

Generation of Independent Contact Regions on Objects Reconstructed from Noisy Real-World Range Data

Krzysztof Charusta, Robert Krug, Todor Stoyanov, Dimitar Dimitrov and Boyko Iliev
Center of Applied Autonomous Sensor Systems (AASS), Örebro University, Sweden
Email: {krzysztof.charusta, robert.krug, todor.stoyanov, dimitar.dimitrov, boyko.iliev}@oru.se

Abstract—The synthesis and evaluation of multi-fingered grasps on complex objects is a challenging problem that has received much attention in the robotics community. Although several promising approaches have been developed, applications to real-world systems are limited to simple objects or gripper configurations. The paradigm of Independent Contact Regions (ICRs) has been proposed as a way to increase the tolerance to grasp positioning errors. This concept is well established, though only on precise geometric object models. This work is concerned with the application of the ICR paradigm to models reconstructed from real-world range data. We propose a method for increasing the robustness of grasp synthesis on uncertain geometric models. The sensitivity of the ICR algorithm to noisy data is evaluated and a filtering approach is proposed to improve the quality of the final result.

I. INTRODUCTION

Robotic grasping of unfamiliar objects is still a serious challenge despite many years of research. In a typical robotic application, a manipulator equipped with a specific gripper (e.g. parallel jaws) can grasp various objects, relying on a set of known geometric properties. This strategy is usually applied to simple object shapes, due to the limited functionality of the gripper. Similarly, a number of approaches address more complex shapes by partitioning the target object into simple primitives [1]. When multi-fingered robotic hands are employed to grasp non-trivial objects, a common practice is to perform grasp synthesis on precise geometric models. Such models might be available in industrial applications but in unstructured environments the object model has to be built from sensor data. Although methods for reconstructing 3D shapes from sensor data exist, the problem of grasp synthesis on uncertain object models has not been thoroughly investigated.

Grasping methods that operate in conjunction with online sensing and model reconstruction can be roughly divided in two categories, depending on the type of the object model. One group of approaches use computer vision to extract grasp-related object features and compute appropriate grasp configurations. Morales et al. [2] use a vision based approach for the generation of planar grasps from 2D object contours. The authors also assess the reliability of the grasps using an experience database. Another approach based on visual features proposed by Saxena et. al. [3] learns grasps for a parallel gripper, based on 2D images. Both of these prior works however are not suitable for grasping of complicated 3D objects with a multi-fingered robotic hand. The second major group of methods concentrate on generating grasps

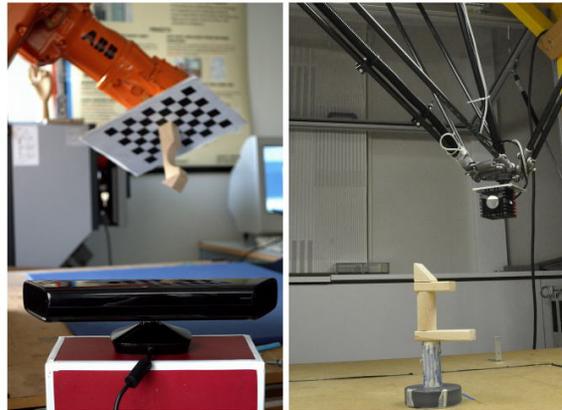


Fig. 1. Acquisition system setups: (Left) A Kinect camera observing an object mounted on the end-effector of an ABB IRB140 manipulator; (Right) An example object is scanned by a SwissRanger SR4000 camera mounted on an ABB IRB340 manipulator.

for reconstructed 3D object models. For example, Bone et al. [4] present a vision based method which creates 3D models of unknown objects and generates a force-closure grasp. However, the method is limited to objects which can be grasped by a parallel-jaw gripper. Another method for grasping of unknown 3D objects is proposed by Hübner and Kragic [5]. They obtain a 3D point cloud from a stereo camera and decompose it into a constellation of boxes which are then used for grasp synthesis. This approach however relies on the existence of sufficiently similar pre-learned grasp primitives and thus does not apply to arbitrary objects.

Analytical approaches to multi-fingered grasp generation rely on perfect knowledge of the object geometry. As a result, generated grasps would guarantee the satisfaction of a desired quality criterion only if the robot hand grasps the object exactly at the prescribed contact points. In light of this fact, it has been recognized that an equally important property of a grasp is its robustness to positioning inaccuracies, *i.e.*, grasps that are less sensitive to modeling and positioning errors are desired. In this context the notion of Independent Contact Regions (ICRs) was suggested by Nguyen [6]. He defined the set of optimal independent regions with the largest minimal radius, which yield a force-closure grasp if each finger is placed anywhere within its respective region. The concept was extended to the computation of independent regions for three-finger grasps on planar objects [7] and four-finger grasps of polyhedral objects by Ponce et al. [8]. In [9], Pollard addresses the synthesis of whole-hand grasps

