Low-Resolution Perception for Robotic Packing

Giuseppe Fabio Preziosa * **Federico Vignoni** *
Chiara Castellano * **Marco Faroni** *
Andrea Maria Zanchettin * **Paolo Rocco** *

* *The authors are with Politecnico di Milano, Piazza Leonardo da Vinci, 32. Milano (Italy). e-mail: (giuseppefabio.preziosa, federico.vignoni, chiara2.castellano, marco.faroni, andreamaria.zanchettin, paolo.rocco)@polimi.it*

Abstract: This work tackles the problem of scalable perception for robotic packing with low-cost, low-resolution depth sensing. We propose a framework where reconstruction cues drive next-view selection and grasp evidence updates a per-object stability estimate, jointly deciding *what* to acquire next and *when* to grasp. During the reconstruction, a low-resolution Next Best View (NBV) strategy explicitly avoids redundant views while preserving task-relevant geometry. We validate the approach in two steps: (i) an ablation study of the utility function under very low resolution, and (ii) a full end-to-end evaluation across policies, showing how *low-resolution* perception is a practical, scalable option for robotic packing.

Keywords: Robot perception and sensing, Robotic grasping and manipulation, Smart production and logistics in manufacturing, Robotics in manufacturing systems.

1. INTRODUCTION

Robotic packing lies at the intersection of flexible manufacturing and modern logistics. The task—detecting incoming items, estimating size, selecting and executing grasps, and placing objects into suitable containers—remains challenging despite advances in perception. While the workflow is consistent across applications, variability in object geometry and arrangement makes solutions broadly transferable. In practice, however, container selection and object arrangement still rely on tacit knowledge, limiting repeatability and consistency. Prior work has largely focused on maximizing in-box spatial efficiency: Wang and Hauser (2019) optimize placement under stability constraints, while Wang and Hauser (2021) extend this to both known and unknown objects, typically assuming high-resolution sensing and offline planning. More recent efforts target deployability, using minimalistic grippers (Shome et al. (2019)) and RGB cameras (Hong et al. (2020)) to reduce system complexity. In this context, we investigate whether effective packing requires detailed perception or only task-relevant information, and how far perception can be reduced without compromising throughput and reliability.

Low-resolution sensing has proven effective in other domains (e.g., safety) Ceriani et al. (2013), but object understanding remains challenging due to limited detail and the need for multiple viewpoints. However, robotic packing does not require full 3D reconstruction: task-sufficient estimates of object dimensions are often enough for container selection and placement. Prior work shows that partial reconstructions (25–30%) can support robust manipulation Alliegro et al. (2022); Ten Pas et al. (2017). The challenge becomes more pronounced in multi-object scenarios, where multiple acquisitions are needed and efficient decisions about which object to scan and when it is ready to pick are critical for throughput. Traditional *Next Best View* (NBV) strategies, designed for high-fidelity reconstruction, often lead to redundant poses and limited viewpoint diversity under low resolution. Moreover, dense but partial reconstructions can mislead grasp synthesis, producing grasps that fail on the full object. To address this, we extend our previous work Preziosa et al. (2025) by proposing an end-to-end perception–planning pipeline for multi-object robotic packing with low-cost, low-resolution depth sensors. Specifically, we integrate the LR–NBV strategy from Preziosa et al. (2025), which reduces redundancy through density-aware exploration, within a framework that fuses grasp hypotheses to determine whether an object is ready for picking or requires further views. A video overview is available at `https://youtu.be/9uRI0bNqhDk`.

2. RELATED WORK

The task of selecting the viewpoint that maximizes information about a scene is known as the *Next Best View* (NBV) problem. As denoted by Scott et al. (2003), NBV methods are commonly divided into *model-based* and *model-free* approaches. Model-based methods validate a known object model against observations, while model-free methods—our focus—assume no prior knowledge and iteratively reconstruct the scene. These pipelines typically rely on volumetric representations, such as probabilistic voxel grids (e.g., OctoMap), where space is discretized into free, occupied, and unknown regions. Given a set of sampled poses, the most informative one is typically selected through a utility function, often based on ray casting. These utilities quantify informativeness and can be adapted to both task objectives and hardware constraints.

