Multi-View Picking:
Next-best-view Reaching for Improved Grasping in Clutter

Douglas Morrison[1], Peter Corke[1] and Jürgen Leitner[1]

Abstract— Camera viewpoint selection is an important aspect of visual grasp detection, especially in clutter where many occlusions are present. Where other approaches use a static camera position or fixed data collection routines, our Multi-View Picking (MVP) controller uses an active perception approach to choose informative viewpoints based directly on a distribution of grasp pose estimates in real time, reducing uncertainty in the grasp poses caused by clutter and occlusions. In trials of grasping 20 objects from clutter, our MVP controller achieves 80% grasp success, outperforming a single-viewpoint grasp detector by 12%. We also show that our approach is both more accurate and more efficient than approaches which consider multiple fixed viewpoints. Code is available at `https://github.com/dougsm/mvp_grasp`

I. INTRODUCTION

Grasping and transporting objects is a canonical robotics problem which has seen great advancements in recent years, especially with regards to detection of grasp poses for previously unseen objects using only visual information. As the performance of these visual grasp detection systems has improved, so has the difficulty of standard benchmarking tasks for evaluation, seeing a shift away from grasping relatively simple, isolated objects to grasping geometrically and visually challenging objects in cluttered environments [1].

The use of cluttered environments and complex objects has resulted in more visually challenging scenarios for grasp detection, with the level of clutter impacting grasp detection performance [1]. Recent work has shown that improved visual information from point cloud fusion [2, 3] or viewpoint selection [4] can improve the quality of grasp estimates in clutter. However, these typically are treated as separate, fixed data gathering step and do not directly take into account grasp detections from multiple viewpoints.

At the same time, improvements in grasp detection models and computational hardware have seen the time required to visually detect grasps (especially using deep learning methods) decrease from tens of seconds [5] to less than a second [6, 7] to small fractions of a second [8]. Consequently, the largest contributor to grasp execution time has become the movement of the robotic arm executing the grasp, making grasp detection from multiple viewpoints feasible with very little impact to overall execution time.

To address the added difficulties of grasping in clutter, we propose to use the act of reaching towards a grasp as a method of data collection, making it a meaningful part of the grasping pipeline, rather than just a mechanical necessity.
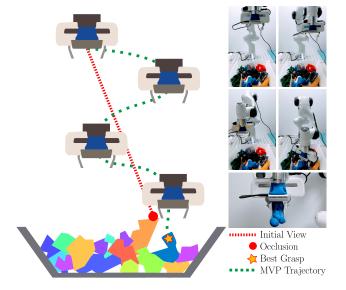
Fig. 1. Our Multi-View Picking (MVP) controller considers multiple informative viewpoints for an eye-in-hand camera while reaching for a grasp in clutter to reduce uncertainty in the grasp pose estimate caused by clutter and occlusions. This allows it to perform high quality grasps that may not be clearly visible from the initial viewpoint.

To achieve this, we develop the Multi-View Picking (MVP) controller, which selects multiple informative viewpoints for an eye-in-hand camera while reaching to a grasp in order to reduce uncertainty in the grasp pose estimate caused by clutter and occlusion, resulting in an overall improvement in grasp success (Fig. 1). Unlike previous works in active perception for grasping which employ object-specific heuristics [4] or a secondary task such as point cloud reconstruction [2, 3, 9], our approach directly uses entropy in the grasp pose estimation to influence control.

We validate our approach through trials of grasping from piles of 20 objects in clutter, and compare our results to baselines which represent common visual grasp detection approaches. Using our MVP controller, we achieve an 80% success rate in grasping from clutter, a 12% increase compared to a single-viewpoint grasp detection approach. Additionally, our method outperforms a baseline which considers multiple viewpoints over a predefined trajectory, achieving a higher grasp success rate with fewer distinct viewpoints and reduced mean time per grasp. This highlights the advantages of using our information-gain based approach which focuses on salient areas, reducing unnecessary data collection. By varying the cost associated with data collection during reaching, we show that it is possible to trade off between success rate and execution time, allowing the system to be optimised for either raw grasp success rate or overall efficiency.

