A Grasping Metric based on Hand-Object Collision*

Matthew Sundberg, Ryan Sherman, Ammar Kothari[1],
Ravi Balasubramanian[1], Ross L. Hatton[1], and Cindy Grimm[1]

***Abstract*— To date the design of grasping metrics has largely focused on finding (and calculating) specific features that are (potentially) relevant to grouping or characterizing grasps, and particularly on metrics that might predict success (or failure) of a grasp. These metrics leverage human knowledge of physical interaction and are typically relatively quick to compute. One drawback to them, however, is that they are heterogeneous (eg, some combination of number of contact points, force vectors, positional or joint data), often specific to the robotic hand employed (eg, joint angles) and rarely take the full object shape into account (often reducing the shape geometry via PCA to a simple 3-vector). From a machine-learning perspective this makes it challenging to combine and learn over mixed data sets. A more subtle challenge is that the metrics (particularly contact points) are unstable, in that very small movements of the geometry can result in big changes in the number and location of contacts.**

In this paper we explore an alternative metric which is hand and object agnostic, and very stable with respect to small changes in the geometry of the hand or object. Although computationally more expensive than existing, specialized approaches (and also higher dimensional), we propose that it may be more suited to machine learning analysis. At heart, this metric simply captures the ways in which the object is free to "twist" or move out of the hand.

I. Introduction

Robotic grasping is a difficult task, both because of the complexity of the problem space (a typical robotic hand and arm can have over 10 degrees of freedom) and because of the difficulty of representing the possible physical interactions between the hand and the object, which also varies with the object's geometry. Existing approaches to quantifying these interactions have focused — with good cause — on manually-designed metrics that capture specific physical quantities that are likely to be useful such as contact points and forces, joint angles and torques [1] and on "summary" measures such as the PCA of the object's volume, the wrench metric, and the enclosing volume [1] that reduce complicated physical interaction or shape properties down to a few numbers. These metrics have the advantage of reducing this complicated space down to a (relatively) small set of metrics that can be calculated fairly quickly, and have shown some success in predicting grasp success or failure [2].

There are challenges, however, with using these metrics to learn across multiple hands and objects. First, the data is heterogeneous — joint angles, forces, contact points, etc — which makes balancing their contribution (what weights to apply) more difficult. Second, some data (such as joint angles) can't be transferred from one hand to another. Third, other data (such as number of, and location of, contact points) are very unstable. Fourth, object geometry is only poorly captured with existing approaches.

To address these issues, we propose a grasp metric that trades off specificity and conciseness for uniformity and stability. Essentially, this metric records what percentage of the object intersects the hand as the object is moved out of the hand. To make this meaningful (and computationally doable) we define a uniform sampling strategy for transforming the object that takes into account both rotation and translation. This creates a vector of numbers that are homogeneous (intersection volumes), the same for all hands and objects, changes smoothly as the hand morphology/object shape and object pose in the hand change, and captures more detailed aspects of the object geometry.

The limitations of this metric are that it does not directly capture forces or torques (as the wrench metric does) and it is computationally more expensive. Preliminary results show, however, that it is stable under small perturbations and produces a metric that reflects, in part, human notions of grasp similarity.

We first define the mathematics behind the method, then the actual implementation, including normalization for object volume and hand morphology aimed at ensuring the sampling is as similar as possible. We then discuss the specific data set we used to test the algorithm with.

II. Related work

A. Shape analysis

We were inspired, in part, by shape descriptors used to identify similar shapes and parts of shapes (see review in [3], and specifically [4]). Unlike grasp metrics — which have primarily been designed for direct testing of grasp effectiveness — shape descriptors are used to identify similar *shapes*. Our observation is that, given a sufficiently large data base of successful and unsuccessful grasps, finding a good grasp can be reduced to finding a grasp that is "similar" to one that has been shown to be effective.

