Probabilistic Visual Verification for Robotic Assembly Manipulation

Changhyun Choi and Daniela Rus

***Abstract*— In this paper we present a visual verification approach for robotic assembly manipulation which enables robots to verify their assembly state. Given shape models of objects and their expected placement configurations, our approach estimates the probability of the success of the assembled state using a depth sensor. The proposed approach takes into account uncertainties in object pose. Probability distributions of depth and surface normal depending on the uncertainties are estimated to classify the assembly state in a Bayesian formulation. The effectiveness of our approach is validated in comparative experiments with other approaches.**

I. Introduction

Robotic manipulation is the process of using robots' hands to rearrange robots' environment [1]. For successful manipulation, robust perception of the state of the environment is very important [2]. While there have been active efforts in robotic perception, the state-of-the-art results are often restricted to single entity pose estimation/tracking approaches which are mainly suitable for simple pick-and-place tasks. To proceed toward more complex assembly manipulation tasks we have to address several challenges:

- **Self-occlusions**: When objects are assembled together, self-occlusions naturally occur. (e.g. a bolt is screwed into a part or a peg-in-hole task)
- **Sensor noise**: Measurements are subject to sensor noise. A proper sensor model is crucial for robust estimation.
- **Uncertainty in pose**: Due to sensor noise, estimated object pose is always uncertain. It is thus required to take into account the uncertainty in the pose estimate of an ongoing task.

Traditionally, factory assembly lines have relied on automated machine vision technology [3], [4], [5]. However, most of the systems require well structured settings such as controlled illuminations and carefully designed fixtures. The visual features for the visual inspection are often manually defined and depend on specific tasks.

The conventional assembly lines need to be more flexible [6], and hence assembly verification should be versatile. We envision a reconfigurable verification system in which a mobile manipulator augmented with a sensor inspects the assembly manipulation state. As the verification system works in untethered settings, it has to cope with various uncertainties, such as in sensor measurements and in pose estimates of assembly parts, as well as the self-occlusions.

Pioneering work in inspection includes [7], [8] where the 'verification vision' (VV) problem is introduced. The VV system inspects the location of an object via several visual operators. The main difference between this problem and general object pose recognition is that *a large amount of prior knowledge about object types and placements are available.*

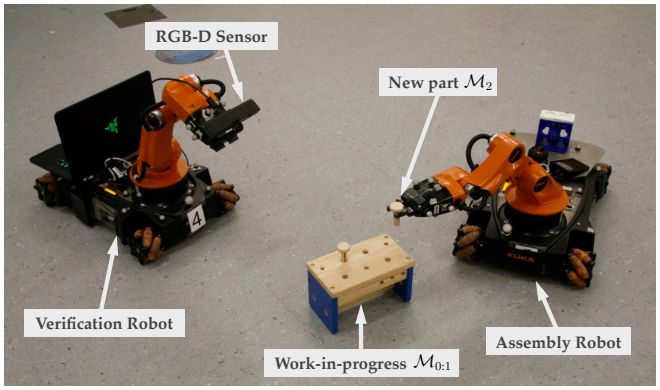


Fig. 1: **Visual verification system.** A heterogeneous robotic team is considered in this work where one mobile manipulator is designated as an assembly robot while the other robot visually verifies assembly operations using an RGB-D sensor attached on its end effector. The assembly robot is inserting the new peg $\mathcal{M}_2$ to the work-in-progress $\mathcal{M}_{0:1}$. Note that our visual verification approach is not restricted to this multi-robot system; it can be applied to any robot systems which have manipulator(s) and a depth sensor.

In this paper, we propose an approach for visual verification exploiting prior knowledge. We assume multi-step assembly tasks and wish to verify the correctness of every step, where a new part is assembled to the work-in-progress. Fig. 1 describes our visual verification system, which consists of one assembly mobile manipulator and a verification robot with a depth sensor. An RGB-D sensor is chosen as a depth sensor since the depth channel is suitable for textureless assembly parts. For optimal decisions on the assembly state, we formulate the verification problem as a Bayesian classification. The main contributions in this paper are as follows:

- A versatile visual verification approach which performs inspection tasks without significant setup costs and is applicable to semi-structured settings.
- A noise model for depth and surface normal measurements, which is denominated as Per-pixel Gaussian Noise Model (PGNM). Depth values are modeled as a 1D Gaussian along the axis of ray, while surface normal values are modeled as a 2D Gaussian distribution in the tangent plane on the unit sphere S^2.
- A generic visual verification framework which takes into account pose uncertainties as well as sensor noise. Naive Bayes classification formulation with the PGNM leads to robust classification of assembly state.
- Applications to sequential robotic assembly scenarios:

*This work was supported by the Boeing Company.

The authors are with Computer Science & Artificial Intelligence Lab, Massachusetts Institute of Technology, Cambridge, MA 02139, USA {cchoi,rus}@csail.mit.edu

block assembly and peg-in-hole.

This paper is organized as follows. We review existing work related to our problem in Section II. We begin with a toy example which is a simplified 1D case in Section III, and our probabilistic visual verification extending to 2.5D case is presented in Section IV. Section V and Section VI further describe PGNM and Bayesian classification, respectively. Finally, experimental evaluations and comparisons to a recent method are shown in Section VII.

II. Related Work

Industrial vision systems have been extensively studied and employed in factory assembly lines [3], [4], [5] for many applications ranging from assembly to inspection. A typical setup for industrial visual inspection is to point one or more sensors at objects commonly transported by a conveyor belt. The imaging devices as well as the transporting system are carefully placed in the factory environments. The location and orientation of objects are thus typically known, and the illumination of the environment is strictly controlled [3]. The most common sensor is a monocular camera, but range or depth sensors have become popular because of their insensitivity to ambient illumination, easier background subtraction, and direct relation to the surfaces of objects [9], [10]. All these approaches assume known pose of targets and require a carefully designed setup.

Object localization has been tackled by employing various features such as edges, lines [11], corners, or keypoint descriptors [12]. A classical detection approach is the template matching [13] in which a set of edge templates is matched to a test edge image in an exhaustive sliding window approach. Corners and keypoint descriptors were actively adopted for generic object recognition [12], [14], [15] and 6-DOF localization of highly textured objects [16]. It is worth noting that these work does not take advantage of the prior knowledge of object location; they rather search for the object in a brute-force manner. Since the prior knowledge of placement is not considered, self-occlusions is not properly handled. Uncertainties in sensor or pose are often ignored in many work.

One of the popular localization algorithms hinged on the pose prior is the Iterative Closest Point (ICP) algorithm [17]. ICP is an iterative algorithm which gradually minimizes the error of correspondences between two point clouds. Among various distance metrics, the effectiveness of the point-to-plane error metric has been outstanding [18]. While it converges to an optimal pose if the starting pose is within the basin of convergence, it could not be free from self-occlusions. Uncertainties in pose estimates also highly depend on the quality of data associations and geometric constraints of scenes.

Bayes classification has been well studied in machine learning literature and has been applied to various applications: spam filtering [19], binary skin color detection [20], and background subtraction [21]. For each class, a conditional probability and a prior distribution are generally learned from data. By Bayes' theorem, the conditional and prior distributions are fused to estimate a posterior distribution. The final decision is then determined by maximum *a posteriori* (MAP) [22].

