions match actual failure rates. We run 1000 simulated pick-and-place operations with different uncertainties and discuss how our method can improve decision-support during operation. We also include a small dataset collected by a 3D camera from real-world objects, test the transferability of simulation results to these data and a pose-estimation algorithm, and examine the impact of data quality.

*Keywords*: failure prediction, risk-based decision-making, uncertainty, robotic interaction

## 1. Introduction

### 1.1. *Background*

Robotic object-interaction is a common task in many industries. From very standardized assembly lines in manufacturing over logistics handling to remote operations for servicing and Inspection, Maintenance and Repair (IMR) tasks in challenging environments. Important factors for the use of robots, besides economy and efficiency, are reachability and risk reduction. This is especially the case in space and subsea operations and is a reason why divers have been mostly replaced by Remotely Operated Vehicles (ROVs) in the Oil & Gas industry. In addition to the environment being potentially harmful for humans, the tasks themselves can impose a risk. If a subsea maintenance job fails, it can lead to delays in production, creating direct loss of income, or worse, causing environmental damage through leakage of harmful substances.

IMR tasks require frequent interactions with known objects such as valves, wires, hoses, and other equipment on the submerged infrastructure. In order to automate these tasks, or assist human operators in their execution, the operational ”scene” has to be understood. The objects in question and especially their poses (location and orientation) have to be detected, and the desired manipulation actions have to be defined. As these interactions typically involve risks, it is additionally desirable to base decisions on both the potential (negative) outcomes of a failed interaction, and its corresponding probability.

The project SAFESUB (Transeth et al. (2024)) aims to acquire and incorporate higher-quality spatial uncertainty information about subsea objects to provide better and safer IMR operations. A perception pipeline consisting of a new three-dimensional (3D) camera and Computer-Aided Design (CAD) based object-detection is developed to provide not only the objects' poses but especially for providing *accurate* estimates about the spatial uncertainty of these poses. Figure 1

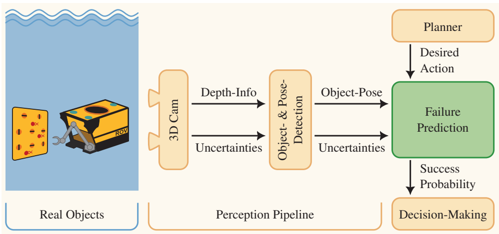


Fig. 1. Perception pipeline: A 3D camera attached to an ROV and object-detection algorithm provide the estimates for pose and uncertainty of real world objects as inputs to the failure prediction that is focused on in this work.

gives an overview over the elements in the perception pipeline.

### 1.2. *Related work*

**Uncertainty** Most research and applications dealing with uncertainties in object localization, especially in cybernetics, focus on the correct fusion and combination of different pieces of information coming from various sources or points in time. The main goal is to minimize the resulting uncertainty and typically involves the use of Kalman filters, factor graphs, and Simultaneous Localization and Mapping (SLAM) techniques.

Some publications, such as Risholm et al. (2021), mention the relevance of spatial uncertainties in object detection for the success of subsequent interaction tasks, such as robotic grasping. They also highlight the importance of adjusting acceptable uncertainty thresholds for high-risk underwater operations. Bari et al. (2023) and Brault et al. (2021) consider uncertainties during interaction- and path-planning to optimize the desired trajectories. However, quantified spatial uncertainty information is rarely used to predict end-to-end interaction outcomes.

**Interaction Success Prediction** Sogi et al. (2024) propose a long-time-horizon success prediction to optimize task planning. They employ a neural network to extract a low count of relevant features of an image series of the scene to be worked on and use these features as input for Success-or-Failure-Classification. Both neural networks, classifier and feature extraction, are trained and evaluated on generated and simulated robotic interaction tasks. They also mention the relevance of success-prediction for operator decision-support.

Pastor et al. (2011) apply iterative prediction of action-success based on all available sensor-data but focus mainly on the learning of complex motor skills. Rubert et al. (2019) focus on the specific task of grasping objects using robotic grippers and ways of predicting the success thereof. They compare predictions learned from different data- and feature-sets including simulated physics, human judgment, real robot executions and a set of quality metrics describing different object and robotic properties (shape, volume, posture, etc.) but not localization uncertainty.

Also focusing on grasping task success, Baek et al. (2022) employ uncertainties in object interaction to learn which feature-variations influence the success most and to select grasp locations on objects with high probability of success. While incorporating uncertainty in their decision-making, they do not predict the actual success of a task.