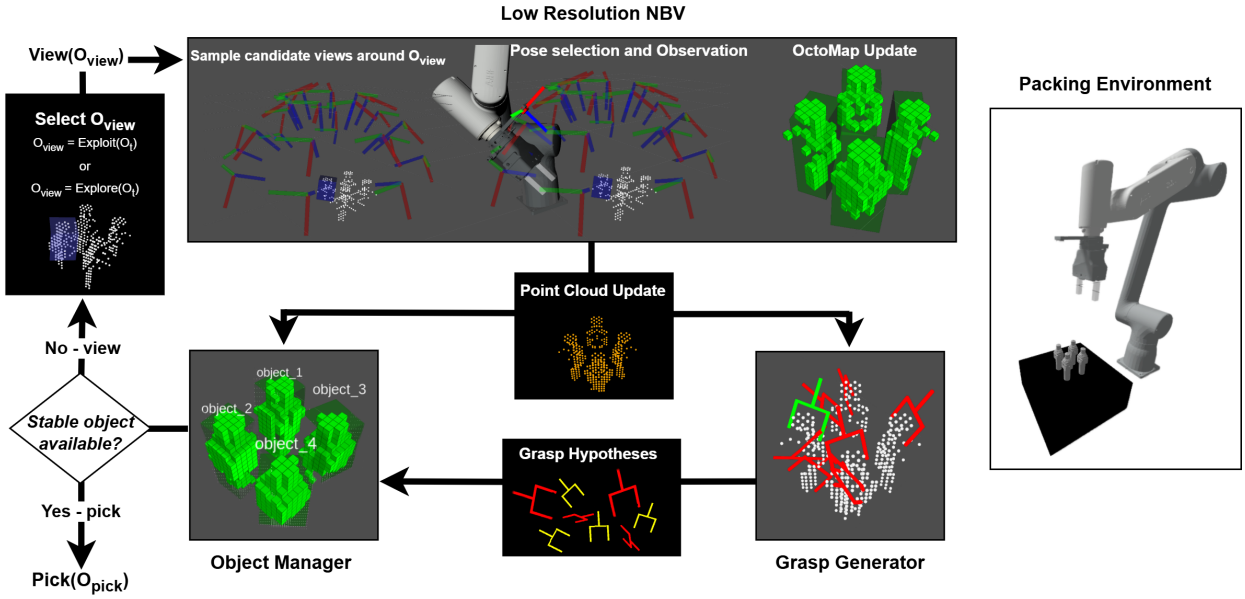

Fig. 1. Pipeline overview. The Object Manager selects *pick* or *view*. For *view*, a target object is chosen (Exploration/Exploitation) and LR–NBV plans the next pose. The Grasp Generator outputs scored 6-DoF grasps. The loop repeats until all objects are removed.

Most classical approaches (Selin et al. (2019)) focus on exploration gain, i.e., maximizing unobserved space. Several work incorporates hardware-aware terms, such as minimizing robot motion or favoring optimal sensor incidence angles (Massios et al. (1998)). However, these formulations are not well suited to low-resolution sensing. Sparse observations make nearby viewpoints hard to distinguish, leading to similar utility scores and redundant acquisitions. While redundancy can be mitigated through penalties on revisited areas as Naazare et al. (2022), such strategies are less effective in object-centric scenarios, where local aliasing between neighboring poses remains a challenge. To address this, we prioritize under-sampled regions while reducing overlap with already observed volumes.

Beyond reconstruction-driven NBV, recent work integrates grasp planning into view selection. Gualtieri and Platt (2017) rank candidate grasps using a view database and alignment heuristics, but their reliance on prior data limits generalization and can bias exploration under partial observations. Similarly, Arruda et al. (2016) refine geometry near predicted contacts for multi-finger grasping, requiring accurate local surface information that is difficult to obtain with low-resolution sensing. In cluttered environments, Morrison et al. (2019) combine pixel-wise grasp proposals with information gain, though their effectiveness relies on high-quality RGB-D data. More closely related, Breyer et al. (2022) propose a closed-loop NBV strategy that iteratively refines grasp hypotheses until a stable grasp is found. This stability criterion remains appealing under low-resolution sensing, as grasp quality can improve with additional views. We adopt a similar principle but extend it to multi-object packing, introducing an object-centric scheduler that determines which item to observe or pick. Unlike Breyer et al. (2022), which assumes known object properties, our method estimates object size and pose online from minimal initial observations and updates them as new data become available.

3. METHOD OVERVIEW

We aim to minimize the number of acquisitions—and thus the makespan—by deciding at each iteration whether to refine an object reconstruction (*view*) or to execute a grasp (*pick*). This decision is taken by the Object Manager (Sec. 3.2) based on two inputs: the current geometric state and the grasp hypotheses. The former is represented by an OctoMap M_t and its associated point cloud P_t, both updated whenever a new observation arrives. If *view* is chosen, LR–NBV (Sec. 3.3) selects the next pose $x_T \in \mathrm{SE}(3)$; the sensor at x_T returns $D(x_T) = \{p_i\}_{i=1}^{R} \subseteq \mathbb{R}^3$, where p_i are the 3D points measured at pose x_T and R is set by the sensor resolution. The observation is fused to update $M_t = \mathcal{F}(M_{t-1}, D(x_T))$ and the derived P_t, and thus the per-object reconstructions extracted from M_t. In parallel, at every map update, the Grasp Generator (Sec. 3.1) takes as input the latest P_t to propose and score candidate grasps; these cues feed back into the Object Manager to arbitrate the pick–vs–view decision at the next iteration. Fig. 1 depicts the flow and module interactions.