II. RELATED WORK

In order to grasp and transport a wide range of unknown objects, a robotic system cannot rely on using offline information such as object models or object-specific grasp poses. Instead, it must use geometric information to compute stable grasp poses for previously unseen objects. Recently, many approaches combining visual inputs with machine-learning techniques – primarily Convolutional Neural Networks (CNNs) – have been widely and successfully applied to this problem [1, 2, 5, 6, 8, 10, 11], which we refer to as visual grasp detection.

The robustness of many visual grasp detection systems is sensitive to factors such as sensor noise, robot control inaccuracies and visual occlusion, which is especially prevalent in cluttered environments. While the detrimental effects of sensor noise and poor robot precision can be minimised as part of the visual grasp detection algorithm, e.g. through sensor noise injection during training [1, 6, 7, 11], overcoming the issue of occlusion in clutter requires multiple camera viewpoints to be considered. For example, ten Pas *et al.* [2] showed that computing grasp poses using a fused point cloud from many viewpoints along a predefined trajectory resulted in a 9% increase in grasp success rate compared to using a point cloud collected from two static cameras. Rather than rely on a fixed data collection routine, Arruda *et al.* [3] use an active perception approach to choose viewpoints which specifically aid point cloud reconstruction near potential finger contact points in an efficient manner.

Broadly, active perception is defined as the situation where a robot "adopts strategies for decisions of sensor placement or sensor configuration" in order to perform a task [12]. It is a concept which has been applied to a wide variety of robotic tasks, such as mapping [13, 14], object modelling [15], object identification [16, 17] and path planning [18]. Common strategies for active perception focus on planning the expected next best action to efficiently minimise measurement uncertainty or maximise information gain via a metric such as Shannon entropy [19] or KL divergence [20].

Active perception approaches have been applied to robotic grasping in prior work, but rather than directly use grasp quality as a metric rely on a secondary task such as object modelling [21, 22], point cloud reconstruction while searching for graspable geometry [3, 9] or localising previously seen objects [23] to generate next best view commands. This is partly due to the difficulty in defining a probability distribution over predicted grasp poses [4], which we overcome by using a visual grasp detection system which generates a pixelwise distribution of grasp pose estimates [8].

Gualtieri *et al.* [4] apply active perception directly to grasp detection. They compute a distribution of viewpoints for object classes which are likely to improve the quality of detected grasps. However, this approach requires knowledge of the object's class and doesn't easily generalise to cluttered environments or more complex objects. They also show improved grasp estimates using a heuristic approach where the camera is aligned to the best detected grasp.

III. MULTI-VIEW PICKING

The choice of camera viewpoints plays an important role in the quality of visual grasp detection. In this work, we apply active perception techniques to compute the next best viewpoint for a robot with an eye-in-hand camera in real time while reaching for an object. Unlike previous work, our approach does not rely on any object-specific knowledge or heuristics, does not use a fixed data collection routine and uses visual grasp detection observations directly rather than a secondary metric such as point cloud reconstruction.

We develop an information-gain controller, which we call the Multi-View Picking (MVP) controller. The MVP controller combines visual grasp predictions from multiple camera viewpoints along a trajectory to a grasp pose, and seeks to minimise the uncertainty (entropy) associated with the grasp pose prediction by altering the trajectory to include informative viewpoints, overcoming the challenges associated with visual grasp detection in clutter. Our implementation of the MVP controller is described in the following sections, with an overview given in Fig. 3.

A. Viewpoint Trajectory

We consider the case of a robotic arm with an antipodal gripper and an eye-in-hand camera reaching from an arbitrary starting pose to a grasp pose $\mathbf{g}$. At any instance during this motion, the camera's viewpoint $\mathbf{p}$ is the 3D position of the camera, centred above a point in the workspace (x, y) at a height z, which we constrain to be parallel to the xy-plane. We define a viewpoint trajectory for grasping as a set of K discrete viewpoints $\mathcal{T} = \{\mathbf{p}_0, ..., \mathbf{p}_K\}$ from which we make a visual grasp detection observation while reaching for a grasp. The initial viewpoint $\mathbf{p}_0$ is at a fixed height $z_{\max}$, and the trajectory ends when the camera reaches a height $z_{\min}$, at which point the best detected grasp is executed.