**Success Prediction using Uncertainties** Although the *minimization* of spatial uncertainty and the prediction of task-success are common in existing research, the combination of predicting the outcome of interactions with known objects based mainly on their localization uncertainty to support decision-making for human operators or autonomous systems is rare.

### 1.3. *Objective of paper*

The goal of the presented work is to make use of the knowledge about the spatial uncertainty in decision making and risk evaluation. In section 2 we propose a methodology to use the quantified spatial uncertainty elements of an interaction-object as indicator values to predict the failure-probability for a given, known robotic interaction task. By simulating an interaction scenario multiple times with various uncertainty values, we learn from the seen outcome and provide a failure-estimate for new encounters before the manipulation task itself starts. The implementation of a simple pick-and-place interaction is demonstrated in section 3, before presenting (section 4), and discussing (section 5) the results.

## 2. Methods

To assess the usability of spatial uncertainty information to predict the failure of an interaction, we propose to train a probabilistic binary classifier for each type of interaction using the elements of uncertainty as input features and the binary outcome of the interaction (success or failure) as the classes.

### 2.1. *Spatial information and uncertainty*

The pose of an object includes the location and orientation of its origin within a reference frame:

$$\boldsymbol{\xi} = [\boldsymbol{p}, \boldsymbol{\Theta}] = [x, y, z, \phi, \theta, \psi] \tag{1}$$

is the 6 Degrees of Freedom (DOF) pose and its decomposition in translational and rotational elements.

Its uncertainty can be expressed using the variance $\boldsymbol{\sigma}^2$ or standard deviation $\boldsymbol{\sigma}$ of these elements:

$$\begin{aligned} \boldsymbol{\sigma^2} &= [\sigma_x^2, \sigma_y^2, \sigma_z^2, \sigma_\phi^2, \sigma_\theta^2, \sigma_\psi^2] \\ \boldsymbol{\sigma} &= \sqrt{\boldsymbol{\sigma^2}} = [\sqrt{\sigma_x^2}, \dots, \sqrt{\sigma_\psi^2}] \end{aligned} \tag{2}$$

We use the standard deviation values in $\boldsymbol{\sigma}$ as the uncertainty features for classification.

### 2.2. *Classifier*

A Classifier is an algorithm that predicts which class or category an instance belongs to based on a set of features. A *probabilistic* classifier also estimates the likelihood of each category.

We use a planned, known interaction task on a detected object as the instance $i$, the object's pose uncertainty $\boldsymbol{\sigma}$ as the features, and the success ($S$) or failure ($F$) of the task as categories.

Using a probabilistic classifier we can then get an estimate of the likelihood of success and its complement (failure):

$$\begin{aligned} &P(\text{success}) \approx \\ &\hat{P}(i \in S|\boldsymbol{\sigma}) = \text{classifier.predict}(\boldsymbol{\sigma}) \\ &P(\text{failure}) = 1 - P(\text{success}) \end{aligned} \tag{3}$$

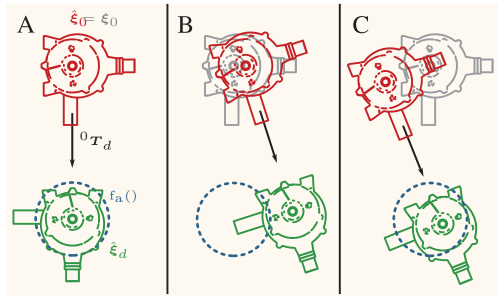


Fig. 2. Pick-and-place interaction and acceptance for a brass part: *A:* Desired interaction without uncertainty. *B:* The estimated start pose $\hat{\boldsymbol{\xi}}_0$ is different from the actual pose $\boldsymbol{\xi}_0$, resulting in the center of the estimated target $\hat{\boldsymbol{\xi}}_d$ and the potential end pose to be outside off the acceptance region $\mathrm{f_a}()$. *C:* The center of the end pose is inside the acceptance region, the large offset in the estimated start pose makes it unlikely though that the part would get picked up correctly.

The classifier is trained in a *supervised* way, meaning that the correct outcome is provided:

$$\text{classifier.learn}(\boldsymbol{\sigma}, \text{outcome} \in \{F, S\}) \tag{4}$$

### 2.3. *Interaction*

An interaction can be loosely defined by the object to be acted on, it's start- and it's goal-state. To be able to evaluate the success, acceptance criteria have to be defined.