3.1 Grasp Generator

Whenever the occupancy map M_t is updated with a new acquisition $D(x_T)$, its point cloud P_t is fed to this module, which returns a list of candidate grasps:

$$G_t = \mathcal{G}(P_t) = \{\, g_{t,1}, \ldots, g_{t,N_t} \,\}, \qquad g_{t,m} = (T_{t,m}, q_{t,m}),$$

where each grasp $g_{t,m}$ consists of a 6-DoF pose $T_{t,m} \in \mathrm{SE}(3)$ and an associated quality score $q_{t,m} \in \mathbb{R}$. The Grasp Generator operates directly on the point cloud and can be either purely geometric or learning-based; in both cases, it estimates plausible gripper poses and assigns a score that reflects grasp stability and feasibility (e.g., antipodal consistency, collision risk via M_t, and reachability). This procedure is executed at every map update so that G_t stays aligned with the evolving scene.

3.2 Object Manager

The Object Manager fuses geometric evidence and grasp hypotheses across iterations to decide *what to view* and *when to pick*. Its guiding idea is *stability by re-observation*: in low-resolution settings, early reconstructions may be partial and induce grasps that reflect only local geometry and are thus unreliable. By monitoring how grasp proposals evolve as the map improves, the module favors grasps that reappear consistently over time and treats them as more trustworthy. To evaluate this stability and arbitrate between further reconstruction (view) and grasping (pick), the Object Manager processes the raw point cloud P_t from the occupancy map M_t and the *global* grasp set produced by the Grasp Generator. With that information, it maintains the scene objects as:

$$\mathcal{O}_t = \{o_t^{(1)}, \dots, o_t^{(N_t)}\},\ o_t^{(n)} = \big(B_t^{(n)},\ \mathcal{G}_t^{\text{obj}}(o_t^{(n)}),\ S_t(o_t^{(n)})\big)$$

where $B_t^{(n)}$ is the oriented bounding box (extent and pose), $\mathcal{G}_t^{\text{obj}}(o_t^{(n)})$ is the per-object grasp list for $o_t^{(n)}$, with generic entry $(T_{o,j},\, q_{o,j},\, c_{o,j},\, \hat{s}_{o,j})$. The first two terms $(T_{o,j}, q_{o,j})$ are as introduced in Sec. 3.1; $c_{o,j} \in \mathbb{R}$ is an observation counter tracking re-occurrences of equivalent grasps; and the grasp score $\hat{s}_{o,j} \in \mathbb{R}$ accounts for both the current quality and the counter, with the specific update detailed in Alg. 1. Finally, $S_t(o_t^{(n)}) \in \mathbb{R}$ denotes the object-level stability used for the pick/view decision. The module comprises two stages: Object Identification, and Pick-or-View Arbitration.

Object identification is performed directly from the raw point cloud P_t, without prior knowledge of object pose or size. The occupancy map M_t is updated by classifying voxels as free, occupied, or uncertain, enabling segmentation of the scene. We apply DBSCAN by Deng (2020) to cluster points based on density. Each cluster defines an object hypothesis, enclosed in an oriented bounding box (OBB). The OBB encodes object extent, defines a dynamic region of interest (ROI) for view planning, and supports collision checking. Temporal consistency is enforced by matching objects in $\mathcal{O}_t$ and $\mathcal{O}_{t-1}$ using centroid displacement and bounding-box overlap, allowing tracking, detection of new objects, and handling of occlusions.

The pick-or-view arbitration performs the core decision of the Object Manager. It first collects representative object poses *(line 2)* and associates each incoming grasp (T_i, q_i) to the nearest object via a pose–distance criterion, then checks for matches with previously stored grasps *(lines 3–6)*. If a match is found, the observation counter c and stability $\hat{s}$ are updated *(lines 8–9)*; otherwise, a new entry is added *(line 11)*. Object-level stability is then computed by aggregating grasp scores *(lines 14–15)*. If an object stability exceeds τ_{stab}, it is selected for picking *(lines 17–20)*; otherwise, a view action is triggered *(lines 23–27)*. In this case, the target object O_{view} is selected by π_{view}: *Exploration* chooses the least-stable object, while *Exploitation* selects the most-stable one, and VIEW(O_{view}) is executed *(line 27)* via LR–NBV (Sec. 3.3).