Bolles's VV [7], [8] uses edge, corner, region operators and uses Bayes' theorem to estimate the posterior probability of correct and surprise given the value of operators. Our work is different from VV in three aspects. 1) While VV uses very limited number of visual features, our approach exploits full dense depth and surface normal data. Whereas the operator/feature pairs were manually defined by a human operator in VV, our approach does not require any manual setup. 2) Although VV employs a Bayesian formulation, it is restricted to selecting good operator features. We aggregate the confidence of each individual pixel resulting in a final decision along with its overall confidence. 3) As feature distributions in VV are learned from a set of fixed-view 2D images, it is constricted by the fixed viewpoint. Our approach, however, is viewpoint invariant as 3D shape models are employed along with 2.5 depth measurements, and this leads to a versatile visual verification.

The advent of dependable/affordable depth sensors enables robots to exploit reliable depth information for various robotics problems: bin-picking [23], [24], SLAM [25], [26], object tracking [27], and visual inspection [28]. The approach in [28] uses a ray-tracing method to count the number of different depth pixels and to decide if the object is well placed or not. This approach is, however, limited as it does not consider uncertainty at all nor takes into account the depth discrepancy information. Our approach is compared with this approach in Section VII.

III. 1D Visual Verification: A Toy Example

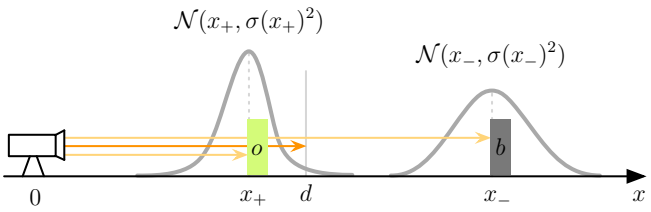


Fig. 2: **1D visual verification example.** A noisy range sensor is at 0 location, and a target object o and a background b are placed at x_+ and x_-, respectively. If the sensor reads d, is it from o or b, and what is the probability for each case? (Please read text for details.)

For simplicity, consider the 1D case in Fig. 2 which describes a noisy 1D depth sensor placed at coordinate 0. An object o (the right green box) is placed at x_+, and there is a background wall b at x_-. Both locations x_+ and x_- are known *a priori*. Suppose that the probability distribution of the sensor reading follows a Gaussian distribution, and the standard deviation $\sigma(x)$ is proportional to the squared distance from the sensor to the object as

$$\sigma(x) = \eta x^2$$

where η is a constant which represents the intrinsic characteristic of the sensor. When the sensor reads the depth d, the probability of d given x_+ is $p(d|x_+) = \mathcal{N}(d; x_+, \sigma(x_+)^2)$. If o is not properly placed or a different object is placed,

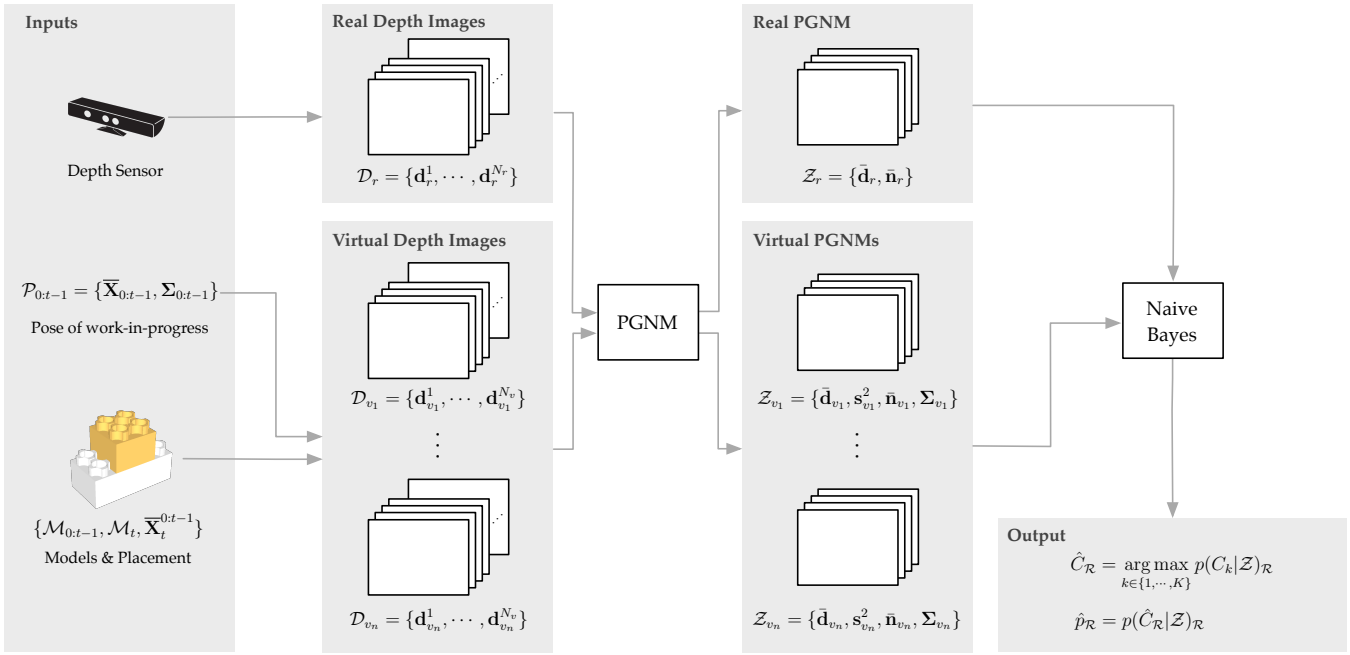


Fig. 3: **Overview.** Object shape prior $\mathcal{M}_{0:t-1}$, $\mathcal{M}_t$, and the their placement $\overline{\mathbf{X}}_t^{0:t-1}$ are given. The pose estimate of the work-in-progress $\mathcal{P}_{0:t-1}$ is available from the previous pose or ICP. Several sets of virtual depth images $\mathcal{D}_{v_1}, \cdots, \mathcal{D}_{v_n}$ are then rendered from the prior, and PGNMs $\mathcal{Z}_{v_1}, \cdots, \mathcal{Z}_{v_n}$ from the virtual depth images are estimated. A set of real depth images $\mathcal{D}_r$ is obtained from a depth sensor, and the mean images of depth and surface normal $\mathcal{Z}_r$ are calculated. Finally, the most likely class $\hat{C}_\mathcal{R}$ and its probability $\hat{p}_\mathcal{R}$ are decided via the naive Bayesian classification.

the sensor may return a depth value around the background x_-. Analogous to $p(d|x_+)$, it is similarly determined that $p(d|x_-) = \mathcal{N}(d; x_-, \sigma(x_-)^2)$. The main question is:

> *Given a sensor measurement d, what is the most likely location (among x_+ and x_-), and what is the probability of it.*

More formally, what are the posterior probabilities for both cases, $p(x_+|d)$ and $p(x_-|d)$, and which one is the most likely. Bayes' theorem gives a way to estimate the posterior by combining both conditional and prior probabilities as

$$p(x_+|d) = \frac{p(d|x_+)p(x_+)}{p(d)} = \frac{p(d|x_+)p(x_+)}{p(d|x_+)p(x_+) + p(d|x_-)p(x_-)}$$

where $p(x_+)$ and $p(x_-)$ are prior probability distributions which are often problem dependent. If we consider only two cases (x_+ and x_-), the other posterior probability is $p(x_-|d) = 1 - p(x_+|d)$. Once the posterior probabilities are calculated, the most probable class $\hat{c}$ and its probability $\hat{p}$ are determined as

$$\hat{c} = \underset{c\in\{+,-\}}{\arg\max}\, p(x_c|d), \quad \hat{p} = p(x_{\hat{c}}|d).$$

Note that $\{\hat{c}, \hat{p}\}$ is the function of (d, x_+, x_-). This is a simple Bayesian classification example in which only two cases were considered. In Section IV, we address more realistic and general scenarios.