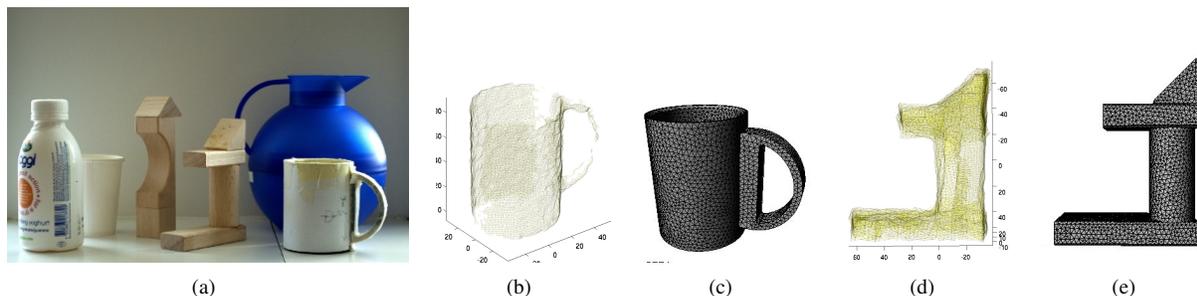


Fig. 2. *Considered target objects:* (a) The selected set of objects whose models were used in the evaluation; (b) & (d) Reconstructed meshes based on unfiltered point clouds acquired by range sensing; (c) & (e) Corresponding ground-truth CAD-model meshes; Note the surface distortions and double layers on the sensor-acquisition based meshes;

on 3D objects based on a prototype force-closure grasp and introduces an approach that can account for a user specified task related quality measure. Roa and Suárez [10] suggested an approach, which grows independent regions for precision grasps on discretized objects. In a previous work [11], the authors presented an efficient algorithm for the computation of such independent regions.

The contribution of this work is two-fold: firstly, it investigates the sensitivity of the ICR generation algorithm to uncertainty in the target object’s geometric model. Secondly, a preprocessing method is proposed and the quality of the resulting ICRs is evaluated. Applied to noisy real-world object models, our method significantly improves the performance and the robustness of the ICR algorithm.

This paper is organized as follows: the next section provides an overview of the reconstruction of 3D shape models, the particular sensor setup used in this article and the algorithm used for ICR generation. Section III provides an analysis of the sensitivity of the ICR algorithm to model noise and a description of the proposed filtering approaches. Section IV presents the ICR quality evaluation methodology utilized in this paper. Section V discusses the experimental results on several test objects. Finally, we conclude with a summary of the main results and the limitations of the approach.

II. BACKGROUND & MOTIVATION

Geometrically motivated grasp synthesis algorithms for articulated multi-fingered manipulators, including the ICR approach [11] considered in this work, usually assume the availability of precise models of the target objects. Models, usually represented as polygonal meshes, can be produced either by a human designer or acquired automatically through expensive scanning equipment [12]. While such an approach is reasonable for applications in traditional static factory automation, in many interesting scenarios accurate object models are not available. One prominent example is the mobile manipulation task that usually requires on-board sensing capabilities and online reconstruction of object models [13].

We are interested in evaluating the ICR paradigm on target object models synthesized from noisy range-data and suitable preprocessing methods of these models. Online model acquisition requires many sub-problems to be solved, such as on-the-fly scan segmentation, scan registration, view-planning and meshing [14]. This a complex topic and out of the scope

of this paper. Here, the model acquisition was carried out in a straightforward manner described in the following section.

A. 3D Shape Reconstruction from Range Measurements

A multitude of light and relatively inexpensive range sensors are currently available for use in the robotics community. In this work, two range sensors – a Microsoft Kinect structured light camera and a SwissRanger SR4000 Time of Flight (ToF) camera, have been used to sample points from a set of target objects. The Kinect is an off-the-shelf sensor that provides VGA-size, relatively high quality depth images. Its measurement accuracy was assessed to be satisfactory for object modeling and thus the Kinect range data was used directly, without performing further calibration routines. In contrast, the accuracy of the SR4000 sensor is affected by several error sources, typical for a ToF-based measurement system. Satisfactory precision can however be achieved using relatively straightforward calibration procedures (see e.g. Fuchs et. al. [15]). Even more precise surface reconstruction with a ToF sensor is possible [14] using dynamic error models and non-rigid scan registration, though at greater computational costs.

The two aforementioned scanners have been used in the two different acquisition setups shown in Fig. 1. The SR4000 camera was mounted on the end-effector of an ABB IRB340 industrial manipulator. The range sensor was calibrated to compensate for lens distortions using a checkerboard pattern and standard image rectification algorithms available in the OpenCV [16] library. Additional thresholding and filtering were employed to reduce common illumination induced ToF errors, in a manner similar to the calibration procedures in [15]. In order to obtain a complete object scan, the manipulator was programmed to move around the target object, placed in the center of its workspace. In the second sensor set up, the Kinect camera was placed in a fixed position and oriented towards the workspace of an ABB IRB140 industrial manipulator (see Fig. 1). A checkerboard-textured support was fixed to the manipulator end-effector. Target objects were then mounted on the support and rotated around a single axis. The initial pose of the camera relative to the checkerboard pattern was estimated using standard computer vision procedures.