3.3 Low Resolution Next Best View

This module describes the *view* action used in Alg. 1 *(line 27)* by the Object Manager. Given the view target

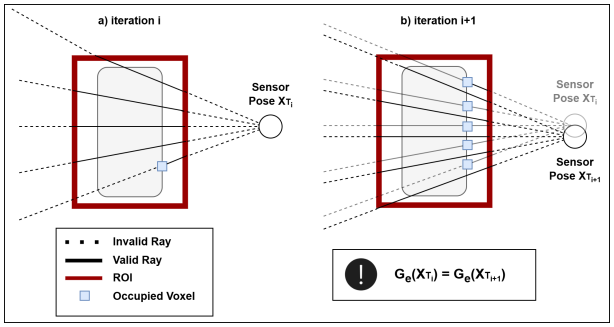

Fig. 2. Ray validity and Exploration gain. Low resolution leads to similar scores and redundant views. Image from Preziosa et al. (2025).

O_{view}, the view–planning module evaluates candidate sensor poses x_T to decide the next acquisition. Candidates are sampled on a viewing sphere centered at O_{view}. The selection among candidate poses is guided by the LR–NBV utility introduced in our prior work Preziosa et al. (2025), which we report here for completeness. The utility is tailored for low-resolution sensing: it balances exploration of uncertain space with a preference for sparse, under-represented regions, thereby discouraging revisits. Formally, it is a weighted sum of two gains—the Exploration gain (G_e) and the Density gain (G_d):

$$U(x_n) = w_e G_e(x_T) + w_d G_d(x_T) \tag{1}$$

with $w_e, w_d > 0$ indicating the per-gain weights; these are adjustable to balance the contributions of each term for a given sensing modality. Both gains rely on a ray-casting procedure that emulates the sensor at a candidate pose. Each ray $\boldsymbol{r} \in \mathcal{R}(x_n, \text{FoV}, \text{res})$ originates from the sensor pose x_T and is given by $\boldsymbol{r}(t) = \mathbf{o} + t\mathbf{d}$, where $\mathbf{o} \in \mathbb{R}^3$ is the origin, $\mathbf{d} \in \mathbb{R}^3$ the direction, and t the distance along

Algorithm 1 Pick Or View

Require: $\mathcal{O}_t$, G_t, thresholds $\tau_d, \tau_g, \tau_{\text{stab}}$
view policy $\pi_{\text{view}} \in \{\text{EXPLORE}, \text{EXPLOIT}\}$

1: **while** $|\mathcal{O}_t| > 0$ **do**
2: $\quad\mathbf{T}^{\text{obj}} \leftarrow [\,\bar{T}_t(o) \text{ for } o \in \mathcal{O}_t\,]$
3: $\quad$**for all** $(T_i, q_i) \in G_t$ **do**
4: $\quad\quad n^\star \leftarrow \text{FINDNEAREST}(T_i, \mathbf{T}^{\text{obj}}; \tau_d)$
5: $\quad\quad\mathbf{T}^{\text{grasp}} \leftarrow [\,T_{n^\star,j} \text{ for } (T_{n^\star,j}, \cdot) \in \mathcal{G}_t^{\text{obj}}(o_t^{(n^\star)})\,]$
6: $\quad\quad j^\star \leftarrow \text{FINDNEAREST}(T_i, \mathbf{T}^{\text{grasp}}; \tau_g)$
7: $\quad\quad$**if** $j^\star \neq$ none **then**
8: $\quad\quad\quad c_{n^\star,j^\star} \leftarrow c_{n^\star,j^\star} + 1$
9: $\quad\quad\quad \hat{s}_{n^\star,j^\star} \leftarrow q_{n^\star,j^\star}\, c_{n^\star,j^\star}$
10: $\quad\quad$**else**
11: $\quad\quad\quad$add $(T_i,\ q_i,\ c{=}1,\ \hat{s}{=}q_i)$ to $\mathcal{G}_t^{\text{obj}}(o_t^{(n^\star)})$
12: $\quad\quad$**end if**
13: $\quad$**end for**
14: $\quad$**for all** $o \in \mathcal{O}_t$ **do**
15: $\quad\quad S_t(o) \leftarrow \sum_{(T,q,c,\hat{s}) \in \mathcal{G}_t^{\text{obj}}(o)} \hat{s}$
16: $\quad$**end for**
17: $\quad\mathcal{R}_t \leftarrow \{\, o \in \mathcal{O}_t \mid S_t(o) \geq \tau_{\text{stab}} \,\}$
18: $\quad$**if** $\mathcal{R}_t \neq \emptyset$ **then**
19: $\quad\quad O_{\text{pick}} \leftarrow \arg\max_{o \in \mathcal{R}_t} S_t(o)$
20: $\quad\quad\text{PICK}(O_{\text{pick}})$
21: $\quad$**else**
22: $\quad\quad$**if** $\pi_{\text{view}} = \text{EXPLORE}$ **then**
23: $\quad\quad\quad O_{\text{view}} \leftarrow \arg\min_{o \in \mathcal{O}_t} S_t(o)$
24: $\quad\quad$**else**
25: $\quad\quad\quad O_{\text{view}} \leftarrow \arg\max_{o \in \mathcal{O}_t} S_t(o)$
26: $\quad\quad$**end if**
27: $\quad\quad\text{VIEW}(O_{\text{view}})$
28: $\quad$**end if**
29: **end while**

the ray. Let t^* be the first point where $\boldsymbol{r}(t)$ intersects the object $o_t^{(i)}$, we define ray validity as:

$$V(\mathbf{r},t) = \begin{cases} 1 & \text{if } \mathbf{r}(t) \cap \text{ROI} \neq \emptyset \text{ and } \exists t^* \leq t : \mathbf{r}(t^*) \in o_t^{(i)}, \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