IV. Probabilistic Visual Verification

In the previous section, we presented a toy example using a 1D sensor. Now we consider a 2.5D depth sensor

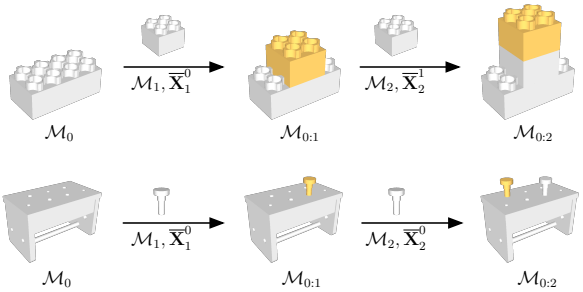


Fig. 4: **Assembly sequences.** Two sequential assembly scenarios are considered in this work: block assembly (upper) and peg-in-hole (lower).

case. The 2.5D sensor returns depth images in which each pixel value represents the 1D depth along the axis of ray of the sensor, and hence we can regard this sensor as a 2D grid of the 1D sensors. In this section we present our visual verification approach which is based on the Bayesian classification explained in Section III with an extension to surface normal distributions as well as depth distributions.

A. Assembly Scenarios

Assembly procedures often consist of sequences of atomic assembly operations. Fig. 4 describes two assembly scenarios considered in this paper: a block assembly and a peg-in-hole task. At time $t = 1$, a new part $\mathcal{M}_1$ and the initial part $\mathcal{M}_0$ are assembled according to the relative placement $\overline{\mathbf{X}}_1^0 \in SE(3)$. Similarly, at time $t = 2$, another new part $\mathcal{M}_2$ is added to the work-in-progress $\mathcal{M}_{0:1}$. Please note that the base model of the relative placement might be assembly dependent; the relative placement of the new part in the peg-in-hole is $\overline{\mathbf{X}}_2^0$, while it is $\overline{\mathbf{X}}_2^1$ in the block assembly. The goal of visual verification is to confirm if the new part (highlighted

as yellow) is properly placed or assembled at each assembly step.

B. Overview

The overall flow of our probabilistic visual verification is presented in Fig. 3. Object shape models of the work-in-progress $\mathcal{M}_{0:t-1}$ and the new object $\mathcal{M}_t$, and their relative placements $\overline{\mathbf{X}}_t^{0:t-1}$ are known *a priori*. The pose mean and uncertainty of the work-in-progress $\mathcal{P}_{0:t-1}$ can be given from the previous pose or an object localization module which will be explained in Section IV-C. From these information, virtual depth images are generated synthetically, and Per-pixel Gaussian Noise Models (PGNMs) (Section V) are estimated to serve as conditional probabilities for the final Bayesian classification (Section VI). Real depth images are gathered from a depth sensor, and mean of both depth and surface normal are estimated. Once PGNMs for virtual depth images and the mean estimates of both real depth and surface normal are estimated, the naive Bayesian classification decides the most likely class $\hat{C}_{\mathcal{R}}$ as well as its probability $\hat{p}_{\mathcal{R}}$.

C. Pose of Work-in-progress

Visual verification has significant prior knowledge. The pose for the work-in-progress within a multi-step assembly operation is not an exception. As considered assembly scenarios in this work are sequential assembly, the prior pose of the object is available with high fidelity. In other words, we do not need to re-localize the object from scratch at every step. However, the prior pose tends to be perturbed by the assembly manipulation at the previous step, and hence the prior pose estimate should be improved at each step. When the prior pose estimate is reasonably accurate, the Iterative Closest Point (ICP) algorithm [17] commonly converges to the optimal pose estimate.

The goal of ICP is to find the optimal motion estimate $\hat{\boldsymbol{\xi}}_i$ that minimizes the point-to-plane energy function $\mathcal{E}(\cdot)$ [18]:

$$\hat{\boldsymbol{\xi}}_i = \arg\min_{\boldsymbol{\xi}_i} \mathcal{E}(\boldsymbol{\xi}_i)$$

where $\boldsymbol{\xi}_i \in \mathbb{R}^6$ is the 6-DOF motion vector of i-th iteration. The optimal motion is generally estimated via solving the normal equation: $\hat{\boldsymbol{\xi}}_i = (\mathbf{H}^\intercal\mathbf{H})^{-1}\mathbf{H}^\intercal\mathbf{y}$ where $\mathbf{H} \in \mathbb{R}^{|\mathcal{A}|\times 6}$, $\mathbf{y} \in \mathbb{R}^{|\mathcal{A}|}$, and $|\mathcal{A}|$ is the number of point data associations (for details, please refer to [17], [18], [26]). The uncertainty covariance $\boldsymbol{\Sigma} \in \mathcal{S}_+^6$ [1] associated with the pose estimate can be determined by

$$\boldsymbol{\Sigma} := \sigma_\epsilon^2(\mathbf{H}^\intercal\mathbf{H})^{-1} \tag{1}$$

where σ_ϵ denotes error between each registered point correspondence [29]. An unbiased estimate of σ_ϵ^2 is calculated as $s_\epsilon^2 = \mathcal{E}(\hat{\boldsymbol{\xi}}_i)(|\mathcal{A}|-k)^{-1}$ where k is the number of parameters to be estimated[2]. Please note that the uncertainty covariance $\boldsymbol{\Sigma}$ is proportional to the energy $\mathcal{E}(\hat{\boldsymbol{\xi}}_i)$.

[1] $\mathcal{S}_+^n$ is the set of positive-semidefinite matrices $\mathcal{S}_+^n := \{\mathbf{M} \in \mathbb{R}^{n\times n} \mid \mathbf{x}^\intercal\mathbf{M}\mathbf{x} \geq 0, \forall \mathbf{x} \in \mathbb{R}^n\}$.

[2] In our case, $k = 6$ since we are estimating 6-DOF pose variable.

V. Per-pixel Gaussian Noise Model (PGNM)

In this section we introduce a Gaussian noise model for both depth and surface normal. We name it as the Per-pixel Gaussian Noise Model (PGNM).

A. Rendering with Pose Uncertainty

The pose of the work-in-progress $\mathcal{P}_{0:t-1}$ includes the mean $\overline{\mathbf{X}}_{0:t-1}$ and the covariance $\boldsymbol{\Sigma}_{0:t-1} \in \mathcal{S}_+^6$. As the covariance represents the uncertainty of the pose, it is necessary to take into account the uncertainty when we render the virtual depth images $\mathcal{D}_v$. It is, however, not straightforward to model the distribution analytically, since the depth distributions are function of multiple variables; note that depth variations in each pixel of $\mathcal{D}_v$ are not only subject to the degree of uncertainty $\boldsymbol{\Sigma}$, but also depend on the mean pose $\overline{\mathbf{X}}$ and the geometric shape of the object $\mathcal{M}_{0:t-1}$.