B. Grasp metrics

Caging, computation of force closure, and related techniques such as contact regions (see eg [5]) are closely related to the metric presented here. One challenge with these metrics is that they are based on detecting contact points — shift the object's pose slightly (or move the fingers) and the set of contacts shifts. This makes the metric somewhat unstable to small perturbations, and if a contact disappears

*Supported in part by NSF Grant CNS 1359480.

[1]Faculty of Mechanical, Industrial, and Manufacturing Engineering, Oregon State University first.last@oregonstate.edu

then the force at that contact point also disappears. Our metric trades direct, physics-based calculations that provide specific information about the grasp for regularized sampling and contact calculation that is more stable.

Our metric is also closely related to Pinto & Gupta's image plus hand-movement metric [6]. Both metrics use movement of the hand as an input; our metric samples movement of the hand relative to the object, theirs the translation and rotation of the hand relative to the object bin. They work from image data, which incorporates shape information for the object and the hand; we directly use the shape of the object and the hand. While we have not attempted to use our metric in grasp optimization/evaluation, their deep neural network approach should be directly applicable to our metric.

III. Methods

Our metric calculates the amount of intersection of the hand and the object as the object is moved out of the hand in a uniformly sampled set of directions. We record the intersection volume along each transformation path, until the object has cleared the hand. The theoretical contribution of his paper is how to create a uniformly sampled set of transformation paths (Section III-A). On the more practical implementation side we also need to define how to calculate the intersections (Section III-B), how to consistently orient the hand and object so that the transformation samples are aligned and how to normalize for object shape and volume (Section III-C). The latter is primarily to account for objects where the grasp only encloses a small part of the shape, such as the spray handle of a bottle. Note that, for the purposes of comparing two grasps we could just try all possible starting configurations and take the minimum distance between the two metrics over all configurations [7] but that would introduce additional computational cost.

A. Transformations

We have three goals in defining the transformation sampling pattern. The first is to capture what happens as the object moves out of the grasp, the second is to sample (as uniformly as possible) the space of all possible rigid body transformations of the object with respect to the hand, and the third is to use a *regular* sampling pattern that is amenable to hierarchical representation or sub-sampling. To satisfy the first goal, we need to define a set of transformation *sequences* that each move the object (simultaneous rotation and translation) out of the hand. The *union* of all of these paths should cover the space of transformations. An example of these paths is shown in Figure 1. The second and third goals are satisfied by defining a regular sampling of the combined translation and rotation space.

The correct way to generate these paths is to use a discretized version of exponentiation. Essentially, the path is defined by a translation direction T (in object space) along with a simultaneous rotation R (again in object space). To define the path, we integrate, applying TR to the object at each time step. This produces the paths shown in Figure 1. By uniformly sampling the set of translation directions and rotations, we can generate a uniformly distributed set of transformation paths. The union of all of the poses along all of the paths provides a (relatively) uniform sampling of poses that is denser around the initial pose, but is still regular. This is in contrast to simply generating random poses [8], which would satisfy the uniform sampling goal but would not provide any structured relationship within the sampling pattern.

More formally, we split up the sampling into a body-frame translation velocity ($\dot{T} = (\dot{T_x}, \dot{T_y}, \dot{T_z}), ||\dot{T}|| = 1$), a body-frame rotation direction $\omega = (\dot{\psi}, \dot{\phi}, \dot{\theta})$, and a rotation speed $||\omega||$. Each set of these parameters defines a helical path on $SE(3)$, where for velocities arranged as

$$m(T,\omega) = \begin{bmatrix} 0 & -\dot{\theta} & \dot{\phi} & \dot{T_x} \\ \dot{\theta} & 0 & -\dot{\psi} & \dot{T_y} \\ -\dot{\phi} & \dot{\psi} & 0 & \dot{T_z} \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{1}$$

the object pose at time t is

$$M(T,\omega,t) = \exp(mt) \tag{2}$$

$$= \begin{bmatrix} & R^{3\times3} & & x \\ & & & y \\ & & & z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

where the matrix M is a combined rotation-translation matrix that results from integrating the translation direction $\dot{T}$ and rotation direction and amount ω in object space.