As illustrated in Fig. 2a, a ray segment is valid only within the ROI and when not occluded by the object, reflecting that regions beyond occlusions are not observable; thus only visible workspace portions contribute to reconstruction. With this, the Exploration gain $G_e(x_T)$ is:

$$G_e(x_T) = \sum_{\boldsymbol{r} \in \mathcal{R}(x_T, \text{FoV}, \text{res})} \sum_{t_k=0}^{1} V(\boldsymbol{r}, t_k) W(\boldsymbol{r}, t_k) \quad (3)$$

where t_k are uniformly spaced samples between $t = 0$ and $t = 1$. The term $W(\boldsymbol{r}, t_k)$ measures the contribution of each segment to volumetric exploration with traversed voxel volume as in Selin et al. (2019). Combining validity and volumetric terms focuses acquisitions on high-uncertainty areas, maximizing novel information. However, as shown in Fig. 2b, under low-resolution sensing, nearby poses can yield identical G_e. Since G_e ignores local point density, it may promote redundant views. To address this, we define the Density gain:

$$G_d(x_T) = \sum_{\boldsymbol{r} \in \mathcal{R}(x_T, \text{FoV}, \text{res})} \sum_{t_k=0}^{1} V(\boldsymbol{r}, t_k) \cdot \frac{1}{1 + \rho(\boldsymbol{r}, t_k)} \quad (4)$$

Here, $\rho(\boldsymbol{r}, t_k)$ is the number of object points within a spherical neighborhood of radius r_d centered at $\boldsymbol{r}(t_k)$. This downweights rays through dense areas and prioritizes under-sampled regions, thereby reducing redundancy.

4. EXPERIMENTAL VALIDATION

We experimentally validate two aspects of the method. Study A (Sec. 4.1) assesses the effectiveness of the proposed low-resolution utility in selecting informative poses under severe sensing constraints. Study B (Sec. 4.2) evaluates the behavior of the full pipeline with the Object Manager—i.e., when to keep reconstructing or pick.

4.1 Study A: Utility Function Validation

This study builds on our previous validation (Preziosa et al. (2025)). We compare LR-NBV with a standard *NBV* method that uses the same *Exploration gain* (Sec. 3.3) but omits *Density gain*, reflecting typical reconstruction-oriented NBV methods. Since the experiment targets *reconstruction only*, both methods use the same marginal-gain stopping criterion: at iteration t, for pose x_{T_t} with utility U_t, the marginal gain $\Delta_t = U_t(x_{T_t}) - U_{t-1}(x_{T_{t-1}})$ is computed, and acquisition stops when $\Delta_t \leq \varepsilon_{\text{mg}}$. As a second reference, we include a heuristic (*HEU*) that samples 13 viewpoints along an ellipse centered on the workspace, assuming known object pose. Each of the four objects in Fig. 3 is reconstructed five times with all methods.

Table 1. Study A parameters.

sampled_poses	w_e	w_d	OctoMap_res[m]
50	500	0.05	0.01

Experiments were carried out on a real UR5e manipulator equipped with a VL53L8CX time-of-flight sensor mounted on the end effector. The device has a 45° × 45° field of view and an 8×8 pixel array, yielding extremely sparse observations. Computation ran on a Lenovo Yoga Pro 7i Gen 9 (Intel® Core™ i7-13700H, 32 GB RAM). Table 1 reports the parameter values used in all experiments. We evaluate performance using four metrics: (i) bounding-box dimensions with intersection-over-union (IoU) to quantify reconstruction accuracy—relevant for container-size selection in packing lines; (ii) number of poses, used as a proxy for time efficiency in packing scenarios; (iii) reconstructed surface coverage; and (iv) count of useless poses, namely viewpoints that fail to increase coverage and may temporarily degrade reconstruction quality. Fig. 3 summarizes the metrics over five runs per object.

We applied the Mann–Whitney U test to compare *LR-NBV*, *NBV*, and *HEU* across all metrics and objects. *LR-NBV* achieves statistically significant gains (p<0.05) in IoU for *horizontal box* and *gripper*. For *vertical box* and *flange*, significant differences (p<0.05) appear in at least one bounding-box dimension (length, width, or height), indicating more accurate size estimates in several cases relative to the baselines. In terms of efficiency, *LR-NBV* consistently requires fewer poses (p<0.05)—often less than half of *NBV*—while maintaining comparable reconstruction percentages. This efficiency stems from better prioritization of informative viewpoints, reducing redundancy. Conversely, *NBV* exhibits more *useless poses*, reducing coverage and requiring extra views.