A tractable solution is to sample a set of poses from the probability density function $\{\overline{\mathbf{X}}_{0:t-1}, \boldsymbol{\Sigma}_{0:t-1}\}$ and to generate a set of virtual depth images $\mathcal{D}_v$ to model uncertainty for each pixel. To sample from the pose, the covariance $\boldsymbol{\Sigma}_{0:t-1}$ is decomposed by the Cholesky factorization as $\boldsymbol{\Sigma}_{0:t-1} = \mathbf{L}\mathbf{L}^\intercal$ where $\mathbf{L} \in \mathbb{R}^{6\times 6}$ is a lower triangle matrix. The sampled pose $\widetilde{\mathbf{X}}$ is then obtained by $\widetilde{\mathbf{X}} = \exp(\mathbf{L}\tilde{\boldsymbol{\xi}})\ \overline{\mathbf{X}}$ where $\tilde{\boldsymbol{\xi}} \in \mathbb{R}^6$ is sampled from the standard multivariate normal distribution $\mathcal{N}(0, \mathbf{1}^{6\times 6})$ and $\exp : \mathfrak{se}(3) \rightarrow SE(3)$ [30]. When the virtual depth views $\mathbf{d}$ are generated, the depth noise due to disparity [31] is added by

$$\sigma_z = \Big(\frac{m}{fb}\Big)\mathbf{d}^2\sigma_d \tag{2}$$

where f is the focal length of the sensor, b is the length of baseline between the infrared projector and camera, m is one of the parameters of a linear normalization, and σ_z and σ_d are the standard deviations of the triangulated depth and the normalized disparity, respectively[3].

B. PGNM for Depth

The set of virtual depth images $\mathcal{D} = \{\mathbf{d}^1, \mathbf{d}^2, \cdots, \mathbf{d}^{N_v}\}$ is obtained via the rendering with the sampled pose $\widetilde{\mathcal{S}} = \{\widetilde{\mathbf{X}}_1, \widetilde{\mathbf{X}}_2, \cdots, \widetilde{\mathbf{X}}_{N_v}\}$ where N_v is the number of rendered views. The mean and variance of each pixel from $\mathcal{D}$ are estimated by

$$\bar{\mathbf{d}} := \frac{1}{N_v}\sum_{i=1}^{N_v}\mathbf{d}^i \tag{3}$$

$$\mathbf{s}^2 := \frac{1}{N_v - 1}\sum_{i=1}^{N_v}(\mathbf{d}^i - \bar{\mathbf{d}})^2. \tag{4}$$

Note that $\bar{\mathbf{d}}$ and $\mathbf{s}^2$ are unbiased estimates of the mean and variance from the given samples.

C. PGNM for Surface Normal

Unit surface normal vectors reside on the unit sphere S^2 which is a 2D manifold in 3D Euclidean space,

$$S^2 = \{\mathbf{x} \in \mathbb{R}^3 \mid \|\mathbf{x}\|_2 = 1\}. \tag{5}$$

[3] The typical values of $\frac{m}{fb}$ and σ_d for RGB-D sensors are 2.85×10^{-3} and 0.5, respectively [31].

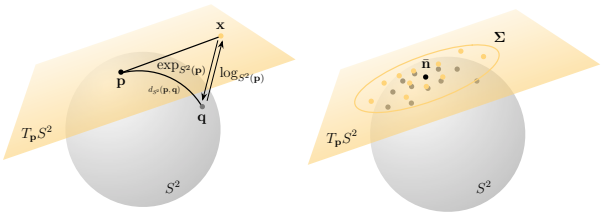


Fig. 5: **Mappings between S^2 and its tangent space $T_{\mathbf{p}}S^2$, and 2D Gaussian distribution on the $T_{\mathbf{p}}S^2$.** (Please see text for details.)

When distributions on S^2 are modeled, it is convenient to consider in a tangent plane, not directly on the manifold surface S^2 [32].

Let $\mathbf{p}$ and $\mathbf{q}$ be two points on S^2 as $\mathbf{p}^\intercal\mathbf{p} = \mathbf{q}^\intercal\mathbf{q} = 1$ and let $T_{\mathbf{p}}S^2$ represent the tangent space to S^2 at $\mathbf{p}$:

$$T_{\mathbf{p}}S^2 = \{\mathbf{x} \in \mathbb{R}^3 \mid \mathbf{x}^\intercal\mathbf{p} = 0\}. \tag{6}$$

The left image in Fig. 5 describes $\mathbf{p}$, $\mathbf{q}$, and $T_{\mathbf{p}}S^2$ on S^2. The geodesic distance on S^2 between $\mathbf{p}$ and $\mathbf{q}$ is defined by the angle between $\mathbf{p}$ and $\mathbf{q}$: $d_{S^2}(\mathbf{p},\mathbf{q}) = \arccos(\mathbf{p}^\intercal\mathbf{q})$.

The Riemannian exponential map, $\exp_{S^2(\mathbf{p})} : T_{\mathbf{p}}S^2 \to S^2$, maps a point $\mathbf{x}$ in the linear tangent space $T_{\mathbf{p}}S^2$ at the point $\mathbf{p}$ onto the unit sphere S^2:

$$\mathbf{x} \mapsto \mathbf{p}\cos(\|\mathbf{x}\|) + \frac{\mathbf{x}}{\|\mathbf{x}\|}\sin(\|\mathbf{x}\|) \tag{7}$$

and the Riemannian logarithm map, $\log_{S^2(\mathbf{p})} : S^2/\{-\mathbf{p}\} \to T_{\mathbf{p}}S^2$ which is the inverse of $\exp_{S^2(\mathbf{p})}$, can be obtained as:

$$\mathbf{q} \mapsto (\mathbf{q} - \mathbf{p}\cos\theta)\frac{\theta}{\sin\theta} \tag{8}$$

where $\theta = d_{S^2}(\mathbf{p},\mathbf{q})$. As $d_{S^2}(\mathbf{p},\mathbf{q})$ is geodesic distance, Euclidean distance $\|\mathbf{x}\|_2$ on $T_{\mathbf{p}}S^2$ is equivalent to $d_{S^2}(\mathbf{p},\mathbf{q})$ as $d_{S^2}(\mathbf{p},\mathbf{q}) = \|\mathbf{x}\|_2 = \|\log_{S^2(\mathbf{p})}(\mathbf{q})\|_2$. The geodesic distance and Riemannian maps between the unit sphere S^2 and its tangent space are described in [32] and its supplementary material. For further details of Riemannian geometry, please refer to [33].

PGNM estimates a distribution of unit surface normals on the tangent space $T_{\bar{\mathbf{n}}}S^2$ from $\mathcal{D}$. As $T_{\bar{\mathbf{n}}}S^2$ is the 2D subspace, the modeled Gaussian distribution would be 2D Gaussian distribution with the mean $\bar{\mathbf{n}}$[4] and covariance $\mathbf{\Sigma}$ as depicted in the right image of Fig. 5.