We sample these trajectories at a set of times $0 \leq t_i \leq 1$, where for simplicity's sake we assume that the length of the paths we want to generate is 1 (we can always scale the combined hand and object geometry to ensure that a path length of 1 takes the object out of the hand's enclosing volume, see III-C). By construction, sampling evenly along t produces evenly spaced samples along the path. In this paper we used 10 steps for each path, but discarded the origin since it is the same for all paths, for a total of 9 samples.

To sample the rotation and translation directions we used a spiral phyllotaxis pattern mapped to the sphere [9], which produces a relatively uniform sampling of points on the sphere for any given number of points. Moreover, changing the number of points changes the sample locations in a well-defined way. This pattern provides the translation direction (3 degrees of freedom) and the rotation axis (3 degrees of freedom) with a fourth degree of freedom the *amount* of rotation (a quarter or half turn in the positive and negative directions).

To generate our samples we pick the number of translation directions N_t, the number of rotation axes N_a and the number of rotation amounts N_r. We combine these in all possible ways, plus a no-rotation option and a no-translation option, resulting in $9(N_t * N_a * N_r + N_t + N_r * N_a)$ samples.

We experimented with sampling the translations more than the rotations ($N_t = 28$, $N_r = 7$, $N_a = 2$), the rotations more than the translations ($N_t = 10$, $N_r = 7$, $N_a = 4$), and a balanced amount of each ($N_t = 18$, $N_r = 6$, $N_a = 4$). The

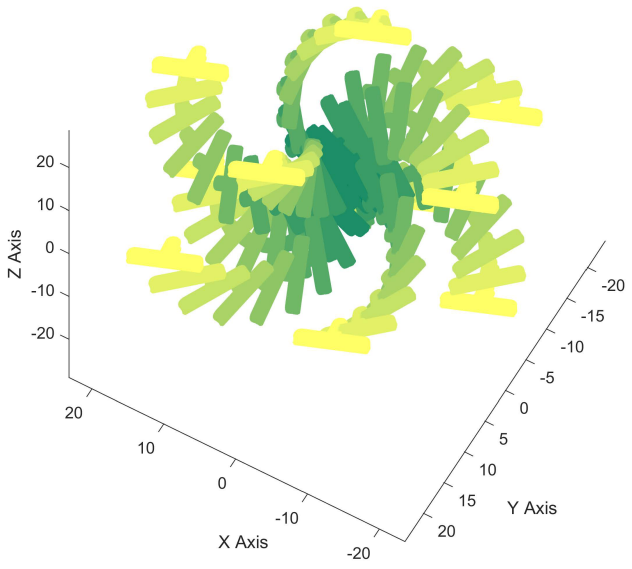

Fig. 1. A small number of example paths, colored by time, for a T-shaped object.

results for the noise test were qualitatively similar for all three cases (see Figure 6).

B. Intersections

To calculate the intersections we need a reasonably fast method that is suitable for a range of geometry. Essentially, we represent the hand geometry as an inside-outside function and the object as a set of uniformly-sized interior voxels/cubes plus a set of surface half-sized cubes/hemispheres. The volume calculation is then simply a matter of transforming the object cubes and summing up the number of cubes who's centers are inside.

This formulation will tend to over-estimate the intersection volume. We could perform a more exact volume calculation by, for example, finding the zero-level surface within the boundary object cubes, however, the extra cost is not warranted in our method. More specifically, we are looking for the *pattern* of intersection volume change, not the exact volume intersection value. Therefore, as long as the approach (roughly) over-estimates the value in the same way for all objects, the inaccuracy will not have a noticeable effect.