4.2 Study B: Full Pipeline Evaluation

In this study we evaluate the end-to-end robotic packing pipeline (Sec. 3) under realistic operating conditions. We compare the two view policy (Sec. 3.1): *Exploration XPLR*, which at each step selects the cluster with the *lowest* stability score to spread sensing across objects, and *Exploitation XPLT*, which selects the cluster with the *highest* stability score to focus on a specific object and trigger an early grasp. The two types of objects used in the experiments are shown in Fig. 5, which are actual production parts used in a packing line. To isolate sensing constraints, we evaluate two effective input resolutions: R_{60} (60×60) and R_{30} (30×30).

Experiments were conducted on an *ABB GoFa 12* collaborative arm with a *MaixSense A010* time-of-flight sensor mounted on the end effector (70° × 60° FoV, 100×100 resolution). Inputs were downsampled to the effective resolutions R_{60} and R_{30}. Grasp generation is performed by *GraspNet* (Fang et al. (2020)) in a depth-only configuration, producing 6D grasp hypotheses with associated scores from the point cloud. In each episode, three objects in random poses are provided on a pallet, and the robot transfers them sequentially into a container; a single initial randomized acquisition is adopted. We report: (i) grasp success rate (%), (ii) mean surface reconstruction (% coverage) at grasp time, (iii) failure breakdown by cause, and (iv) makespan to empty the pallet with the required acquisitions to grasp each object. Policy and resolution define a 2 × 2 design (R_{60}/R_{30}, XPLT/XPLR), evaluated on two object types. For each setting, we perform 16 trials (8 per object), yielding 64 trials and $T = 192$ grasp attempts, with no re-attempts. The experiment continues until the pallet is cleared.

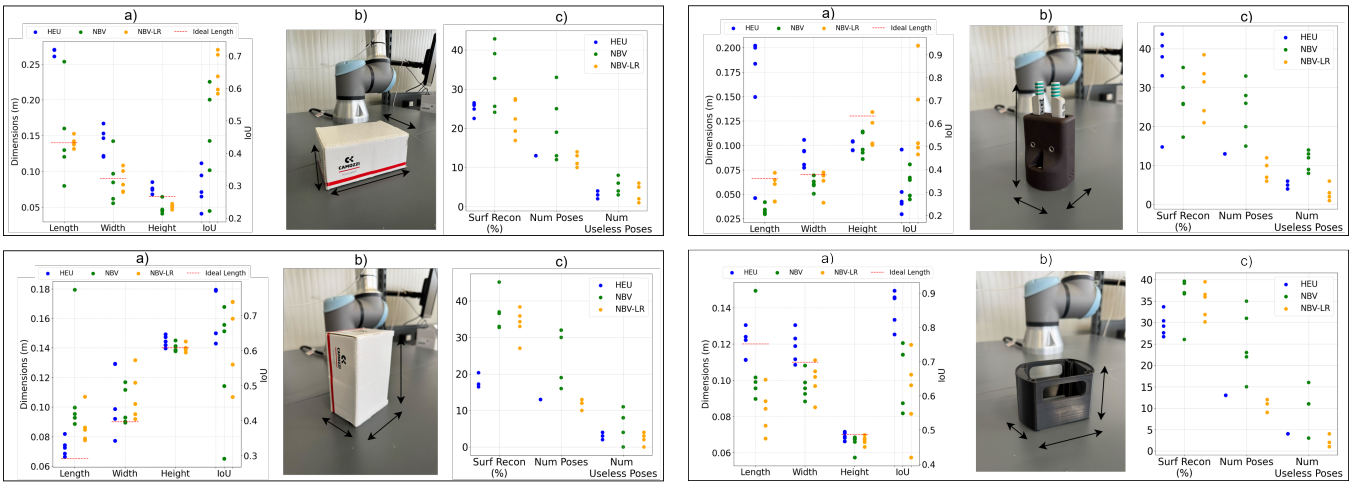

Fig. 3. Acquisition results for four objects: horizontal box (top left), vertical box (bottom left), gripper (top right), and flange (bottom right). For each object: (a) estimated dimensions and IoU; (b) canonical dimensions; (c) surface coverage, number of poses, and useless poses. Image from Preziosa et al. (2025).