B. Visual Grasp Detection

For visual grasp detection, we use the real-time Generative Grasping Convolutional Neural Network (GG-CNN) from [8]. Given a depth image as input, the GG-CNN produces a pixelwise visual grasp detection $\mathbf{x} = (\mathbf{Q}, \mathbf{\Phi}, \mathbf{W})$, representing the grasp quality (the chance of grasp success), angle of rotation around the vertical axis and gripper width at each pixel of the input respectively (Fig. 2). Each pixel of $\mathbf{x}$ represents an antipodal grasp $\mathbf{g} = (\mathbf{c}, \phi, w, q)$, parameterised by the grasp's centre position $\mathbf{c} = (x, y, z)$, rotation $\phi \in [0, \pi]$, gripper width w and quality $q \in [0, 1]$. $\mathbf{c}$ is computed from the measured depth and the camera's intrinsic parameters.

RGB (Not Used)   Depth

0   1 0   π 0.02   0.12
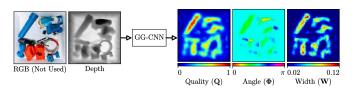Quality (**Q**)   Angle (**Φ**)   Width (**W**)

Fig. 2.   We use the GG-CNN grasp prediction network [8]. Given a depth image, the GG-CNN predicts the quality, rotation around the vertical axis and gripper width for a grasp at the position corresponding to each pixel.
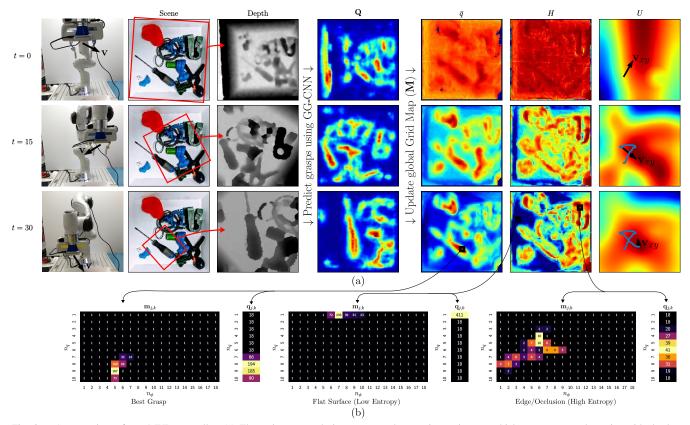
Fig. 3. An overview of our MVP controller. (a) Three time steps during an example grasping trajectory which converges to the point with the best detected grasp pose (the screwdriver handle), considering informative viewpoints corresponding to areas of high expected information gain while reaching. From left to right: The position of the robot (and camera); the collection of objects being grasped; the depth image captured from the current pose $\mathbf{p}_t$; the quality output of the GG-CNN ($\mathbf{Q}$); the average quality at each grid cell ($\bar{q}$); the entropy at every grid cell (H); the utility of a viewpoint (U) above each cell in the workspace, from which the velocity command $\mathbf{v}$ is calculated, and the traversed path superimposed in blue. (b) The histograms $\mathbf{m}_{j,k}$ and $\mathbf{q}_{j,k}$ which represent the counts of observations at, from left to right, grid cells with the best estimated grasp pose, a flat surface with low entropy and, a highly occluded area which results in a high measurement entropy.

Other visual grasp detection systems regress a single grasp pose [24, 25] or perform classification on sampled grasp candidates [5, 6], which do not easily lend themselves to defining a probability distribution over grasp estimates. The GG-CNN is an ideal component of our active perception system because it directly generates a distribution of grasp estimates and also gives real-time computation (approx. 50Hz). An additional advantage of GG-CNN is that it would be trivial to mask the pixelwise grasp estimates with an off-the-shelf semantic segmentation system if object-specific grasping was required.