Because many IMR operations on various valves and objects can essentially be broken down into combinations of pick-and-place or pick-and-turn tasks (grasping an object or tool, moving it and releasing it), we choose this as our basic interaction.

We use the poses $\boldsymbol{\xi}_0$ as the start state, $\boldsymbol{\xi}_d$ as the desired state, and $\boldsymbol{\xi}_{end}$ as the actual outcome state.

$$\mathrm{f_a}(\boldsymbol{\xi}_d, \boldsymbol{\xi}_{end}) \rightarrow \{F, S\} \tag{5}$$

is a function to check if the outcome is acceptable, and the task successful.

Figure 2 visualizes an interaction for a simple pick-and-place operation for a brass part (see subsection 3.1), as well as the target region defined by $\mathrm{f_a}$ to accept the task as successful. It also highlights a potential difference between the estimated start state $\hat{\boldsymbol{\xi}}_0$ and the real start state.

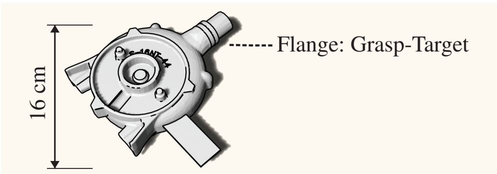


Fig. 3. Cast brass part of a water-pump used in perception and simulation to demonstrate the proposed method. Illustration based on CAD data.

## 3. Implementation

To assess the utility of the spatial uncertainty in itself while the real perception pipeline is still under development in the project, we chose to simulate the interaction with generated data as a first step.

### 3.1. *Datasets*

We use two datasets with generated uncertainty data for demonstration. *Dataset 1* is the main dataset and is sampled from given uncertainty distributions to train the classifier and test its feasibility. The uncertainty in *Dataset 2* is obtained using a trained 6D pose estimation algorithm applied to objects captured by a 3D camera, and demonstrates an early implementation of the perception pipeline. The object available for creating the second dataset is a component of a water-pump as illustrated in Figure 3. Therefore we use this object for the interaction simulation of both datasets.

### 3.2. *Interaction & simulation*

The goal of the pick-and-place interaction is to move the object relative to its start pose $\boldsymbol{\xi}_0$ by 20cm along it's own x-axis and turn it by 90° around its z-axis (${}^0\boldsymbol{T}_d$). The new center position should be within 5cm of the desired location, with rotation errors penalized by 10cm/radiant.

$$\begin{aligned} \boldsymbol{\xi}_0 &= [0.6, 0.05, 0.02, 0, -\pi, 0] \\ {}^0\boldsymbol{T}_d &= [0.2, 0, 0, -\pi, 0, 0] \end{aligned} \tag{6}$$

$$\mathrm{f_a}(\boldsymbol{\xi}_d, \boldsymbol{\xi}_{end}) : \begin{cases} S, & \text{if } \parallel \boldsymbol{\xi}_d - \boldsymbol{\xi}_{end} \parallel_2 \leq 0.05 \\ F, & \text{otherwise} \end{cases}$$

The object is placed within reach in front of a *Franka Emica*[a] *Panda* robot inside a simulated environment. The robot's gripper aims to grab the part by its extended flange, assumming the part is in the estimated pose $\hat{\boldsymbol{\xi}}_0$ that is provided by the perception pipeline.

The simulation is realized using *Gazebo Sim Harmonic*. Motion-planning, -control and -execution using *Robot Operating System (ROS) 2 Jazzy*, *MoveIt2* and `pymoveit2`[b].

### 3.3. *The perception pipeline*

The perception pipeline (cf. Figure 1) aims to provide the estimated pose $\hat{\boldsymbol{\xi}}_0$ of a detected object, along with *accurate* spatial uncertainty information $\boldsymbol{\sigma}$ based on the real object's pose $\boldsymbol{\xi}_0$.

$$[\hat{\boldsymbol{\xi}}_0, \boldsymbol{\sigma}] = \mathrm{perception.\,detect}(\boldsymbol{\xi}_0) \tag{7}$$

To test the proposed prediction method, we generate *Dataset 1* containing 1000 entries by sampling poses from a normal distribution. A small *Dataset 2* with six entries is generated by sampling poses from a trained object-pose estimation algorithm, as part of the intended perception pipeline.