Table 2. Grasp success

Object	R60–XPLR	R60–XPLT	R30–XPLR	R30–XPLT
Obj A	62.5%	**70.8%**	54.2%	54.2%
Obj B	66.7%	58.3%	**83.3%**	75.0%

Grasp success We executed $T = 192$ grasp attempts overall: $S = 126$ successes (65.6%) and $F = 66$ failures (34.4%). Table 2 reports success rate over 24 attempts for each approach. Reading the table in light of the object properties visible in Fig. 5—Obj A is much larger, whereas Obj B is smaller and includes reflective areas—two trends emerge. First, higher resolution (R_{60}) makes large parts easier to reconstruct and grasp: denser, more coherent geometry stabilizes pose estimation and improves grasp ranking, explaining better success on Obj A. Second, for small, shiny parts, R_{30} often outperforms R_{60}: higher resolution captures more false reflections from reflective surfaces, adding spurious points that confuse clustering; the lower resolution acts like mild spatial smoothing, reducing noise and yielding more reliable grasps on Obj B.

Table 3. Mean reconstruction at grasp time

Object	R60–XPLR	R60–XPLT	R30–XPLR	R30–XPLT
Obj A	42.55%	**56.29%**	39.79%	44.36%
Obj B	54.77%	**59.83%**	50.45%	47.32%

Surface reconstruction At each grasp trial, we measured the surface reconstruction at the decision point (i.e., when the object was judged grasp-ready). The means in Table 3 lie consistently in the 40–60% range across all policy–resolution pairs and both objects. This aligns with our design goal: the pipeline does not seek exhaustive geometry but a partial, grasp-ready model. At the same time it avoids under-reconstructed cases (below 40%) where grasp hypotheses are possible but unstable.

Grasp failures To interpret the coverage results, failures are classified by cause: i) *STAB* (stability–readiness failure): we deem an object grasp–ready when one or more grasp hypotheses persist across iterations. We classify a failure as STAB when the object was grasp–ready and the proposed grasp is consistent with the locally reconstructed patch, yet execution fails because unreconstructed regions are essential for grasp success. ii) *SPUR*: grasps are proposed on spurious points caused by sensor noise, often amplified by reflective surfaces; iii) *UNSEEN*: the object is never sufficiently observed or not segmented. iv) *EXT*: failures due to external constraints (e.g., motion planning or workspace limitations).

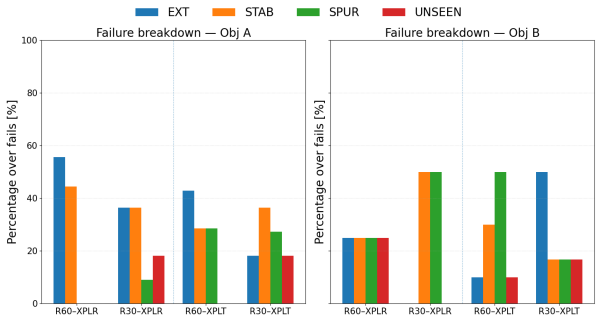

Fig. 4. Failure breakdown by object and condition (percentages over fails)

Across all conditions we observed $F = 66$ failures. Percentages in Fig. 4 are computed *within the failures of each condition*. Focusing on STAB, it accounts for 33.3% of all failures (weighted across conditions). By policy, XPLR averages 37.5% STAB whereas XPLT averages 29.4%, indicating that XPLT reduces false positives by enforcing a stricter, object-specific verification of grasp readiness. Despite this, the stability index under low-resolution reconstruction remains effective, as reflected by the overall grasp success. The breakdown indicates that noise from low-cost, low-resolution sensing is a major error source, strongly influenced by reflectivity and ambient light.

Acquisition curves Fig. 5a and Fig. 5c complement this view by showing the *number of observations* required per object before grasp. In both figures, the two R_{30} curves lie above the corresponding R_{60} curves—consistent with the fact that higher resolution captures more informa-

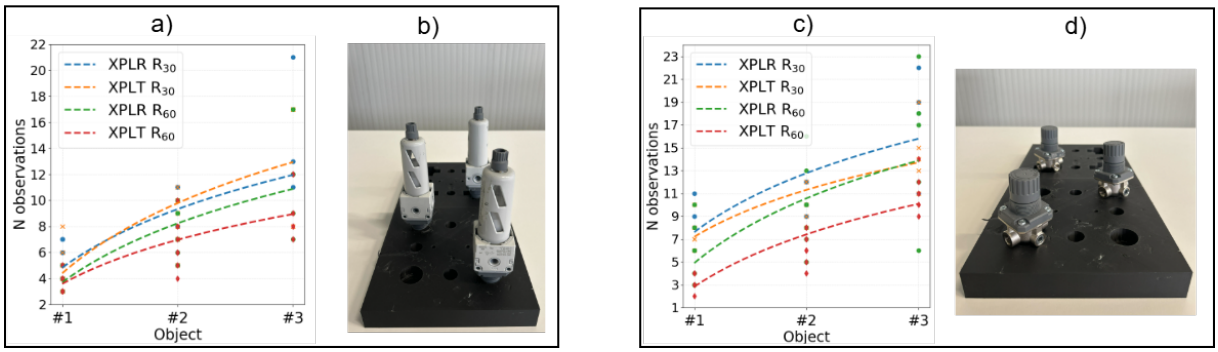

Fig. 5. (a,c) Number of acquisitions required before each grasp for three objects, under different acquisition and resolution modes. The x-axis denotes the grasp index (#1–#3). (b,d) Corresponding experimental setups for Obj A and Obj B.