To estimate the surface normal from depth images $\mathcal{D}$, it is convenient to convert $\mathcal{D}$ to vertex images $\mathcal{V} = \{\mathbf{v}^1, \mathbf{v}^2, \cdots, \mathbf{v}^{N_v}\}$ where each vertex image $\mathbf{v}(\mathbf{p}) : \mathbb{R}^2 \to \mathbb{R}^3$ maps an image coordinates $\mathbf{p} \in \mathbb{R}^2$ to a 3D $\mathbf{xyz}$ vertex coordinates. The vertex image is obtained via

$$\mathbf{v}(\mathbf{p}) = \mathbf{d}(\mathbf{p})\mathbf{K}^{-1}[\mathbf{p}^\intercal\ 1]^\intercal$$

where $\mathbf{K} := \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{3\times3}$ is the intrinsic matrix of the depth sensor. Their associated normal images $\mathcal{N} = \{\mathbf{n}^1, \mathbf{n}^2, \cdots, \mathbf{n}^{N_v}\}$ where $\mathbf{n}(\mathbf{p}) : \mathbb{R}^2 \to S^2$ are calculated by the cross product followed by normalization as

$$\mathbf{n}(\mathbf{p}) = \frac{\mathbf{v}_u(\mathbf{p}) \times \mathbf{v}_v(\mathbf{p})}{\|\mathbf{v}_u(\mathbf{p}) \times \mathbf{v}_v(\mathbf{p})\|_2}$$

where $\mathbf{v}_u(\mathbf{p}) = \frac{\partial\mathbf{v}}{\partial u} = \mathbf{v}(\mathbf{p} + \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]) - \mathbf{v}(\mathbf{p})$ and $\mathbf{v}_v(\mathbf{p}) = \frac{\partial\mathbf{v}}{\partial v} = \mathbf{v}(\mathbf{p} + \left[\begin{smallmatrix}0\\1\end{smallmatrix}\right]) - \mathbf{v}(\mathbf{p})$ [26], [32].

Let $\tau_{\mathbf{p}}(\mathbf{x}) \in \mathbb{R}^2$ be the 2D coordinate of $\mathbf{x}$ on $T_{\mathbf{p}}S^2$. The mean $\bar{\mathbf{n}}_v \in S^2$ and the covariance $\mathbf{\Sigma} \in \mathcal{S}^2_+$ from the normal images $\mathcal{N}$ are estimated by

$$\bar{\mathbf{n}} := \frac{1}{N_v}\sum_{i=1}^{N_v}\mathbf{n}^i \tag{9}$$

$$\mathbf{\Sigma} := \frac{1}{N_v}\sum_{i=1}^{N_v}\left(\tau_{\bar{\mathbf{n}}}(\mathbf{n}^i) - \tau_{\bar{\mathbf{n}}}(\bar{\mathbf{n}})\right)\left(\tau_{\bar{\mathbf{n}}}(\mathbf{n}^i) - \tau_{\bar{\mathbf{n}}}(\bar{\mathbf{n}})\right)^\intercal \tag{10}$$

$$= \frac{1}{N_v}\sum_{i=1}^{N_v}\tau_{\bar{\mathbf{n}}}(\mathbf{n}^i)\tau_{\bar{\mathbf{n}}}(\mathbf{n}^i)^\intercal. \tag{11}$$

Note that $\tau_{\bar{\mathbf{n}}}(\bar{\mathbf{n}}) = \mathbf{0}$ as the origin of the $T_{\bar{\mathbf{n}}}S^2$ is $\bar{\mathbf{n}}$. It is also worth noting that $\tau_{\bar{\mathbf{n}}}(\mathbf{n}^i)$ can be estimated via rotating the origin of the plane $T_{\bar{\mathbf{n}}}S^2$ to the north pole $(0,0,1)^\intercal$ of the S^2 and the $\mathbf{x}$ and $\mathbf{y}$ coordinates of $\log_{S^2(\bar{\mathbf{n}})}(\mathbf{n}^i)$ correspond to $\tau_{\bar{\mathbf{n}}}(\mathbf{n}^i)$. Fig. 6 presents mean estimations of depth and surface normal of an example, and covariance images for the $\bar{\mathbf{n}}_+$ case are shown Fig. 7

[4]Note that the 2D coordinates of the mean $\bar{\mathbf{n}}$ on $T_{\bar{\mathbf{n}}}S^2$ are $\mathbf{0} \in \mathbb{R}^2$.

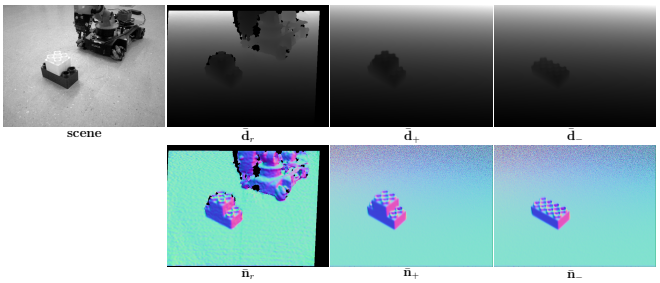


Fig. 6: **Depth and surface normal means.** The grayscale scene image shows the assembly scene after the assembly robot attached the new block (bright block) to the work-in-progress block (dark block). The depth $\bar{\mathbf{d}}_r$ and surface normal $\bar{\mathbf{n}}_r$ means are estimated from a set of real depth images $\mathcal{D}_r$. Once the current pose estimate $\mathcal{P} = \{\overline{\mathbf{X}}_0, \mathbf{\Sigma}_0\}$ is calculated from ICP, virtual PGNMs are estimated. Here only mean depth $(\bar{\mathbf{d}}_+, \bar{\mathbf{d}}_-)$ and normal $(\bar{\mathbf{n}}_+, \bar{\mathbf{n}}_-)$ are shown. Note artificially added depth noises which is more noticeable in normal mean images.

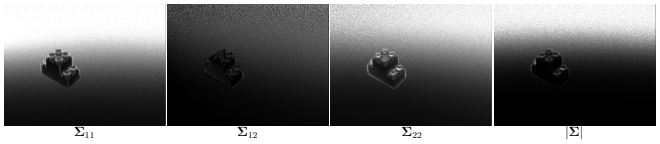


Fig. 7: **Covariance images from the surface normal PGNM.** Covariance matrix elements from the 2D Gaussian surface normal distributions are shown. As $\mathbf{\Sigma}$ is always symmetric, the off diagonal element $\mathbf{\Sigma}_{21}$ is not shown here. Note that covariance elements are higher around the concave and convex areas on the object, since surface normals deviates more on these areas due to the pose uncertainties.

VI. Bayesian Classification

The primary goal of visual verification is to decide what is the most likely class given an input. Let K be the number of possible classes and $\mathcal{Z}$ be the given sensory input.

The problem is then formulated as estimating a posterior probability for each case $p(C_k|\mathcal{Z})$ where $k \in \{1, 2, \cdots, K\}$. As final decision, a maximum *a posteriori* (MAP) estimation of $\hat{C}$ is considered as

$$\hat{C} = \underset{k\in\{1,\cdots,K\}}{\arg\max}\ p(C_k|\mathcal{Z}). \tag{12}$$

The simplest case is the binary decision ($K = 2$) in which the classes are either *success* or *failure*. More general cases consider K possible outcomes with their associated probabilities as

$$p(C_k) = \lambda_k \in [0, 1], \sum_{k=1}^{K} \lambda_k = 1.$$

This distribution is known as categorical distribution $p(C) = \mathbf{Cat}_C(\boldsymbol{\lambda})$ or multinomial distribution. The Bernoulli distribution is a special case when $K = 2$.