**Dataset 1** To guarantee poses with correct associated uncertainties we pick and fix a standard deviation as the uncertainty for each of the 6 DOF and then sample a pose based on these. To allow the classification algorithm to learn, the uncertainty values have to vary between the different entries in the dataset during learning. This is accomplished by picking a new random standard deviation ($\boldsymbol{\sigma}$) before sampling each pose.

$$\boldsymbol{\sigma} \sim \mathcal{N}(\mathbf{0}, [.015, .015, .015, .04, .04, .04]^2) \tag{8}$$

Eq. (8) sets a new pose uncertainty $\boldsymbol{\sigma}$ by sampling from a normal distribution with zero mean and the given variances.

$${}^0\boldsymbol{T}_{\hat{0}} = \boldsymbol{\xi}_{sampled} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}^2) \tag{9}$$

Eq. (9) samples the pose $\boldsymbol{\xi}_{sampled}$ from a normal distribution with zero mean and variance $\boldsymbol{\sigma}^2$. This

[a] https://franka.de/
[b] https://github.com/AndrejOrsula/pymoveit2

pose is then used to offset the true object pose $\boldsymbol{\xi}_0$ to the estimated pose $\hat{\boldsymbol{\xi}}_0$.

$$\hat{\boldsymbol{\xi}}_0 = \boldsymbol{\xi}_0 {}^0\boldsymbol{T}_{\hat{0}} \tag{10}$$

The individual uncertainty $\boldsymbol{\sigma}$ and estimated pose $\hat{\boldsymbol{\xi}}_0$ are stored for each data-entry and later used for training and simulation respectively.

**Dataset 2** To address uncertainty in pose estimation for real-world objects in challenging subsea environments, our detection departs from traditional object pose regression approaches and instead estimates object poses by sampling candidates from a diffusion model (see Ikeda et al. (2024)). This approach enables simulation and analysis of pose variability within the perception pipeline for real-world object pose estimation. We use 3D data of cast manufactured parts (see subsection 3.1) acquired for the scan-to-cad dataset in Mohammed et al. (2023) for training a diffusion model. The data was captured by placing the parts on a turntable and scanning them with a Zivid 3D camera. Additionally the true pose $\boldsymbol{\xi}_G$ of the object is referenced as ground-truth. To simplify the training, the object-detection uses a continuous 6D rotation representation proposed in Zhou et al. (2020) that overcomes the discontinuities of typical rotation representations such as quaternions or Euler angles.

The object detection's output for each data-entry consists of 50 pose-candidates. From these pose candidates we need to define the estimated pose $\hat{\boldsymbol{\xi}}_0$ to be used for simulation and return the uncertainty $\boldsymbol{\sigma}$ in the same form as before. We opt to find the *mean* of candidates $\boldsymbol{\xi}_\mu$ as the base for the estimated pose $\hat{\boldsymbol{\xi}}_0$.

While easy for the position, the averaging of orientation representations is more complex and subject to current research. For simplicity we refer to Hartley et al. (2013) and use the L2-cordial mean with an implementation in `scipy`[c].

Replacing Eq. (9), the offset between mean pose $\boldsymbol{\xi}_\mu$ and ground-truth $\boldsymbol{\xi}_G$ is used as the estimation offset for the simulation of the dataset 2 entries.

$${}^0\boldsymbol{T}_{\hat{0}} = \boldsymbol{\xi}_G{}^\top \boldsymbol{\xi}_\mu \tag{11}$$

### 3.4. *Classification*

We choose *Gaussian Naive Bayes* with categories for success and failure as a simple classifier. It allows for iterative *online* supervised learning during operation and returns not only the most likely category but also the probabilities for a feature set belonging to each category. We use `river`'s[d] implementation.

## 4. Results

For *Dataset 1* we simulated 1000 pick-and-place interactions using the suggested methods and implementation. In 641 cases the interaction was judged a success. Figure 4 visualizes the outcome against the combined translational and rotational uncertainties.

### 4.1. *Binary prediction*

The binary prediction is the most likely category; either Success or Failure.