tive points per view and thus needs fewer acquisitions to reach graspability. All curves exhibit a sublinear progression: after grasping Object #1, many earlier views already contributed evidence for the remaining items, so Objects #2–#3 typically require progressively fewer additional observations. Policy-wise, *XPLR* tracks a clearer logarithmic pattern (it lifts the least-reconstructed item at each step), while *XPLT* is closer to linear because it focuses on a single object until grasp and may leave the next one starting from scratch. This explains the flatter decay of the depth curves noted in both figures; notably, with low-resolution sensing, reflective areas can intermittently corrupt reconstruction and trigger partial tear-down, so—even for the smallest object—the overall makespan can remain comparable to the largest one.

5. CONCLUSION

This paper shows that low-resolution, low-cost perception is a viable solution for robotic packing. We propose an object-centric pipeline that combines a low-resolution NBV strategy (LR–NBV) with grasp-driven reconstruction and a stability by re-observation criterion. Experiments demonstrate that low-resolution sensing can reliably estimate object extents for packing, while the integration of reconstruction and grasp stability enables effective decisions on when to grasp without requiring full geometry. Future work will address more challenging scenarios, including highly reflective objects, by identifying and efficiently sensing problematic regions.

ACKNOWLEDGEMENTS

The authors acknowledge *Camozzi Research Center* for providing robotic equipment, as well as for their support.

REFERENCES

Alliegro, A., Rudorfer, M., Frattin, F., Leonardis, A., and Tommasi, T. (2022). End-to-end learning to grasp via sampling from object point clouds. *IEEE RAL*.

Arruda, E., Wyatt, J., and Kopicki, M. (2016). Active vision for dexterous grasping of novel objects. In *IEEE/RSJ IROS*.

Breyer, M., Ott, L., Siegwart, R., and Chung, J.J. (2022). Closed-loop next-best-view planning for target-driven grasping. In *IEEE/RSJ IROS*.

Ceriani, N.M., Avanzini, G.B., Zanchettin, A.M., Bascetta, L., and Rocco, P. (2013). Optimal placement of spots in distributed proximity sensors for safe human-robot interaction. In *IEEE ICRA*.

Deng, D. (2020). Dbscan clustering algorithm based on density. In *Int. Forum Electr. Eng. Autom.*

Fang, H.S., Wang, C., Gou, M., and Lu, C. (2020). Graspnet-1billion: A large-scale benchmark for general object grasping. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*.

Gualtieri, M. and Platt, R. (2017). Viewpoint selection for grasp detection. In *IEEE/RSJ IROS)*.

Hong, Y.D., Kim, Y.J., and Lee, K.B. (2020). Smart pack: online autonomous object-packing system using rgb-d sensor data. *Sensors*.

Massios, N.A., Fisher, R.B., et al. (1998). A best next view selection algorithm incorporating a quality criterion. In *Proc. British Machine Vision Conference*.

Morrison, D., Corke, P., and Leitner, J. (2019). Multi-view picking: Next-best-view reaching for improved grasping in clutter. In *IEEE ICRA*.

Naazare, M., Rosas, F.G., and Schulz, D. (2022). Online next-best-view planner for 3d-exploration and inspection with a mobile manipulator robot. *IEEE RAL*.

Preziosa, G.F., Castellano, C., Zanchettin, A.M., Faroni, M., and Rocco, P. (2025). Low resolution next best view for robot packing. *IFAC-PapersOnLine*. 14th IFAC Symposium on Robotics ROBOTICS.

Scott, W.R., Roth, G., and Rivest, J.F. (2003). View planning for automated three-dimensional object reconstruction and inspection. *ACM Comput. Surv.*

Selin, M., Tiger, M., Duberg, D., Heintz, F., and Jensfelt, P. (2019). Efficient autonomous exploration planning of large-scale 3-d environments. *IEEE RAL*.

Shome, R., Tang, W.N., Song, C., Mitash, C., Kourtev, H., Yu, J., Boularias, A., and Bekris, K.E. (2019). Towards robust product packing with a minimalistic end-effector. In *IEEE ICRA*.

Ten Pas, A., Gualtieri, M., Saenko, K., and Platt, R. (2017). Grasp pose detection in point clouds. *The Int. Journal of Robotics Research*.

Wang, F. and Hauser, K. (2019). Stable bin packing of non-convex 3d objects with a robot manipulator. In *IEEE ICRA*.

Wang, F. and Hauser, K. (2021). Dense robotic packing of irregular and novel 3d objects. *IEEE Trans. on Robotics*.