By Bayes' theorem, the posterior probability can be expressed via the conditional probability and the prior as:

$$p(C_k|\mathcal{Z}) = \frac{p(\mathcal{Z}|C_k)p(C_k)}{p(\mathcal{Z})} = \frac{p(\mathcal{Z}|C_k)p(C_k)}{\sum_i p(\mathcal{Z}|C_i)p(C_i)}. \tag{13}$$

The denominator, $p(\mathcal{Z}) = \sum_i p(\mathcal{Z}|C_i)p(C_i)$, does not depend on the class C_k, and thus (12) is to choose the highest unnormalized posterior as

$$\hat{C} = \underset{k\in\{1,\cdots,K\}}{\arg\max}\ p(\mathcal{Z}|C_k)p(C_k). \tag{14}$$

The conditional probability distribution $p(\mathcal{Z}|C_k)$ is learned from a set of training data as shown Section V. The sensory measurements $\mathcal{Z}$ are composed of depth and surface normal, $\mathcal{Z} = \{\mathcal{Z}_d, \mathcal{Z}_n\}$. By the naive conditional independence assumption [22], the conditional density can be split as:

$$p(\mathcal{Z}|C_k) = p(\mathcal{Z}_d|C_k)p(\mathcal{Z}_n|C_k). \tag{15}$$

The conditional probability distribution of depth measurement $\mathcal{Z}_d$ is defined by the mean (3) and the variance (4) of the 1D Gaussian distribution in Section V-B as

$$p(\mathcal{Z}_d|C_k) = \mathcal{N}(\bar{\mathbf{d}}_r; \bar{\mathbf{d}}_{v_k}, \mathbf{s}^2_{v_k}) \tag{16}$$

$$= \frac{1}{\sqrt{2\pi\mathbf{s}^2_{v_k}}} \exp\Big(-\frac{(\bar{\mathbf{d}}_r - \bar{\mathbf{d}}_{v_k})^2}{2\mathbf{s}^2_{v_k}}\Big). \tag{17}$$

Similarly, the conditional probability distribution of normal measurement $\mathcal{Z}_n$ is defined by the mean (9) and the covariance (11) of the 2D Gaussian distribution in Section V-C as

$$p(\mathcal{Z}_n|C_k) = \mathcal{N}(\tau_{\bar{\mathbf{n}}_{v_k}}(\bar{\mathbf{n}}_r); \tau_{\bar{\mathbf{n}}_{v_k}}(\bar{\mathbf{n}}_{v_k}), \boldsymbol{\Sigma}_{v_k}) \tag{18}$$

$$= \mathcal{N}(\tau_{\bar{\mathbf{n}}_{v_k}}(\bar{\mathbf{n}}_r); \mathbf{0}, \boldsymbol{\Sigma}_{v_k}) \tag{19}$$

$$= \frac{\exp\Big(-\frac{1}{2}\tau_{\bar{\mathbf{n}}_{v_k}}(\bar{\mathbf{n}}_r)^\intercal \boldsymbol{\Sigma}^{-1}_{v_k} \tau_{\bar{\mathbf{n}}_{v_k}}(\bar{\mathbf{n}}_r)\Big)}{\sqrt{(2\pi)^2|\boldsymbol{\Sigma}_{v_k}|}}. \tag{20}$$

Like our model, when the probability distributions of the feature $\mathcal{Z}$ are Gaussian, the classifier is called Gaussian Naive Bayes (GNB) classifier [22]. In many machine learning applications, these conditional probability distributions or likelihood are learned from training data. In our case, it is intractable as the depth and surface normal measurements not only depends on an object's shape but also are subject to the pose of the object with respect to the sensor. Thus we generate this statistical models online using 3D shape prior.

A. Classification and Confidence

The Bayesian classification in the previous section is done in each pixel. The visual verification, however, is not just for one pixel measurement, but for a verification region. In this section the final classification result as well as the confidence are estimated by aggregating the per-pixel classification results on the region.

Let $\mathcal{R} \subset \mathbb{R}^2$ be the inspection region on the depth image, and $\mathbf{r} \in \mathcal{R}$ is 2D image coordinates in the region. The region $\mathcal{R}$ is determined by rendering the new part model $\mathcal{M}_t$ with the mean pose estimate $\overline{\mathbf{X}}_t = \overline{\mathbf{X}}_{0:t-1}\overline{\mathbf{X}}_t^{0:t-1}$. The classification outcome of each pixel $\mathbf{r}$ is the MAP class $\hat{C}(\mathbf{r})$ and its confidence $p(\hat{C}(\mathbf{r})|\mathcal{Z})$.

A simple approach to fuse a set of pixel-wise estimations is to count the number of each class as

$$p(C_k|\mathcal{Z})_{\mathcal{R}} = \frac{\sum_{\mathbf{r}}^{|\mathcal{R}|} \mathbb{I}(\hat{C}(\mathbf{r}) = k)}{\sum_{i=1}^{K}\sum_{\mathbf{r}}^{|\mathcal{R}|} \mathbb{I}(\hat{C}(\mathbf{r}) = i)} = \frac{1}{|\mathcal{R}|}\sum_{\mathbf{r}}^{|\mathcal{R}|} \mathbb{I}(\hat{C}(\mathbf{r}) = k)$$

where $\mathbb{I}(\cdot)$ is the indicator function. This value, however, does not take into account the pixel-wise confidence $p(\hat{C}(\mathbf{r})|\mathcal{Z})$ as it is solely estimated by the MAP class $\hat{C}(\mathbf{r})$. A better amalgamation would be considering the confidence as

$$p(C_k|\mathcal{Z})_{\mathcal{R}} = \frac{\sum_{\mathbf{r}}^{|\mathcal{R}|} p(\hat{C}(\mathbf{r})|\mathcal{Z}) \cdot \mathbb{I}(\hat{C}(\mathbf{r}) = k)}{\sum_{i=1}^{K}\sum_{\mathbf{r}}^{|\mathcal{R}|} p(\hat{C}(\mathbf{r})|\mathcal{Z}) \cdot \mathbb{I}(\hat{C}(\mathbf{r}) = i)} \tag{21}$$

and the final MAP class and probability for the $\mathcal{R}$ region is determined as:

$$\hat{C}_{\mathcal{R}} = \underset{k\in\{1,\cdots,K\}}{\arg\max}\ p(C_k|\mathcal{Z})_{\mathcal{R}}, \quad \hat{p}_{\mathcal{R}} = p(\hat{C}_{\mathcal{R}}|\mathcal{Z})_{\mathcal{R}}.$$

VII. Experiments

In this section we present experimental results in which our approach (GNB with depth and normal, GDN) is compared with the ray tracing-based visual inspection approach (WO) [28]. Our approach is also evaluated with two variants of our Bayesian classification approach, one using only depth (GNB with depth, GD) and the other one using only the normals (GNB with normal, GN) of the PGNM, to see how these depth and normal are individually effective in this problem. Considered assembly experiments are the first step of both assemblies shown in Fig. 4 where $\mathcal{M}_1$ is assembled to $\mathcal{M}_0$ with the placement pose $\overline{\mathbf{X}}_1^0$. The pose $\mathcal{P}_0 = \{\overline{\mathbf{X}}_0, \boldsymbol{\Sigma}_0\}$ of the work-in-progress $\mathcal{M}_0$ is given by the ICP explained in Section IV-C.