C. Grid Map Representation

To combine observations at time-steps along the viewpoint trajectory, we represent the workspace of the robot as a 2D grid map, M, of $J \times K$ cells. Each cell corresponds an $u \times u$ physical area. The grid map has the advantage of being computationally efficient compared to other representations, such as Gaussian Processes, which could be used in a similar. Within each cell (j,k), grasp quality observations (q) are counted in a vector $\mathbf{q}_{j,k}$, discretised into N_q intervals, and combined grasp quality and angle observations (q, ϕ) are counted in a 2-dimensional histogram $\mathbf{m}_{j,k}$, discretised into $N_q \times N_\phi$ intervals respectively (Fig. 3b). These vectors

represent the distribution of observations at each point, and form the basis for our information gain approach.

A grasp at cell (j,k) is parameterised by the mean of observations within that cell:

$$\mathbf{g}_{j,k} = (\mathbf{c}_{j,k}, \bar{\phi}_{j,k}, \bar{w}_{j,k}, \bar{q}_{j,k}) \tag{1}$$

where $\mathbf{c}_{j,k}$ is the physical position at the centre of the cell and the mean observations are calculated as per below. (We drop the j,k notation for the sake of readability.)

For a single cell, the mean quality observation $\bar{q}$ is given by:

$$\bar{q} = \frac{1}{\sum \mathbf{q}} \sum_{n_q=1}^{N_q} \frac{n_q}{N_q} \mathbf{q}_{n_q} \tag{2}$$

and the mean angle $\bar{\phi}$ is the vector mean of the angle observations [26] weighted by the corresponding grasp quality observations:

$$\bar{\phi} = \arctan \frac{\sum_{n_\phi=1}^{N_\phi} \sum_{n_q=1}^{N_q} \sin(\frac{n_\phi}{N_\phi}\pi) \frac{n_q}{N_q} \mathbf{m}_{n_\phi,n_q}}{\sum_{n_\phi=1}^{N_\phi} \sum_{n_q=1}^{N_q} \cos(\frac{n_\phi}{N_\phi}\pi) \frac{n_q}{N_q} \mathbf{m}_{n_\phi,n_q}} \tag{3}$$

The mean grasp width $\bar{w}$ for a cell is simply the mean of n observations: $\bar{w} = \frac{1}{n} \sum w$

D. MVP Controller

We formulate our MVP controller using an information gain approach, where viewpoints are selected to reduce the uncertainty in the grasp pose observations. Specifically, we aim to reduce the entropy of grasp quality observations in M which correspond to high quality grasps.

We can calculate the entropy of the grasp quality observations a single grid cell as:

$$H(\mathbf{q}) = -\sum_{n_q=1}^{N_q} \frac{\mathbf{q}_{n_q}}{\sum \mathbf{q}} \log \left(\frac{\mathbf{q}_{n_q}}{\sum \mathbf{q}} \right) \qquad (4)$$

Here, we calculate entropy in the quality observations $\mathbf{q}$ only, rather than the full distribution of grasp quality and angle in $\mathbf{m}$ because entropy in the distribution of angle measurements is not always a good indicator of uncertainty in the measurement. For example a small, spherical object can be graspable from any angle with high quality, so may have a high entropy across $\mathbf{m}$ but low entropy when considering only $\mathbf{q}$.

We find that entropy in the distribution of grasp quality measurements is a good candidate for predicting informative viewpoints. The fifth column of Fig. 3a shows the entropy of measurements in a semi-cluttered scene at three time steps during a grasp. As shown in Fig. 3, areas of high entropy are present around areas of clutter, occlusion and complex geometry where the output of the grasp detector is highly variable and dependent on viewpoint, compared to "uninteresting" the areas with certain measurements (and hence low entropy) regardless of viewpoint such as the flat bottom surface of the workspace.

Calculating the expected information gain of an observation from a viewpoint, in terms of reduction in entropy, is an intractable problem in this case. As such, we use the common simplifying assumption that the total entropy of the observed area will provide a good approximation to computing the expected information gain [27]. That is, viewpoints which observe areas of high entropy are likely to be more informative (i.e. reduce entropy) than observing areas which already have low entropy.

Hence, we approximate the expected information gain from an observation at a viewpoint as the weighted sum of entropy of the grid cells observed from that viewpoint:

$$E[I(M, \mathbf{p})] \approx \sum_{j,k \in O_{\mathbf{p}}} P(j,k) \cdot H(\mathbf{q}_{j,k}) \qquad (5)$$

where $O_{\mathbf{p}}$ is the set of grid cell coordinates observable by the camera from viewpoint $\mathbf{p}$, and $P(j,k)$ weights points by a Gaussian function based on their distance from the geometric centre of O, encouraging the controller to view high-entropy areas front-on rather than peripherally.