In both setups, the relative pose between sensor and target was obtained using the end-effector pose. All sensor measurements were then fused in a single reference frame,

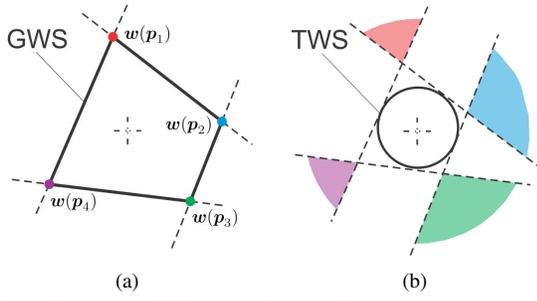


Fig. 3. *Concept of ICR-generation:* (a) Wrenches associated with the contact points \mathbf{p}_n of the initial four-fingered prototype grasp and its corresponding grasp wrench space; (b) The TWS represents the set of wrenches necessary to counterbalance expected disturbances; Hyperplanes associated with facets of the initial GWS are translated in parallel until they are tangent to the TWS; Search spaces (the colored shaded areas) in the wrench space are formed by the intersection of exterior half-spaces corresponding to the shifted hyperplanes (a half-space is designated as exterior if it does not contain the origin, opposed to an interior half-space which contains the origin); A point \mathbf{p}_s on the object’s boundary qualifies for inclusion in region \mathcal{C}_n if its associated wrench lies in the search space associated with \mathbf{p}_n ;

resulting in a complete object point cloud. Subsequently, triangular mesh representations were reconstructed from these point clouds using the ball pivoting surface reconstruction algorithm [17]. The output of this model acquisition procedure is of comparable quality to the results reported in [15], but significantly less accurate than the approach of Cui et. al. [14]. Notably, the accuracy of the final models is similar to the one to be expected in a typical on-line manipulation task, and thus the acquired models constitute a valid target evaluation scenario. Figures 2 (b) & (d) show two geometrical models and the corresponding triangular meshes reconstructed from range data.

B. Independent Contact Regions

Here, we give a brief overview of the algorithm used to compute the independent regions. For more details, the reader is referred to [11]. The computation requires a discretized representation of the target object’s surface as a polygonal mesh of points \mathbf{p}_s ($s = 1, \dots, S$) with corresponding inward-pointing unit normals \mathbf{n}_s . Each point \mathbf{p}_s has associated neighboring points, connected to \mathbf{p}_s by an edge of the polygonal mesh. A triangle in a mesh is defined by the set $\mathcal{T} = \{\mathbf{p}_i, \mathbf{p}_j, \mathbf{p}_k\}$, where the elements of \mathcal{T} are neighboring vertices.

A grasp is defined as a set of N contact points on the object’s surface $\mathcal{G} = \{\mathbf{p}_1, \dots, \mathbf{p}_N\}$. Contact forces \mathbf{f}_s and the resulting torques $\boldsymbol{\tau}_s = \mathbf{p}_s \times \mathbf{f}_s$ are concatenated to wrench vectors $\mathbf{w}_s = (\mathbf{f}_s, \boldsymbol{\tau}_s)$. Depending on the deployed contact model, \mathbf{f}_s has to satisfy certain constraints as described in [11]. Eligible contact models are point contact with/without friction and soft finger point contact. The convex hull over the set of all wrenches that a given grasp can apply to the object is referred to as the *Grasp Wrench Space* (GWS).

The utilized algorithm requires two forms of user-input. First, a set of expected disturbance wrenches needs to be specified. The convex hull over the mirror image of this set is commonly labeled as *Task Wrench Space* (TWS). In this

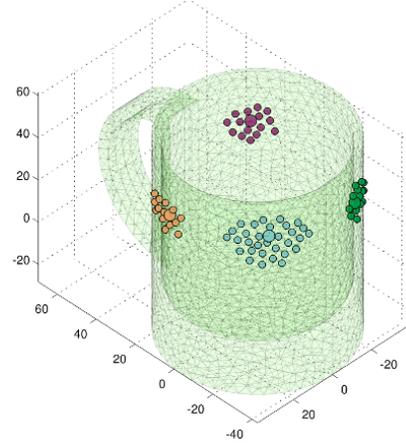


Fig. 4. *Independent Contact Regions:* ICR computed on the CAD-generated mesh of a cup for a four-fingered grasp comprising frictional point contacts; Big dots indicate the initial grasp contact points \mathbf{p}_n ; Finger placement anywhere within the regions ensures that the task requirement is fulfilled;

work we formulate the TWS as a sphere, parametrized by a fraction of the radius of the largest origin-centered insphere of the prototype grasp’s GWS. This relates to the grasp quality measure proposed by Kirkpatrick et al. [18]. Second, an initial prototype force-closure grasp which satisfies the task requirement needs to be available. In a typical scenario, this prototype grasp is provided by a human demonstrator [19] or by a grasp planning algorithm [20].

ICRs are defined as N independent regions \mathcal{C}_n , each one associated with a contact point \mathbf{p}_n of the prototype grasp. The sets \mathcal{C}_n contain points on the target object’s surface, each of which can replace \mathbf{p}_n in \mathcal{G} . By construction, any grasp composed of N contact points, where one point is picked from each region \mathcal{C}_n , will be force closure and preserve the task requirements. Essentially, the set containing these grasps constitutes a grasp family associated with the prototype grasp [9]. It is assumed that the target object is sufficiently discretized to capture local curvature, *i. e.*, grasps with contacts on mesh facets spanned by the points forming the regions \mathcal{C}_n will also guarantee the task requirements. The mesh area formed by the set of all triangles $\{\mathcal{T}_n\}$ stemming from neighboring points which are members of \mathcal{C}_n is denoted as $A(\mathcal{C}_n)$.