Starting with no prior knowledge the method learns from each outcome and predicts before each task. By the end, it shows an overall *accuracy* of 67%, meaning that in 670 of 999 cases (no

[c]https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.transform.Rotation.mean.html

[d]https://riverml.xyz/dev/api/naive-bayes/GaussianNB/

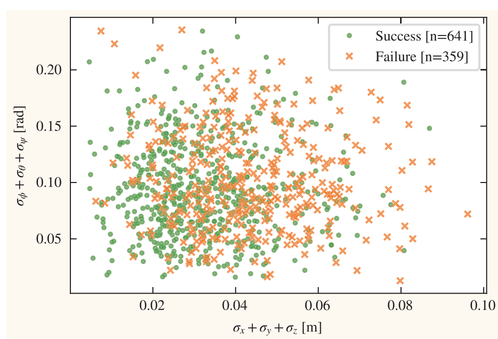


Fig. 4. Interaction outcomes: Successful● and failed× interactions plotted against the summed up uncertainties for location ($\boldsymbol{\sigma}_{x,y,z}$) and orientation ($\boldsymbol{\sigma}_{\phi,\theta,\psi}$).

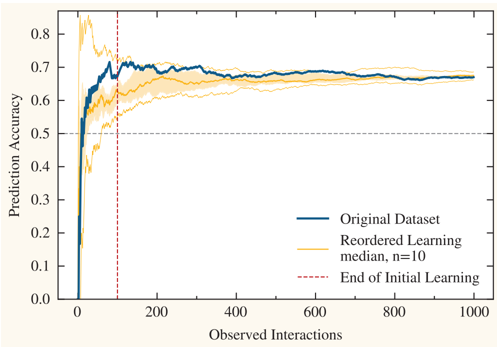


Fig. 5. Accuracy development: The ratio of correct predicted outcomes during 1000 interactions for the original dataset and for 10 learning phases in different randomized order in a boxplot-style: lines for min, max and median, area for interquartile range. Also showing where the initial learning ends.

prediction for the first case) the predicted outcome corresponds to the real outcome.Table 1 shows the complete confusion matrix.

Table 1. Confusion Matrix for *Dataset 1*

| | | Predicted Failure | Predicted Success |
|---|---|---|---|
| Actual | F | 127 | 231 |
| Actual | S | 98 | 543 |

Figure 5 shows the development of the accuracy during learning. Especially in the beginning, when not many outcomes have been seen, it fluctuates substantially. After around 100 interactions it converges towards the final value. To exclude a specific influence of the order of interactions, we repeat the learning with multiple randomized sequences and obtain similar results.

### 4.2. *Probability prediction*

Using a probabilistic classifier, we do not only get the most likely category, i.e., failure or success, but also an estimate of the probability of a feature-set belonging to each category (see Eq. (3)).

For evaluation, we discard the first 100 predictions from the initial learning phase, sort the remaining results by their predicted success probability, and divide them into ten equally sized bins. Each bin includes about 10% of the results. For each bin we determine the median predicted success probability ($\mathrm{M}(\hat{P})$) and compare it to the rate of success in simulation ($R$) within that bin. Results can be seen in Table 2 and Figure 6.

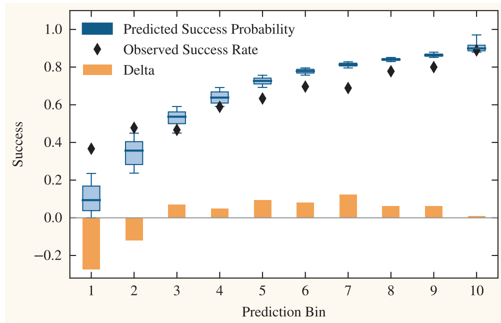


Fig. 6. Probability Prediction after the initial learning phase compared to actual outcome. Each bin includes about 90 simulated interactions.

$$R = \frac{s}{n} \tag{12}$$

with $n$ being the total and $s$ the successful number of observed interactions.

Table 2. Probability prediction after the initial learning phase compared to actual outcome:

| Bin | $\hat{P}(i \in S\|\boldsymbol{\sigma})$ | $n$ | $s$ | $\mathrm{M}(\hat{P})$ | $R$ | $\Delta$ |
|---|---|---|---|---|---|---|
| 1 | $[0.00, 0.23]$ | 90 | 33 | 0.09 | 0.37 | −0.27 |
| 2 | $(0.23, 0.45]$ | 90 | 43 | 0.36 | 0.48 | −0.12 |
| 3 | $(0.45, 0.59]$ | 90 | 42 | 0.54 | 0.47 | 0.07 |
| 4 | $(0.59, 0.69]$ | 90 | 53 | 0.64 | 0.59 | 0.05 |
| 5 | $(0.69, 0.75]$ | 90 | 57 | 0.73 | 0.63 | 0.09 |
| 6 | $(0.75, 0.79]$ | 89 | 62 | 0.78 | 0.70 | 0.08 |
| 7 | $(0.79, 0.82]$ | 90 | 62 | 0.81 | 0.69 | 0.12 |
| 8 | $(0.82, 0.85]$ | 90 | 70 | 0.84 | 0.78 | 0.06 |
| 9 | $(0.85, 0.87]$ | 90 | 72 | 0.86 | 0.80 | 0.06 |
| 10 | $(0.87, 0.97]$ | 90 | 80 | 0.90 | 0.89 | 0.01 |

### 4.3. *Dataset 2*

The second dataset only contains six entries at this point due to the effort needed to collect data, especially ground-truth related to real object's pose. Further, this is a demonstration of the proposed method and the pose estimation is still under development.

It is not possible to re-train the classifier compared to the 100 samples needed to reach a stable prediction accuracy (cf. Figure 5). Instead we use the already trained classifier from *Dataset 1* for outcome-prediction and simulate the same pick-and-place operation to compare the real outcome.

This can give us a first impression on how well the already trained classifier can be transferred to samples that are closer to the intended pipeline but also more costly to generate. Additionally it allows us to evaluate the data quality from the detection in its current state.

The *simulated* outcome for all six cases show failed interactions, although four of them were *predicted* to be successful.

## 5. Discussion

### 5.1. *Framework*

The presented framework, using few features and *Gaussian Naive Bayes* as classifier, is computationally very lightweight and can be implemented on limited hardware while at the same time learning incrementally from new observed outcomes during operation. A bigger challenge in practical implementation, especially with increasing autonomy, is to decide at what point an interaction should be considered as ended to invoke $f_a()$ for determining the true outcome.

### 5.2. *Binary prediction*

The results regarding binary prediction for *Dataset 1* (subsection 4.1) show that the spatial uncertainty alone has information about the successful or failed outcome of an interaction, and that this information *can* be used to predict the outcome. This information can be harnessed even with both a relatively simple representation of the uncertainty ($\sigma$) in combination with a simple classification method. While we, as a first step, work mainly with sampling, simulation and the abstraction of underwater robotic interaction to a generic pick-and-place operation, we argue the concept to hold in general for subsea interaction.

Using iterative learning, the prediction accuracy varies a lot in the beginning and is dependent on the order and outcomes. Though, as Figure 5 shows, it approaches 67% in the dataset shown, independent of order.

It is lower than the grasp-success prediction of 80% in Rubert et al. (2019) but on the other hand contains the prediction of end-to-end interaction (including grasping) and works on a more generalized and abstract feature set. Looking at the *precision* of the prediction the 70% presented here (543 out of 774 predicted successful outcomes where actual successes) are within the range of 62% to 92% reported by Sogi et al. (2024)'s method.

### 5.3. *Probability prediction*

The usage of a probabilistic classifier allows us to predict the likelihood of a successful outcome. The results shown in Table 2 and Figure 6 show a pessimistic prediction in the low end (with a higher success rate than predicted) and a slightly too optimistic prediction for the rest. Apart from the lowest prediction bin, the difference remains below 12%. If there is a set of alternative actions, the predicted success probability for all of them can be used to choose the most promising alternative for execution, as is done for grasping actions in Baek et al. (2022).

The success- and failure-probability prediction of an interaction should be especially relevant for risk-aware decision-making when combined with information about potentially negative consequences of a failed interaction. While a decision solely based on this approach might be too uncertain for high risk environments, the prediction provided with the proposed method can be *one* valuable information source.

### 5.4. *Sources of data*

The second dataset highlights that some work is needed to align the data-qualities used for training and prediction and that datasets from a different source need careful evaluation. The poses do not have a big variance, comparing their mean to the ground truth, however, shows an offset, revealing a high precision but too low accuracy of the current object-pose detection to be used with the pre-trained classifier. In general, each learned outcome is currently only valid for a particular combination of object, interaction and configuration (arm, gripper, trajectory planner, environment, etc.). While a transferability of the classifier is likely once the provided real data and the one trained on are qualitatively close enough to each other, different configurations most likely need retraining.