To predict the next best viewpoint, we calculate the utility of moving to a viewpoint centred above each cell in M, which represents the desirability of moving to each viewpoint as:

$$U(M, \mathbf{p}) = E[I(M, \mathbf{p})] - \gamma r(\mathbf{p}) \qquad (6)$$

where $r(\mathbf{p})$ is the cost associated with moving to the viewpoint $\mathbf{p}$, tunable by the *exploration cost* parameter γ. To encourage the controller to observe areas nearby the best detected grasp, rather than irrelevant distant points, the cost term represents the horizontal (xy) distance to the grid cell with the highest average grasp quality, centred at $\mathbf{c}$:

$$r(\mathbf{p}) = \|\mathbf{p}_{xy} - \mathbf{c}_{xy}\| \cdot \left(1 - \frac{\mathbf{p}_z - z_{\min}}{z_{\max} - z_{\min}} \right) \qquad (7)$$

The second term scales the cost based on the vertical position of the camera in the trajectory ($\mathbf{p}_z$). At the beginning of the trajectory there is zero cost, encouraging exploration of the workspace, which increases linearly as the end-effector descends, causing the controller to converge to the best grasp.

The MVP controller generates a horizontal velocity command $\mathbf{v}_{xy}$ in the direction of maximum utility, shown in the final column of Fig. 3a. To enable direct comparison between our experiments, we scale the vertical component of velocity $\mathbf{v}_z$ such that the absolute end-effector velocity $|\mathbf{v}|$ is constant.

IV. ROBOTIC EXPERIMENTS

We test and validate our approach through a number of trials of robotic grasping in clutter, with the setup is shown in Fig. 4a. Our experiments are described in the following sections. Our software implementation of the system in the form of ROS nodes, primarily written in python, will be made available online.

A. Equipment

Experiments are performed using a Franka Emika Panda robot, fitted with custom, 3D-printed gripper fingers using the design from [28]. We use an Intel Realsense D435 depth camera, which is mounted to the robot's end-effector.

B. Objects

We use a set of 40 objects, comprising 20 "adversarial" 3D-printed objects from the DexNet 2.0 [6] dataset[1] (Fig. 4b), and 20 household objects (Fig. 4c). The adversarial objects have complex geometry, making them difficult to perceive and grasp. The household objects contains objects that are a wide variety of sizes and shapes and include deformable objects and visually challenging objects which are transparent or black.

C. Method

For each experimental run, 20 objects are chosen at random, 10 from each of the adversarial and household sets, and emptied into a 30×30cm bin in an unstructured jumble. The robot then grasps objects one by one and places them into a second bin until all objects have been removed. A grasp is counted as a success if the object is successfully transported to the second bin. Scales on the second bin are used to record the success or failure.

We perform 9 experiments using the MVP controller, comprising 7 runs each, varying the *exploration cost* γ while

---

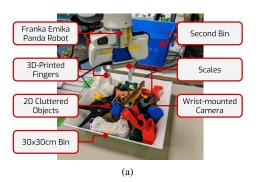[1]3D-printable mesh files are available from https://berkeleyautomation.github.io/dex-net/

Fig. 4. (a) Our experimental setup. (b) The adversarial object set. (c) The household object set.

TABLE I

PARAMETERS USED DURING EXPERIMENTS

Parameter		Value
J, K	Grid Map Size	68
u	Grid Cell Size	5mm
N_q	Quality Bins	10
N_ϕ	Angle Bins	18
$z_{\max}$	Starting Height	0.5m
$z_{\min}$	Final Height	0.2m
	Controller Update Rate	10Hz
$|\mathbf{v}|$	End-effector Velocity (During Reach)	0.1m/s
γ	Exploration Cost	**Varied**

keeping all other parameters listed in Table I constant. Additionally, we perform baseline experiments for comparison as described in the next section.