The basic idea behind the deployed algorithm for computing ICRs is shown in Fig. 3. Although the results in Section V were generated with grasps utilizing the frictional point contact model, for ease of understanding, the computational principle is illustrated in a hypothetical 2D wrench space for a grasp with frictionless point contacts. Regions \mathcal{C}_n are grown from the contact points \mathbf{p}_n associated with the prototype grasp. The number and distribution of points forming these regions depend on the geometry of the GWS corresponding to the provided initial prototype force-closure grasp, and the disturbances considered via the TWS. Figure 4 shows an example of ICR computed via the outlined algorithm. An extensive benchmark of the algorithm can be found in [11], where it is shown that run times in the envisioned setting are typically well below one second on a standard PC.

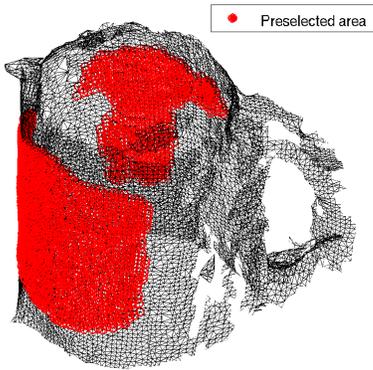


Fig. 5. *Filtering*: Example mesh generated after smoothing the raw point cloud (obtained via the SR4000 TOF-camera) according to Algorithm 1;

III. ICR ON RECONSTRUCTED OBJECTS

A. Noise Sensitivity

A point on the object’s boundary qualifies as a member of an independent contact region \mathcal{C}_n , if wrenches associated to that point are inside respective search regions in the wrench space (see Fig. 3-(b)). This constitutes a demanding constraint which makes growing of ICRs very sensitive to the noise in point position and vertex normal direction and relates to several distinctive errors occurring in the reconstructed meshes. First, due to the high levels of random noise, the reconstructed meshes exhibit rough surfaces, whereas the original object is piecewise smooth. Two other sources of errors are significant to the application scenario discussed – the occurrence of layered surfaces and holes in the meshes. Both of these artifacts are induced by sensor noise present in the sampled point clouds.

We generated CAD-model meshes for all target objects, which were treated as ground-truth for comparison purposes. Our early experiments showed that regions computed on sensor-acquisition based meshes are typically much smaller than corresponding ones on CAD-model meshes, thus making the usefulness of such independent regions questionable. In order to address these issues and to compensate for the model errors induced by sensing, a filtering method, which is carried out before meshing the raw point cloud, has been developed. It consists of two steps:

- 1) Point cloud smoothing,
- 2) Preselection of points.

Both techniques are explained in the following section.

B. Filtering strategy

1) *Point cloud smoothing*: The first step is inspired by a technique originating in the field of mobile robotics, namely the Three-Dimensional Normal Distributions Transform (3D-NDT). The 3D-NDT is a spatial representation paradigm originally developed by Magnusson et. al. [21] for use in point set registration. Andreasson et. al. [22] note that in most real world environments sampled surfaces can be represented accurately by flat, disc-shaped Gaussian probability distribution functions (PDF).

Algorithm 1 *Point cloud smoothing*

- 1: **Input:** Sets of points $\{\mathbf{p}_s\}$, sliding window radius r
 - 2: **Output:** Updated sets of points $\{\mathbf{p}_s^{new}\}$
 - 3: $\mathcal{W}_s = \emptyset$
 - 4: **for** all \mathbf{p}_s **do**
 - 5: center the window \mathcal{W}_s at \mathbf{p}_s and include neighboring points
 - 6: $\boldsymbol{\mu}_s, \boldsymbol{\Sigma}_s \leftarrow$ fit Gaussian PDF to \mathcal{W}_s
 - 7: $\lambda_s^{(i)}, \mathbf{v}_s^{(i)} \leftarrow$ eigenvalues and eigenvectors of $\boldsymbol{\Sigma}_s$
 - 8: update point: $\mathbf{p}_s = \mathbf{p}_s^{new}$ according to (1)
 - 9: **end for**
-

The proposed smoothing method is summarized in Algorithm 1. The presented technique employs a spherical sliding window \mathcal{W}_s with a pre-fixed constant radius r which iterates over all points \mathbf{p}_s in the point cloud. Following the central idea of the 3D-NDT, a Gaussian PDF $\mathcal{N}(\boldsymbol{\mu}_s, \boldsymbol{\Sigma}_s)$ is fitted to the point samples in each window \mathcal{W}_s . A three-dimensional normal PDF can be viewed as a confidence ellipsoid, whose size and orientation are determined by the covariance matrix $\boldsymbol{\Sigma}_s$. After eigenvalue decomposition of $\boldsymbol{\Sigma}_s$ we obtain eigenvalues $\lambda_s^{(i)} : i = 1, 2, 3$, $\lambda_s^{(1)} \geq \lambda_s^{(2)} \geq \lambda_s^{(3)}$, and associated orthonormal eigenvectors $\mathbf{v}_s^{(i)}$. Since the planes spanned by the two eigenvectors corresponding to the two dominant eigenvalues approximate the local surface shape, the point cloud is smoothed via projecting points \mathbf{p}_s onto the aforementioned planes:

$$\mathbf{p}_s^{new} = \mathbf{p}_s + \mathbf{v}_s^{(3)} (\boldsymbol{\mu}_s - \mathbf{p}_s)^T \mathbf{v}_s^{(3)}. \quad (1)$$