We use samples on the object's surface to account for thin object geometry (eg, stems of wine glasses). We are less concerned with thin hand geometry, in part because robotic hands tend to not have really thin elements, and in part because we can use relatively high sampling on the hand geometry. The resolution of the hand grid has a one-time cost (scanning the geometry), but the inside/outside calculation of the object cubes depends only on the number of object cubes, not on the resolution of the hand grid.

More specifically, we use VOXELISE in MATLAB [10] to convert the hand geometry into an inside/outside function, and to compute the interior cubes for the object. We use MeshLab's Poisson disk re-sampling function [11] to generate the points on the surface. We chose the number of samples based on the surface area of the object and the specified grid resolution for the object, ensuring that the Poisson disk radius is (approximately) 1/16 of the object's grid size.

For the objects in this paper we used a hand grid resolution of 100 and an object grid resolution of 50.

Let V_o be the volume of an object cube. The intersection volume calculation is simply the number of object interior cubes inside of the hand times V_o, plus the number of boundary sample points inside of the hand times $(V_o/2 \times 1/16)$.

We discuss volume normalization in Section III-C.

C. Normalization

Although the *descriptor* is the same for all hand/object pairs, there are issues of normalization for different hands:

- Where the paths start with respect to the hand — i.e., what is considered the origin?
- The orientation of the paths with respect to the hand — i.e., what direction do the objects travel in?
- The overall enclosing volume of the hand — i.e., how far along the path is considered the "same"?
- Accounting for the volume occupied by the hand geometry itself.

We define the coordinate system using the hand's geometry, essentially placing the origin in the center of the hand's enclosing volume, using the orientation of the palm to define the orientation of the coordinate system ("out" of the palm and "up" from the enclosing finger(s). More specifically, close the fingers loosely. Let d be the distance from the palm to the outside of the fingers, as measured along the normal vector from the palm. We place the origin half-way along this vector, and scale the system so that $d/2$ is 1.

Note that the center of the rotation system is NOT defined by the object's center.

We also normalize for the hand volume, representing the intersection as the percentage of hand inside the object over the volume of the hand. It does not make sense to normalize for the object's volume since the object's size relative to the hand would swamp the calculations.

IV. Evaluations and Results

In this section we perform comparisons on an existing grasp data set in order to evaluate the metric in practice. The test examines how the metric behaves when noise is added to the object's position and orientation within the hand.

A. Noise test

We perform two noise tests that looks at the behavior of the metric as the object's position and orientation are varied slightly in the hand, and as the joint angles of the fingers are varied slightly.

Figure 2 shows the objects and grasps used for the noise evaluations. These 6 grasps were chosen somewhat at random from the 150 available grasps with the goals of 1) choosing grasps that were "different" in that they used a different number of fingers or different finger configurations; 2) the grasps were located at different positions relative to

the object's center; 3) the geometry was different; 4) the geometry was a different size relative to the hand.

Comparison function: We tried both the L^2 norm of the values and the L^2 norm of the sequence *differences* (how much the intersection values changed along the trajectory). For the noise tests the L^2 norm was more discriminatory so it was chosen for the majority of figures in this paper. We show an example of the L^2 norm of the differences in the right hand side of Figure 6.

Object pose: We added noise to the object's pose by moving its center by ε of the overall maximum length of the object and rotating it around its center by an arbitrary axis by π/δ. We moved each object 6 times for each noise level. We used two noise levels, $\varepsilon = 0.0125, \delta = 64$ and $\varepsilon = 0.025, \delta = 32$. An example of all of the generated positions for one object is shown in the top of Figure 4. This produces a total of $2 \times 6 \times 6$ distinct data points. We compare each data point to all of the others, and classify them as between-object ($(6 \times (6-1))/2 \times (6 \times 6) = 540$ comparisons per noise level) or within-object ($(6 \times (6 \times (6-1))/2 = 90$ comparisons per noise level). We compare the within-object noise distribution versus the between-object noise distribution in the bottom of Figure 4. For both noise levels the within-object L^2 difference distribution is distinct from the between-object distribution.