D. Baselines

We compare our results to three baselines which represent common methods in other robotic grasping work. We complete 5 runs of each baseline using the method above. Where relevant, the parameters in Table I (including the end-effector velocity) are kept constant to allow for the best possible comparison.

Single Viewpoint Most work in visual grasp detection considers only a single viewpoint for grasp detection [1, 5, 6, 10, 11]. In this baseline we always execute the best grasp detected by the GG-CNN from a single viewpoint centred above the workspace.

Fixed Data Collection ten Pas *et al.* [2] increase their grasp success rate by performing point cloud fusion along a fixed trajectory. Because our method uses grasp estimates rather than point clouds, we use our grid map representation to combine GG-CNN predictions along a fixed, spiral trajectory (Fig. 5), considering 25 and 50 uniformly spaced viewpoints in two experiments.

No Exploration Gualtieri *et al.* [4] showed an increase in grasp success by aligning the camera to the axis of the best detected grasp. In this baseline we disable the exploration of our MVP controller, instead always generating a velocity command to align the camera to the best detected grasp at each time step.

V. RESULTS

The results of our experiments are shown in Table II. We assess each experiment based on three metrics:

- *Success Rate:* The overall ratio of successful grasps to grasp attempts across all runs.
- *Mean Time per Pick:* The average time per grasp attempt (in seconds), regardless of success.
- *Mean Picks Per Hour (MPPH):* The overall efficiency of the system representing the average rate of successful picks per hour calculated as 3600 [s/h] ÷ (Mean Time per Pick [s]) × (Success Rate)

It is important to note that we include the MPPH measurement as a way of comparing the different results, so are primarily concerned with the relative difference of values rather than their overall magnitude which is highly dependent on the fixed end-effector velocity that we use.

A. Exploration During Grasping

We first investigate the effect of trading off between exploration and execution time. Varying the *exploration cost* from $\gamma = 0.7$ (minimum exploration) to $\gamma = 0.0$ (maximum exploration) results in a 6% increase in grasp success rate, improving from 74% to 80%, at the expense of 1.4 seconds per grasp on average. Two example trajectories for $\gamma = 0.1$ and $\gamma = 0.7$ are shown in Fig. 5.

As the increased time associated with exploration does not scale linearly with the increase in grasp success rate, the overall efficiency of the system, measured in MPPH, is maximised between the two extremes, for $\gamma = 0.2$, where there is an increase in grasp success rate (79%) but minimal extra time cost. As a result, the MVP controller can be optimised for either raw grasp success rate or overall efficiency by adjusting the cost of exploration.

As γ is increased, the results begin to plateau in the range 0.5 to 0.7 due to the the cost term becoming dominant in this range and causing to controller to perform minimal exploration and converge directly to the grasp.

B. Comparison to Baselines

Single Viewpoint Our approach outperforms the single viewpoint baseline approach in terms of both success rate, improving up to 12% (68% vs. 80%), and MPPH, increasing

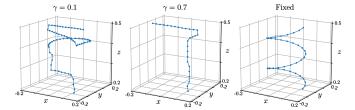|	Ours (MVP Controller)									Baselines			
	Exploration Cost (γ)									*Single View*	*No Expl.*	*Fixed Data Collection*	
	0.0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7			25 Views	50 Views
Total Attempts	174	175	178	171	181	182	179	189	188	196	187	137	125
Failures	35	35	38	36	42	42	45	50	49	62	48	37	28
Mean Viewpoints	44	43	39	36	35	35	35	35	35	1	35	25	50
Success Rate (%)	**80**	**80**	79	79	77	77	75	74	74	68	74	73	78
Mean Time (s)	10.5	10.2	9.8	9.2	9.0	9.1	9.0	9.1	9.1	**8.8**	9.1	11.4	11.4
MPPH	273	282	288	**308**	307	305	299	292	292	281	293	230	245

Fig. 5. Three viewpoint trajectories, for $\gamma = 0.1$, $\gamma = 0.7$ and the *Fixed Data Collection* baseline.
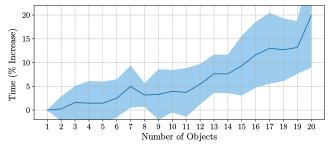

Fig. 6. Mean time per pick increases with the number of objects present, compared to the mean time for one object, averaged over all experimental runs (all values of γ). Shaded area shows one standard deviation.

by approximately 10% from 281 to 308 despite an increased mean time per pick. This reinforces the assertion that considering multiple viewpoints is an effective way to overcome the visual challenges associated with grasping in clutter.