The aforementioned ball pivoting algorithm was used to generate meshes and vertex normals for the smoothed point cloud. Early experiments showed a significant improvement in terms of the size of the yielded contact regions \mathcal{C}_n . However, smoothing introduced another problem, namely the occurrence of false positive ICR-areas. Such areas appear when contact points are included in regions \mathcal{C}_n on a mesh reconstructed from range data, although they are not qualified for inclusion in the ground-truth regions yielded on a corresponding CAD-model mesh (see Fig. 6). False positive areas proved to arise frequently in object surface zones where the PDFs according to Algorithm 1 were spherical rather than flat. Due to the nature of the utilized scanning equipment, such non-flat ellipsoids are caused by high levels of sensor noise or fine details in the sampled surface, as well as corners between two surfaces. Thus, we developed a simple point preselection method which excludes points \mathbf{p}_s with a high uncertainty level from the ICR-computation as elaborated below.

2) *Point preselection*: The second proposed preprocessing step prohibits uncontrolled ICR growth into object regions that are hard to reconstruct using the previously presented smoothing technique. Points excluded in this selection process cannot belong to any region \mathcal{C}_n and thus cannot be part of the initial grasp \mathcal{G} . The selection is based on the following two criteria:

- cardinality of \mathcal{W}_s ,

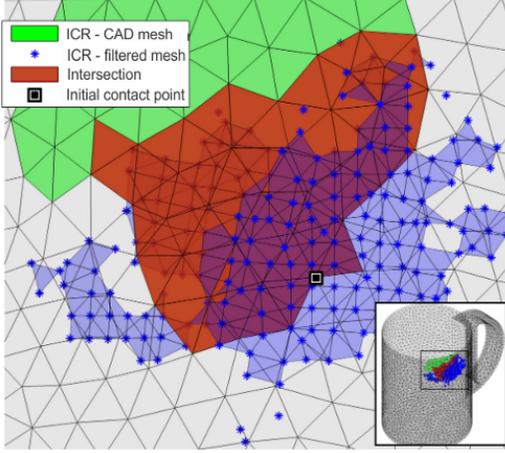


Fig. 6. *Comparison of ICRs*: The blue regions were computed on a mesh reconstructed from a sensor-acquired point cloud after filtering according to Section III-B; Regions generated on the corresponding CAD-based mesh are depicted in green. The parts of the region on the reconstructed mesh which do not overlap with the ground-truth region constitutes a false positive area;

- flatness $F(\mathbf{p}_s) = \sqrt{\frac{\lambda_s^{(2)}}{\lambda_s^{(3)}}}$,

which are easily obtained during the smoothing stage. The intention is to exclude points \mathbf{p}_s which are isolated and/or yielding low flatness of the associated PDFs. Experiments indicated that good decision thresholds are the respective mean values of the two selection criteria. Thus, a point \mathbf{p}_s is excluded if the cardinality of the associated window \mathcal{W}_s is smaller than the average cardinality over all points and/or the flatness $F(\mathbf{p}_s)$ is smaller than the average flatness over all points.

The proposed strategy does not affect the ICR algorithm itself, but it might prohibit the inclusion of some eligible points in regions \mathcal{C}_n . However, it also decreases the chance of false positives by restraining point inclusion in surface areas with high variance along the vertex normal directions. It limits the ICR growth in areas where Algorithm 1 can give uncertain results, such as curves and edges. Furthermore, isolated flying points are filtered out efficiently.

3) *Filtering complexity*: Algorithm 1 requires iteration of the sliding window \mathcal{W}_s over all S points \mathbf{p}_s in the point cloud and inclusion of neighboring points. Assuming a k -d tree representation of the point cloud, this requires time of order $\mathcal{O}(S \log(S))$. Computing the Gaussian PDF in step 6 of Algorithm 1 is linear in the number of points contained in \mathcal{W}_s . All other steps, including the point preselection, are time constant. Thus, the overall complexity can be stated as $\mathcal{O}(S(\log(S) + \mathbb{E}[|\mathcal{W}_s|]))$. Here, $\mathbb{E}[|\mathcal{W}_s|]$ denotes the expectation value of the cardinality of \mathcal{W}_s , which depends on the density of the point cloud and the chosen window radius r .

IV. EVALUATION METHODOLOGY

For convenience, henceforth meshes reconstructed from unfiltered point clouds are referred to as *raw meshes*, meshes reconstructed from filtered point clouds are denoted as *filtered meshes* and meshes stemming from CAD-models are called *CAD meshes*. Also, contact regions computed on these meshes are augmented with according superscripts, *i. e.*, \mathcal{C}_n^R

Algorithm 2 Intersection Estimation for one region \mathcal{C}_n^S

- 1: **Input**: Independent regions \mathcal{C}_n^C and \mathcal{C}_n^S
 - 2: **Output**: Intersection region \mathcal{I}_n
 - 3: initialize $\mathcal{I}_n = \emptyset$
 - 4: **for** all $\mathbf{p}_s^S \in \mathcal{C}_n^S$ **do**
 - 5: $\mathbf{p}^C \leftarrow P_{S2C}(\mathbf{p}_s^S)$ map \mathbf{p}_s^S onto the CAD-mesh
 - 6: define line \mathcal{L} parallel to normal \mathbf{n}^C and passing through \mathbf{p}_s^S
 - 7: find the triangles $\{\mathcal{T}_I^C\} \in \{\mathcal{T}_n^C\}$ intersected by \mathcal{L}
 - 8: **if** $\exists \{\mathcal{T}_I^C\}$ **then**
 - 9: Add $\{\mathcal{T}_I^C\}$ to the intersection: $\mathcal{I}_n \leftarrow \mathcal{I}_n \cup \{\mathcal{T}_I^C\}$
 - 10: **end if**
 - 11: **end for**
-

stands for a region computed on a raw mesh, \mathcal{C}_n^F denotes an underlying filtered mesh and \mathcal{C}_n^C indicates a region on a CAD mesh.