Joint angle: For the joint angle test we varied the joint angles of the fingers for the spray bottle grasp. We randomly added five levels of noise to the joint angles (1-5% of the joint range). We generated 20 random perturbations at each level. Figure 3 compares the distribution of errors for each percentage level.

In Figure 5 we look at the variation in L^2 norms on a per-object basis. Within-object: We see that the Bottle body and Box grasps have the highest L^2 differences and spreads, which is because these objects are both large and the center of noise rotation is outside of the hand. The other four are similar. As we increase the noise the mean L^2 difference increases as expected. Between-object: Increasing the noise (top left) increases the standard deviation and mean L^2 values for each object pair slightly; see Figure 2 to see which pairs are most similar. For comparison's sake we plot a light gray line at the maximum L^2 value for the within-object, small noise group, and the minimum L^2 value for the between-object, small noise group.

V. Discussion

The tests here are fairly simplistic, but do show the metric is discriminatory and relatively insensitive to small perturbations in position and changes in sampling rates. More analysis is needed to determine how well the metric behaves when comparing different hand geometry and what the best trade-off is between sampling rate and computational complexity. The current computation time is around 10 seconds (approximately 200 trajectory samples) to a minute (approximately 600 trajectory samples) with un-optimized MATLAB code.

VI. Conclusion

In conclusion we have shown the feasibility of a geometry-only metric that is largely hand and object shape agnostic, and which is amenable to machine-learning approaches.

VII. Acknowledgment

Funded in part by NSF grants CNS 1730126 and CNS 1659746. We would also like to thank the Saturday Academy's ASE program.

Appendix

References

[1] M. A. Roa and R. Suárez, "Grasp quality measures: review and performance," *Autonomous robots*, vol. 38, no. 1, pp. 65–88, 2015.

[2] A. K. Goins, R. Carpenter, W.-K. Wong, and R. Balasubramanian, "Evaluating the efficacy of grasp metrics for utilization in a gaussian process-based grasp predictor," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 3353–3360.

[3] P. Heider, A. Pierre-Pierre, R. Li, and C. Grimm, "Local shape descriptors, a survey and evaluation," in *Proceedings of the 4th Eurographics Conference on 3D Object Retrieval*, ser. 3DOR '11. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2011, pp. 49–56. [Online]. Available: http://dx.doi.org/10.2312/3DOR/3DOR11/049-056

[4] R. Gal, A. Shamir, and D. Cohen-Or, "Pose-oblivious shape signature," *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 2, pp. 261–271, Mar. 2007. [Online]. Available: http://dx.doi.org/10.1109/TVCG.2007.45

[5] A. Bicchi and V. Kumar, "Robotic grasping and contact: a review," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, vol. 1, 2000, pp. 348–353 vol.1.

[6] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 3406–3413.

[7] T. Gatzke, C. Grimm, M. Garland, and S. Zelinka, "Curvature maps for local shape comparison," in *Proceedings of the International Conference on Shape Modeling and Applications 2005*, ser. SMI '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 246–255. [Online]. Available: http://dx.doi.org/10.1109/SMI.2005.13

[8] J. J. Kuffner, "Effective sampling and distance metrics for 3d rigid body path planning," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 4, April 2004, pp. 3993–3998 Vol.4.