No Exploration The *No Exploration* baseline achieves similar results to our MVP controller when using a high *exploration cost*, which is unsurprising since the high *exploration cost* results in minimal exploration. However, for lower values of γ, our MVP controller outperforms this baseline by 6% success rate, highlighting the added benefit of actively exploring based on uncertainty compared to using a heuristic such as aligning with the best detected grasp.

Fixed Data Collection The *Fixed Data Collection* baseline reinforces the idea that incorporating multiple viewpoints can improve the success rate of a grasping system, with both experiments outperforming the *Single Viewpoint* baseline. Furthermore, the 50-viewpoint experiment outperforms the 25-viewpoint experiment by 5% grasp success rate. However, because the fixed trajectory always views the entire workspace uniformly, it results in a constant execution time which is longer than all other experiments and is unable to focus on salient areas of the workspace. As a result, our MVP controller outperforms this baseline with regards to all metrics, including a higher grasp success rate with fewer viewpoints. The main advantage comes from our information gain approach, which is able to focus on salient areas of the workspace to reduce unnecessary data collection.

C. Automatically Adapting to Scene Complexity

As shown with the *Fixed Data Collection* baseline, using a set of fixed viewpoints for data collection can increase grasp detection accuracy. However, it results in longer, constant execution times, and is unable to focus attention on "interesting" parts of the scene. In contrast, our MVP controller is able to actively adapt to the complexity of the scene and provides more a more efficient data collection process. Fig. 6 shows that for our MVP controller the mean time per pick is dependent on the number of objects in the workspace. As the number of objects in the scene increases, and with it the amount of clutter and potential occlusions, so does the mean time per pick, taking on average 20% longer when 20 objects are present compared to a single object.

VI. CONCLUSION

We presented an active perception approach to grasping in clutter, in which we consider visual grasp detections from multiple viewpoints while reaching. Our work reinforces the importance of viewpoint selection and combining data from multiple viewpoints when grasping in clutter, but in contrast to previous work our Multi-View Picking controller uses an information-gain approach to select informative viewpoints that directly seek to reduce entropy in the grasp pose estimates caused by clutter and occlusions.

We validate our approach with several experiments in grasping from clutter. Our MVP controller achieves up to 80% grasp success while picking from cluttered piles of up to 20 objects, including adversarial objects with complex geometry, outperforming a single-viewpoint method by 12%.

Additionally, by using an information-gain approach, our MVP controller is able to adapt to the complexity of the scene, unlike other approaches which rely on fixed data collection routines as part of a visual grasp detection pipeline. Compared to such a method, our approach results in a higher grasp success rate while also being more efficient, requiring fewer viewpoints and less time per grasp.

REFERENCES

[1] J. Mahler and K. Goldberg, "Learning Deep Policies for Robot Bin Picking by Simulating Robust Grasping Sequences," in *Proc. Conference on Robot Learning (CoRL)*, 2017, pp. 515–524.

[2] A. ten Pas, M. Gualtieri, K. Saenko, and R. Platt, "Grasp Pose Detection in Point Clouds," *The International Journal of Robotics Research (IJRR)*, vol. 36, no. 13-14, pp. 1455–1473, 2017.

[3] E. Arruda, J. Wyatt, and M. Kopicki, "Active vision for dexterous grasping of novel objects," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 2881–2888.

[4] M. Gualtieri and R. Platt, "Viewpoint Selection for Grasp Detection," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 258–264.

[5] I. Lenz, H. Lee, and A. Saxena, "Deep Learning for Detecting Robotic Grasps," *The International Journal of Robotics Research (IJRR)*, vol. 34, no. 4-5, pp. 705–724, 2015.

[6] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics," in *Proc. Robotics: Science and Systems (RSS)*, 2017.