To make any comparison possible, we have to define a mapping of points from one mesh type onto the other. The mapping of a vertex \mathbf{p}_s^C from a CAD-mesh onto a sensor-acquisition based mesh (indicated by the superscript S - either filtered or raw mesh) is defined as:

$$P_{C2S}(\mathbf{p}_s^C) = \underset{\mathbf{p}^S}{\operatorname{argmin}} (\|\mathbf{n}_s^C \times (\mathbf{p}_s^C - \mathbf{p}^S)\|_2), \quad (2)$$

and identifies the point \mathbf{p}^S , belonging to the sensor-acquisition based mesh, which comprises the smallest normal distance to the line along the vertex normal \mathbf{n}_s^C associated with \mathbf{p}_s^C . Similarly, an inverse mapping of a point on the sensor-acquisition based mesh onto the CAD mesh can be formulated:

$$P_{S2C}(\mathbf{p}_s^S) = \underset{\mathbf{p}^C}{\operatorname{argmin}} (\|\mathbf{n}^C \times (\mathbf{p}_s^S - \mathbf{p}^C)\|_2). \quad (3)$$

Note that both projections are formulated with respect to vertex normals of the CAD mesh. This is due to the fact that the directions of vertex normals obtained from reconstructed range-data frequently deviate substantially from ground-truth and the aim is to decouple the evaluation procedure from uncertainties introduced by sensing errors.

To evaluate the ICR algorithm, two sets of experiments were performed. First, the contact region areas $A(\mathcal{C}_n^S)$ obtained on filtered and raw meshes were compared to the corresponding ground-truth area computed on a CAD mesh. We also investigated to what extent the respective regions overlap, in order to detect false positives. This was done utilizing Algorithm 2, which maps vertices from a sensor-acquisition based mesh onto the corresponding CAD mesh $\mathbf{p}^C \leftarrow P_{S2C}(\mathbf{p}_s^S)$ according to (3). Subsequently, a line \mathcal{L} passing through \mathbf{p}_s^S and parallel to \mathbf{n}^C is formulated. All triangles $\{\mathcal{T}_n^C\}$ which are intersected by this line qualify for inclusion in the intersection \mathcal{I}_n . Figure 6 shows an intersection of independent regions grown on a filtered mesh and a CAD mesh.

The aim of the second set of experiments is to assess the effectiveness of the proposed filtering in a real-world scenario considering the encountered problem with false

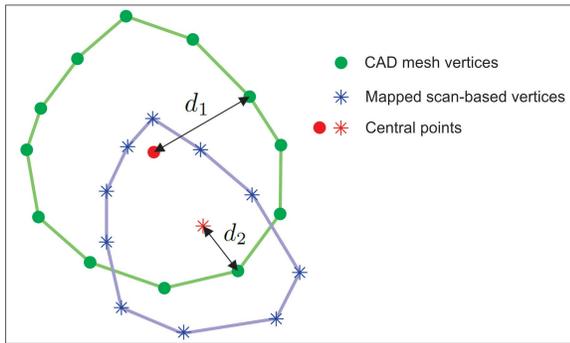


Fig. 7. *Quality criterion Q_n* : Vertices on the border of \mathcal{C}_n^C are depicted in green, the vertices of the corresponding region computed on a sensor-acquisition based mesh are mapped onto the CAD mesh according to (3); The central points (those which minimize the maximum distance to the other vertices in the respective region) are marked red; d_1 and d_2 denote the shortest distance from the respective central points to the border of \mathcal{C}_n^C (note that d_2 is negative in case its associated central point is outside $A(\mathcal{C}_n^C)$); Quality $Q_n = d_2/d_1$;

positives. This test is based on the idea that if we move the initial prototype grasp contacts towards the center of their associated independent regions \mathcal{C}_n (generated on a mesh stemming from range-data), we will achieve higher robustness with respect to finger positioning and sensor-induced modeling errors. To this end, we suggest a simple quality criterion $Q_n = d_2/d_1$ as illustrated in Fig. 7. If $Q_n = 1$, the central point of the considered region coincides with ground-truth and placement of a grasp contact in this point still guarantees the task requirement despite a possible false positive area. Decreasing values of Q_n indicate increased sensitivity to positioning uncertainty. Note that this measure also incorporates the distance to a possible false positive area and thus, large values of Q_n correspond to high robustness.