[9] C. Carlson, "How i made wine glasses from sunflowers," *Blot*, 2011. [Online]. Available: http://blog.wolfram.com/2011/07/28/how-i-made-wine-glasses-from-sunflowers/

[10] A. H. Aitkenhead, "Matlab voxelization software," 2013. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/27390-mesh-voxelisation

[11] M. Corsini, P. Cignoni, and R. Scopigno, "Efficient and flexible sampling with blue noise properties of triangular meshes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 6, pp. 914–924, Jun. 2012. [Online]. Available: http://dx.doi.org/10.1109/TVCG.2012.34

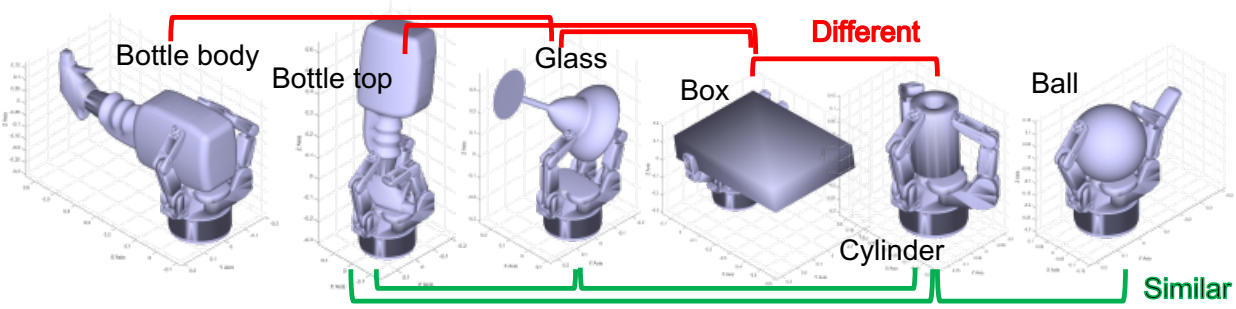

Fig. 2. The 6 grasps selected for the noise tests. The green bars on the bottom denote the four most similar grasps under object movement (as measured by our metric), while the red bars on the top denote the for most dis-similar grasps.

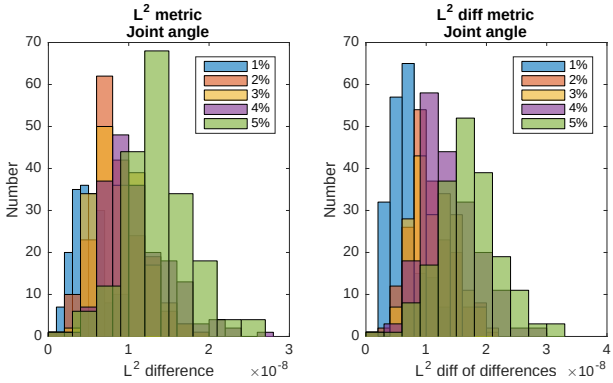

Fig. 3. Distribution of differences as joint noise levels are increased from 1 to 5 percent. Note increasing error as joint angle noise increases, but overall differences are still well below between-grasp differences.

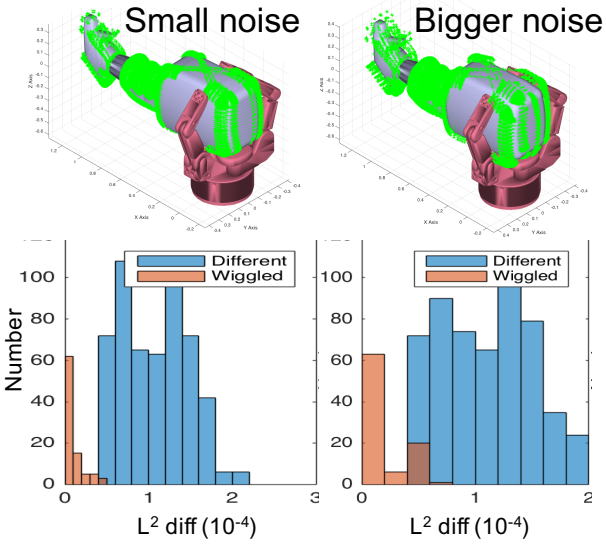

Fig. 4. Top row: Green points are the vertices of the object after noise was added (six copies total). Bottom row: Distribution of L^2 differences for the copies with noise (orange) and the 6 grasps compared to each other (blue).