We evaluate the four approaches with respect to:

- How do they treat ambiguous cases.

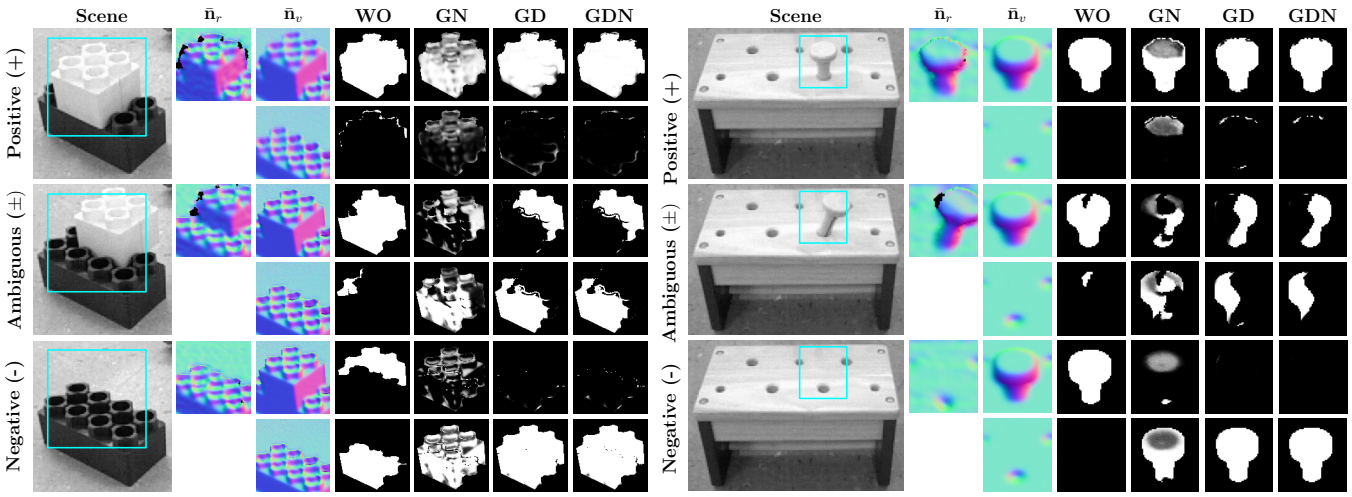
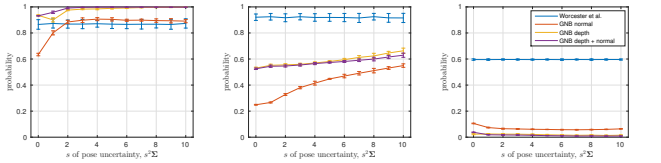
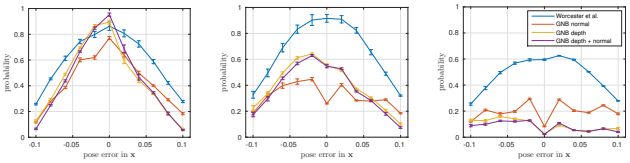


Fig. 8: **Per-pixel probability outputs for three cases (+, ±, −) of the block assembly (left) and the peg-in-hole task (right).** From top to bottom, the scene images show the positive case (+) which is the correct assembly, the ambiguous case (±) where the part is imprecisely assembled, and the negative case (−) in which the part is completely missing. The mean images of surface normal from real $\bar{\mathbf{n}}_r$ and virtual $\bar{\mathbf{n}}_v$ depth are presented next to the scene images. Gray scale images of the four columns to the right represent the probability $p(C_k|\mathcal{Z})$ where the first and second rows for each case depict $p(C_+|\mathcal{Z})$ and $p(C_-|\mathcal{Z})$, respectively. Note that the per-pixel probabilities of WO are binary as it estimate true or false for each pixel. (Best viewed in color)



(a) Positive (+) (b) Ambiguous (±) (c) Negative (−)

Fig. 9: **Probability vs. Pose Uncertainty.** The initial pose uncertainty $\mathbf{\Sigma}$ is gradually increased to see how the four approaches work under the uncertainty. The $\mathbf{y}$-axis represents $p(C_+|\mathcal{Z})$, and hence the plots should be around 1, 0.5, and 0 for the (+), (±), and (−), respectively. Note that [28] estimates as (+) case in the (−) case and quite erroneously confident in the (±) case. (Best viewed in color)



(a) Positive (+) (b) Ambiguous (±) (c) Negative (−)

Fig. 10: **Probability vs. Pose Error.** Given a pose estimate $\overline{\mathbf{X}}$, the pose is translated in $\mathbf{x}$-axis from -10 cm to 10 cm. New $\mathbf{\Sigma}$ estimate was determined for each perturbed pose. The more the pose estimate gets off, the lower the probability $p(C_+|\mathcal{Z})$ is. (Best viewed in color)

- Robustness to pose uncertainties.
- Reliability in estimating the probability with pose offset (i.e. inaccurate pose estimation).

Since all GNB-based visual verification (GDN, GD, and GN) are stochastic, we run multiple times for each evaluation[5] and draw errorbar plots which illustrate mean and standard deviation of the multiple runs.

A. Ambiguous Cases

In this experiment we examine the classification accuracy of the four approaches in three different cases. For all approaches, only two classes were assumed: positive (+, a new part is assembled as expected) and negative (−, the part is entirely missing). Two depth scenes were captured for the two classes and one ambiguous scene (±, the part is placed in a wrong placement) was obtained for test purpose.

Fig. 8 shows per-pixel probability outputs for three cases. For each case, WO, GN, GD, and GDN are presented from left to right; the first row presents $p(C_+|\mathcal{Z})$, while the second row shows $p(C_-|\mathcal{Z})$. When $\mathcal{M}_1$ is well assembled, all approaches show high probabilities in $p(C_+|\mathcal{Z})$ and consequently low probabilities in $p(C_-|\mathcal{Z})$ as $p(C_-|\mathcal{Z}) = 1 - p(C_+|\mathcal{Z})$. When the surface normal distribution is only considered (GN), the probability estimates are not accurate in some regions where surface normals are similar to that of backgrounds (upper planar areas on both the block and the peg).

If $\mathcal{M}_1$ is completely missing (−), all GNB approaches return high probabilities for (−) case. WO is discouraging in this case, since it is misled by the foreground region which actually belongs to $\mathcal{M}_0$, not $\mathcal{M}_1$. GD and GDN are quite similar, while GN is slightly worse. It is clear that depth measurement are favorable in these examples as GD and GDN correctly classify (+) or (−) classes. A distinction between GD and GDN is not observed in this experiment, but we will show it in the following sections.

B. Robustness to Pose Uncertainty

In this section we evaluate how our approach is robust to the uncertainty in the pose of the object. To analyze robustness to pose uncertainty, the pose uncertainty $\mathbf{\Sigma}$ was gradually raised. Fig. 9 presents plots for (+), (±), and (−) cases. As $\mathbf{y}$-axis of the each plot represents $p(C_+|\mathcal{Z})$, the plots are expected to gather around 1, 0.5, and 0 for the (+), (±), and (−) cases, respectively.