The experiments were carried out as follows: for all test objects the CAD mesh, raw mesh and filtered mesh were manually registered in a common coordinate frame. In order to compare ICRs generated on the same object for different mesh types, the following procedure has been applied:

- A N -finger force closure grasp \mathcal{G}^C was generated on the CAD mesh. Grasp contacts were modeled as frictional point contacts with a friction coefficient of $\mu = 0.8$, the corresponding friction cones were approximated with octagonal pyramids.
- The corresponding grasps \mathcal{G}^R and \mathcal{G}^F on the reconstructed meshes were found according to (2).
- It was verified whether \mathcal{G}^F passed the preselection routine defined in Section III-B.3.
- If both grasps \mathcal{G}^R and \mathcal{G}^F fulfilled the given task requirement, \mathcal{C}_n^C , \mathcal{C}_n^R and \mathcal{C}_n^F were computed. The task wrench space (see Section II-B) associated with each grasp was formulated as the largest origin-centered insphere of the GWS associated with \mathcal{G}^C , multiplied with a factor $\alpha = 0.8$.
- The respective intersections of \mathcal{C}_n^R and \mathcal{C}_n^F with \mathcal{C}_n^C were generated using Algorithm 2. Furthermore, the quality measures Q_n for \mathcal{C}_n^R and \mathcal{C}_n^F were computed.

On the CAD meshes of every object in Fig. 2-(a) two test

sets, containing 200 randomly generated four- and five-finger force closure grasps respectively, were produced in order to ensure an unbiased and statistically significant evaluation.

V. EXPERIMENTAL RESULTS

Already during the generation of the test grasp sets we found that the proposed smoothing/preselection method increases the number of grasps which preserve the task requirement after mapping them onto the range-data based meshes. More specifically, 98.3% of the grasps on the filtered mesh were eligible, while only 88% of them fulfilled the task after projection onto the raw mesh.

The first set of experiments compares the ICR areas obtained on sensor-acquisition based meshes with the corresponding areas computed on respective ground-truth CAD meshes. Table I presents the intersection areas as percentages of the ICR areas on the CAD meshes. The same measure is computed also for the false positive areas. All values are medians for all contact regions accumulated for all objects. We chose to report the results in median values rather than mean values, in order to not overvalue a few outlier cases with very large false positive areas. Such cases arise especially when a region on the CAD mesh is very small – e.g. a single point. The results show that the filtering method significantly increases the size of the ICRs. The intersection area on a filtered mesh is, on average, about five times larger than the one on a raw mesh. Essentially, generating regions on raw meshes is inadequate, since their average size is only 12% of the corresponding ground-truth. However, an adverse effect of the proposed filtering is that it can increase the false positive area which, in some extreme cases (Wooden Toy 1 object), reaches a median value of 21%.

In the second experiment we computed quality measures Q_n , as introduced in Section IV, for each independent region generated on range-data based meshes for all test objects. This was done in order to assess the usefulness of these ICRs considering a manipulator which tries to position its fingers in the center of the respective regions. Figure 8 shows histograms of the cumulative quality measure for the 4- and 5-finger grasp test sets. The number of regions \mathcal{C}_n (expressed as a percentage of all regions evaluated in the experiment) with a high quality Q_n is larger when mesh filtering is applied. It is evident that in this context the regions generated on filtered meshes are more robust than their counterparts on raw meshes, despite possible false positive areas.

VI. DISCUSSION

This paper presents a method for generation of stable grasp families on objects reconstructed from real-world 3D-sensing data by employing the concept of independent contact regions. It is shown that the applied ICR algorithm is rather sensitive to inconsistencies in the reconstructed triangular mesh, which seriously limits its applicability in practice. Therefore, we have proposed a filtering approach which significantly increases the size of the generated ICRs and obtains a higher consistency with the ground truth.

TABLE I
INTERSECTION (INT) AND FALSE POSITIVE (FP) AREAS AS A
PERCENTAGE OF THE ICR AREA ON THE GROUND-TRUTH CAD MESH

	Smooth mesh		Raw mesh	
	Int[%]	FP [%]	Int[%]	FP[%]
SR4000				
Cup	56	2.0	14	0.0
Wooden Toy 1	54	20	14	0.0
Wooden Toy 2	49	1.0	4.0	0.0
Kinect				
Cup	62	10	17	0.0
Wooden Toy 1	70	21	17	0.0
Wooden Toy 2	50	1.0	12	0.0
Bottle	59	6.0	14	0.0
Paper Cup	58	11	18	0.0
Coffee pot	73	9.0	2.0	0.0
Average	59	9.0	12	0.0

A side-effect of the proposed approach is the possible existence of false positive areas, which might lead to wrong conclusions about the grasp stability. As indicated by the experimental results, one possible solution to this problem is to shift the original grasping points towards the centers of their respective contact regions. Also, Roa and Suárez [23] recently investigated the influence of uncertainties in the ICR computation analytically, and suggest to choose a conservative friction model in order to obtain valid regions.

Another solution suggested by the experimental results is to select prototype grasps of relatively high quality. This can be done utilizing grasps from a grasp taxonomy or grasps demonstrated by a human teacher [19].

ACKNOWLEDGMENTS

This research has been partially supported by the HANDLE project, funded by the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement ICT 231640.