[7] U. Viereck, A. Pas, K. Saenko, and R. Platt, "Learning a visuomotor controller for real world robotic grasping using simulated depth images," in *Proc. Conference on Robot Learning (CoRL)*, 2017, pp. 291–300.

[8] D. Morrison, P. Corke, and J. Leitner, "Closing the Loop for Robotic Grasping: A Real-time, Generative Grasp Synthesis Approach," in *Proc. Robotics: Science and Systems (RSS)*, 2018.

[9] G. Kahn, P. Sujan, S. Patil, S. Bopardikar, J. Ryde, K. Goldberg, and P. Abbeel, "Active Exploration using Trajectory Optimization for Robotic Grasping in the Presence of Occlusions," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 4783–4790.

[10] L. Pinto and A. Gupta, "Supersizing Self-supervision: Learning to Grasp from 50K Tries and 700 Robot Hours," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 3406–3413.

[11] E. Johns, S. Leutenegger, and A. J. Davison, "Deep Learning a Grasp Function for Grasping under Gripper Pose Uncertainty," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 4461–4468.

[12] S. Chen, Y. Li, and N. M. Kwok, "Active vision in robotic systems: A survey of recent developments," *International Journal of Robotics Research IJRR*, vol. 30, no. 11, pp. 1343–1377, 2011.

[13] C. Stachniss, G. Grisetti, and W. Burgard, "Information Gain-based Exploration Using Rao-Blackwellized Particle Filters," in *Proc. Robotics: Science and Systems (RSS)*, vol. 2, 2005, pp. 65–72.

[14] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on robotics*, vol. 21, no. 3, pp. 376–386, 2005.

[15] R. Pito, "A Solution to the Next Best View Problem for Automated Surface Acquisition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 1016–1030, 1999.

[16] S. D. Roy, S. Chaudhury, and S. Banerjee, "Active recognition through next view planning: A survey," *Pattern Recognition*, vol. 37, no. 3, pp. 429–446, 2004.

[17] R. Monica and J. Aleotti, "Contour-based next-best view planning from point cloud segmentation of unknown objects," *Autonomous Robots*, vol. 42, no. 2, pp. 443–458, 2018.

[18] J. Velez, G. Hemann, A. S. Huang, I. Posner, and N. Roy, "Planning to Perceive: Exploiting Mobility for Robust Object Detection," in *Proc. International Conference on Automated Planning and Scheduling (ICAPS)*, 2011, pp. 266–273.

[19] P.-P. Vázquez, M. Feixas, M. Sbert, and W. Heidrich, "Viewpoint Selection using Viewpoint Entropy," in *Proc. Vision Modeling and Visualization Conference*, 2001, pp. 273–280.

[20] H. Van Hoof, O. Kroemer, H. B. Amor, and J. Peters, "Maximally Informative Interaction Learning for Scene Exploration," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 5152–5158.

[21] G. M. Bone, A. Lambert, and M. Edwards, "Automated Modeling and Robotic Grasping of Unknown Three-Dimensional Objects," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2008, pp. 292–298.

[22] J. Aleotti, D. L. Rizzini, and S. Caselli, "Perception and Grasping of Object Parts from Active Robot Exploration," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 3-4, pp. 401–425, 2014.

[23] D. Holz, M. Nieuwenhuisen, D. Droeschel, J. Stückler, A. Berner, J. Li, R. Klein, and S. Behnke, "Active Recognition and Manipulation for Mobile Robot Bin Picking," in *Gearing Up and Accelerating Cross-fertilization between Academic and Industrial Robotics Research in Europe:* Springer, 2014, pp. 133–153.

[24] S. Kumra and C. Kanan, "Robotic Grasp Detection using Deep Convolutional Neural Networks," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 769–776.

[25] J. Redmon and A. Angelova, "Real-Time Grasp Detection Using Convolutional Neural Networks," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 1316–1322.

[26] K. V. Mardia, *Statistics of directional data*. Academic Press, 2015.

[27] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT press, 2005.

[28] M. Guo, D. V. Gealy, J. Liang, J. Mahler, A. Goncalves, S. McKinley, J. A. Ojea, and K. Goldberg, "Design of Parallel-Jaw Gripper Tip Surfaces for Robust Grasping," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 2831–2838.