Even though the uncertainty increases, the probability estimations of all approaches do not deviate significantly. WO is

[5]We run 10 times in this experiments.

nearly static as it does not take into account uncertainties. Our approach and two variants exhibit similar trend; they are steady after $2^2\boldsymbol{\Sigma}$ for both $(+)$ and $(-)$ cases. When there is no uncertainty ($s = 0$), GN degrades because conditional probabilities in (19) are getting too narrow. It is an expected phenomenon as zero uncertainty invalidates our probabilistic formulation. GD also deteriorates at zero uncertainty, but the ratio is less significant compared to GN. GDN is quite similar to GD but more accurate by considering both depth and normal.

C. Robustness to Pose Offset

In this experiment we compare the robustness to pose error. Since the pose uncertainty $\boldsymbol{\Sigma}$ is increasing according to (1) as the offset of the pose $\overline{\mathbf{X}}$ is increasing, we estimate $\boldsymbol{\Sigma}$ for each perturbed pose. Fig. 10 shows their probability responses with respect to pose error in $\mathbf{x}$-axis. All approaches show lower $p(C_+|\mathcal{Z})$ as pose error increases. In terms of performance, GDN comes first followed by GD. GN follows similar trend, but it seems to vary more with respect to the pose error. WO shows discouraging performance as in Section VII-B. No matter what the case is, WO always returns probabilities favoring C_+ within ± 5 cm in this experiment.

VIII. Conclusion

A depth sensor-based visual verification approach was presented that can be applied from generic robotic manipulation to robotic assembly in flexible/versatile settings. By exploiting prior knowledge of the shapes of objects and their assembly configurations, we formulated the problem as a Bayesian classification wherein the PGNM for both depth and surface normal plays an important role to robustly estimate the maximum likely class and its confidence. Our approach was evaluated in a set of comparative experiments.

References

[1] M. T. Mason, *Mechanics of robotic manipulation*. Cambridge, Mass.: MIT Press, 2001.

[2] C. C. Kemp, A. Edsinger, and E. Torres-Jara, "Challenges for robot manipulation in human environments," *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, p. 20, 2007.

[3] R. T. Chin and C. A. Harlow, "Automated visual inspection: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 557–573, 1982.

[4] T. S. Newman and A. K. Jain, "A survey of automated visual inspection," *Computer Vision and Image Understanding*, vol. 61, no. 2, pp. 231–262, 1995.

[5] E. N. Malamas, E. G. Petrakis, M. Zervakis, L. Petit, and J.-D. Legat, "A survey on industrial vision systems, applications and tools," *Image and Vision Computing*, vol. 21, no. 2, pp. 171–188, 2003.

[6] D. Bourne, "My boss the robot," *Scientific American*, vol. 308, no. 5, pp. 38–41, 2013.

[7] R. C. Bolles, "Verification vision within a programmable assembly system," Ph.D. dissertation, Stanford, 1976.

[8] ——, "Verification vision for programmable assembly." in *Proc. Int'l Joint Conf. Artificial Intelligence (IJCAI)*, vol. 2, 1977, p. 569.

[9] F. Prieto, T. Redarce, R. Lepage, and P. Boulanger, "An automated inspection system," *The International Journal of Advanced Manufacturing Technology*, vol. 19, no. 12, pp. 917–925, 2002.

[10] T. S. Newman and A. K. Jain, "A system for 3d CAD-based inspection using range images," *Pattern Recognition*, vol. 28, no. 10, pp. 1555–1574, 1995.

[11] C. Harris, *Tracking with Rigid Objects*. MIT Press, 1992.

[12] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[13] H. Barrow, J. Tenenbaum, R. Bolles, and H. Wolf, "Parametric correspondence and chamfer matching: Two new techniques for image matching," in *Proc. Int'l Joint Conf. Artificial Intelligence (IJCAI)*, 1977, pp. 659–663.

[14] A. Pinz, "Object categorization," *Foundations and Trends in Computer Graphics and Vision*, vol. 1, no. 4, pp. 255–353, 2005.

[15] V. Lepetit and P. Fua, "Keypoint recognition using randomized trees," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1465–1479, 2006.

[16] A. Collet, M. Martinez, and S. S. Srinivasa, "The MOPED framework: Object recognition and pose estimation for manipulation," *International Journal of Robotics Research*, vol. 30, no. 10, pp. 1284–1306, 2011.

[17] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 239–256, 1992.

[18] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proceedings of International Conference on 3-D Digital Imaging and Modeling (3DIM)*, 2001, pp. 145–152.

[19] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, "A Bayesian approach to filtering junk e-mail," in *Learning for Text Categorization: Papers from the 1998 workshop*, vol. 62, 1998, pp. 98–105.

[20] M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection," *International Journal of Computer Vision*, vol. 46, no. 1, pp. 81–96, 2002.

[21] A. Elgammal, R. Duraiswami, D. Harwood, and L. S. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," *Proceedings of the IEEE*, vol. 90, no. 7, pp. 1151–1163, 2002.

[22] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. The MIT Press, Aug. 2012.

[23] M. Germann, M. D. Breitenstein, H. Pfister, and I. K. Park, "Automatic pose estimation for range images on the GPU," in *Proceedings of International Conference on 3-D Digital Imaging and Modeling (3DIM)*, 2007, pp. 81–90.

[24] C. Choi, Y. Taguchi, O. Tuzel, M.-Y. Liu, and S. Ramalingam, "Voting-based pose estimation for robotic assembly using a 3D sensor," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2012.

[25] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, "RGB-D mapping: Using depth cameras for dense 3D modeling of indoor environments," in *Proc. Int'l Symposium on Experimental Robotics (ISER)*, 2010.

[26] R. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-time dense surface mapping and tracking," in *Proc. Int'l Symposium on Mixed and Augmented Reality (ISMAR)*, 2011, pp. 127–136.

[27] C. Choi and H. I. Christensen, "RGB-D object tracking: A particle filter approach on GPU," in *Proc. IEEE/RSJ Int'l Conf. Intelligent Robots Systems (IROS)*, 2013, pp. 1084–1091.

[28] J. Worcester, M. A. Hsieh, and R. Lakaemper, "Distributed assembly with online workload balancing and visual error detection and correction," *International Journal of Robotics Research*, vol. 33, no. 4, pp. 534–546, 2014.

[29] O. Bengtsson and A.-J. Baerveldt, "Robot localization based on scan-matching—estimating the covariance matrix for the IDC algorithm," *Robotics and Autonomous Systems*, vol. 44, no. 1, pp. 29–40, 2003.

[30] T. D. Barfoot and P. T. Furgale, "Associating Uncertainty With Three-Dimensional Poses for Use in Estimation Problems," *IEEE Transactions on Robotics*, 2014.

[31] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," *Sensors*, vol. 12, no. 2, pp. 1437–1454, 2012.

[32] J. Straub, G. Rosman, O. Freifeld, J. J. Leonard, and J. W. Fisher, "A mixture of Manhattan frames: Beyond the Manhattan world," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2014, pp. 3770–3777.

[33] M. P. do Carmo Valero, *Riemannian geometry*. Birkhuser, 1992.