### 5.5. *Further work*

Although the proposed framework looks promising, the next step is to have a closer look into the data provided by the perception-pipeline and if the methods transfer well to learning on data from real objects. While costly to set up, it would also be interesting to determine the interaction outcome in real-world robotic experiments. Furthermore, the combination of different classifiers, potentially neural networks, and more generalized training, across different object-interaction-combinations might both harness the available information in a more efficient way, allow for more transferability between different scenarios, and open up future research topics.

## 6. Conclusion

This paper focuses on demonstrating a framework for estimating the failure probability of robotic object-interaction from spatial uncertainty data. The framework uses a relatively simple machine learning approach for the prediction, which shows a promising potential to be combined with information on unwanted consequences and used for online risk-aware decision-making.

**Acknowledgement**

This paper is funded by Norwegian Research Council (NRC)'s project #338647. Data for the brass parts manufactured by Mjøs Metallvarefabrikk has been captured in NRC's project #237900. The first author also holds a part-time position at SINTEF Digital's Department of Mathematics and Cybernetics, Norway.

## References

Baek, W.-J., C. Pohl, P. Pelcz, T. Kröger, and T. Asfour (2022, November). Improving Humanoid Grasp Success Rate based on Uncertainty-aware Metrics and Sensitivity Optimization. In *2022 IEEE-RAS 21st Int. Conf. on Humanoid Robots (Humanoids)*, Ginowan, Japan, pp. 786–793. IEEE.

Bari, S., V. Gabler, and D. Wollherr (2023, August). Probabilistic Inference-Based Robot Motion Planning via Gaussian Belief Propagation. *IEEE Robotics and Automation Letters 8*(8), 5156–5163.

Brault, P., Q. Delamare, and P. R. Giordano (2021, May). Robust Trajectory Planning with Parametric Uncertainties. In *2021 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pp. 11095–11101. ISSN: 2577-087X.

Hartley, R., J. Trumpf, Y. Dai, and H. Li (2013, July). Rotation Averaging. *Int. Journal of Computer Vision 103*(3), 267–305.

Ikeda, T., S. Zakharov, T. Ko, M. Z. Irshad, R. Lee, K. Liu, R. Ambrus, and K. Nishiwaki (2024, March). DiffusionNOCS: Managing Symmetry and Uncertainty in Sim2Real Multi-Modal Category-level Pose Estimation. arXiv:2402.12647 [cs].

Mohammed, A., J. Kvam, I. F. Onstein, M. Bakken, and H. Schulerud (2023, January). Automated 3D burr detection in cast manufacturing using sparse convolutional neural networks. *Journal of Intelligent Manufacturing 34*(1), 303–314.

Pastor, P., M. Kalakrishnan, S. Chitta, E. Theodorou, and S. Schaal (2011, May). Skill learning and task outcome prediction for manipulation. In *2011 IEEE Int. Conf. on Robotics and Automation*, pp. 3828–3834. ISSN: 1050-4729.

Risholm, P., P. Ørnulf Ivarsen, K. Henrik Haugholt, and A. Mohammed (2021, October). Underwater marker-based pose-estimation with associated uncertainty. In *2021 IEEE/CVF Int. Conf. on Computer Vision Workshops (ICCVW)*, pp. 3706–3714. ISSN: 2473-9944.

Rubert, C., D. Kappler, J. Bohg, and A. Morales (2019, November). Predicting grasp success in the real world - A study of quality metrics and human assessment. *Robotics and Autonomous Systems 121*, 103274.

Sogi, N., H. Oyama, T. Shibata, and M. Terao (2024, April). Future Predictive Success-or-Failure Classification for Long-Horizon Robotic Tasks. arXiv:2404.03415 [cs].

Transeth, A. A., J. Thorstensen, A. Mohammed, J. T. Thielemann, K. Ening, E. I. Grøtli, B. O. A. Haugaløkken, M. A. Brandt, M. T. Møller, R. P. Hovland, O. S. Erland, I. B. Utne, and I. Schjølberg (2024, September). SAFESUB: Safe and Autonomous Subsea Intervention. In *2024 20th IEEE/ASME Int. Conf. on Mechatronic and Embedded Systems and Applications (MESA)*, pp. 1–8. ISSN: 2835-902X.

Zhou, Y., C. Barnes, J. Lu, J. Yang, and H. Li (2020, June). On the Continuity of Rotation Representations in Neural Networks. arXiv:1812.07035 [cs].