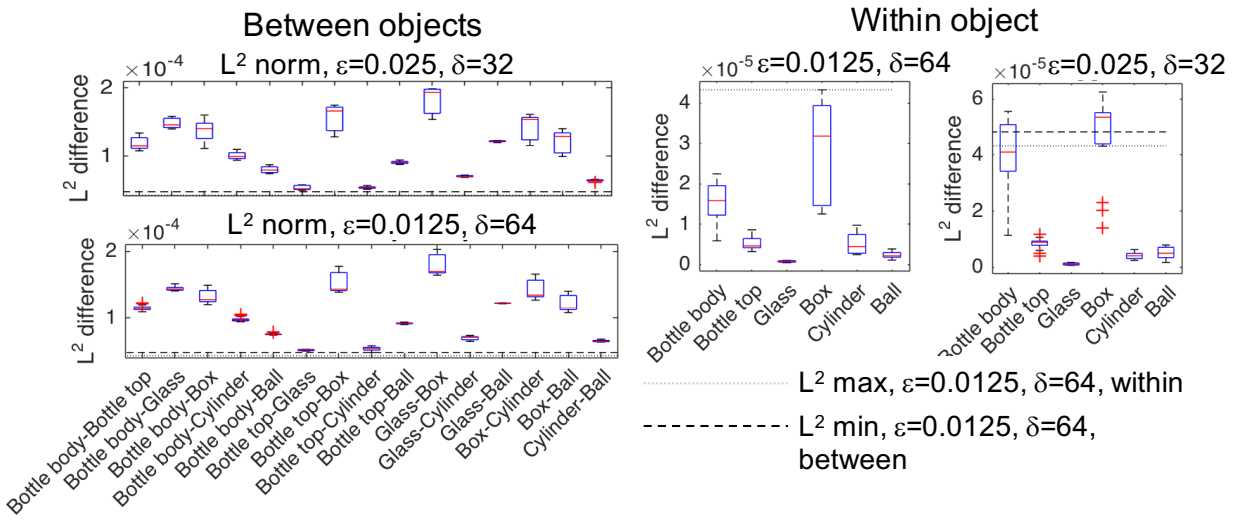

Fig. 5. Object noise. Left: Between-object L^2 differences, top is more noise. Right: Within-object L^2 differences, right is more noise. We plot two lines at the same value on all graphs for comparisons between all four graphs (the plots on the right are largely under the dashed gray line on the left).

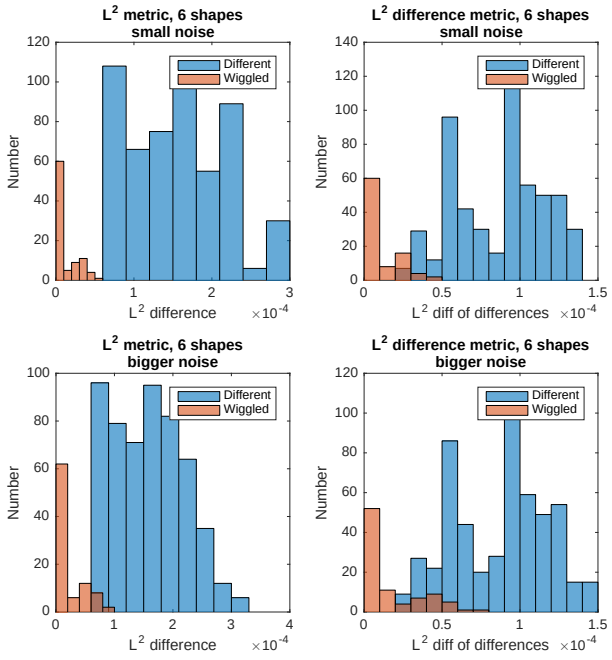

Fig. 6. Increasing the number of rotations sampled versus translations results in error distributions that are qualitatively similar.