REFERENCES

- [1] A. Miller, S. Knoop, H. Christensen, and P. Allen, "Automatic grasp planning using shape primitives," *Proc. of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1824–1829, 2003.
- [2] A. Morales, C. E., A. Fagg, and A. del Pobil, "Using experience for assessing grasp reliability," *International Journal of Humanoid Robotics*, vol. 1, no. 4, pp. 671–691, 2004.
- [3] A. Saxena, J. Driemeyer, J. Kearns, C. Osundu, and A. Ng, "Learning to grasp novel objects using vision," in *Experimental Robotics*. Springer, 2008, pp. 33–42.
- [4] G. Bone, A. Lambert, and M. Edwards, "Automated modelling and robotic grasping of unknown three-dimensional objects," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2008, pp. 292–298.
- [5] K. Hübner and D. Kragic, "Selection of robot pre-grasps using box-based shape approximation," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2008, pp. 1765–1770.
- [6] V.-D. Nguyen, "Constructing force-closure grasps," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, vol. 3, 1986, pp. 1368–1373.
- [7] J. Ponce and B. Faverjon, "On computing three-finger force-closure grasps of polygonal objects," *IEEE Transactions on Robotics and Automation*, vol. 11, no. 6, pp. 868–881, dec 1995.
- [8] J. Ponce, S. Sullivan, A. Sudsang, J.-D. Boissonnat, and J.-P. Merlet, "On computing four-finger equilibrium and force-closure grasps of polyhedral objects," *International Journal of Robotics Research*, vol. 16, no. 1, pp. 11–35, 1997.

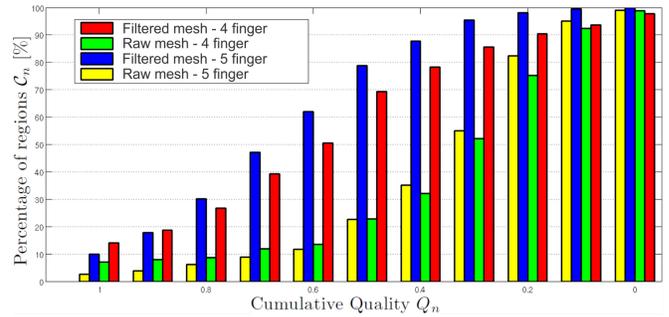


Fig. 8. *Cumulative quality measure Q_n* : Compared are the results obtained by generating ICR on raw and filtered meshes using the test grasp sets for 4- and 5-finger grasps; The histograms show the quality Q_n (see Section IV) accumulated over all the test objects in Fig. 2-(a) and all regions C_n^R and C_n^F respectively;

- [9] N. S. Pollard, "Closure and quality equivalence for efficient synthesis of grasps from examples," *International Journal of Robotics Research*, vol. 23, no. 6, pp. 595–614, 2004.
- [10] M. A. Roa and R. Suárez, "Computation of independent contact regions for grasping 3-d objects," *IEEE Transactions on Robotics*, vol. 25, pp. 839–850, 2009.
- [11] R. Krug, D. Dimitrov, K. Charusta, and B. Iliev, "On the efficient computation of independent contact regions for force closure grasps," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 586–591.
- [12] M. Rahayem and J. Kjellander, "Quadric segmentation and fitting of data captured by a laser profile scanner mounted on an industrial robot," *The International Journal of Advanced Manufacturing Technology*, vol. 52, pp. 1–15, 2011.
- [13] R. Rusu, I. Sucan, B. Gerkey, S. Chitta, M. Beetz, and L. Kavraki, "Real-time perception-guided motion planning for a personal robot," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009, pp. 4245–4252.
- [14] Y. Cui, S. Schuon, D. Chan, S. Thrun, and C. Theobalt, "3D shape scanning with a time-of-flight camera," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 1173–1180.
- [15] S. Fuchs and S. May, "Calibration and registration for precise surface reconstruction with time of flight cameras," *Int. J. Intell. Syst. Technol. Appl.*, vol. 5, pp. 274–284, 2008.
- [16] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.
- [17] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, G. Taubin, et al., "The ball-pivoting algorithm for surface reconstruction," *IEEE Transactions on Visualization and Computer Graphics*, vol. 5, no. 4, pp. 349–359, 1999.
- [18] D. G. Kirkpatrick, B. Mishra, and C. K. Yap, "Quantitative steinitz's theorems with applications to multifingered grasping," in *STOC '90: Proceedings of the twenty-second annual ACM symposium on Theory of computing*, 1990, pp. 341–351.
- [19] S. C. J. Aleotti, "Interactive teaching of task-oriented robot grasps," *Robotics and Autonomous Systems*, vol. 58, no. 5, pp. 539–550, 2010.
- [20] C. Borst, M. Fischer, and G. Hirzinger, "Efficient and precise grasp planning for real world objects," in *Multi-point Interaction with Real and Virtual Objects*, ser. Springer Tracts in Advanced Robotics. Springer Berlin / Heidelberg, 2005, vol. 18, pp. 91–111.
- [21] M. Magnusson, T. Duckett, and A. J. Lilienthal, "3D scan registration for autonomous mining vehicles," *Journal of Field Robotics*, vol. 24, no. 10, pp. 803–827, 2007.
- [22] H. Andreasson, M. Magnusson, and A. J. Lilienthal, "Has Something Changed Here? Autonomous Difference Detection for Security Patrol Robots," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2007, pp. 3429–3435.
- [23] M. A. Roa and R. Suárez, "Influence of contact types and uncertainties in the computation of independent contact regions," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2011, pp. 3317–3323.