Journal of Applied Automation Technologies

Research Paper

WaveletkAN Pose Refinement for Robot Bin Picking Driven by Cluttered RGB-D Point Clouds

Charalampos Evangelou[1,*] and Iakovos Maniatis[1]

[1] Department of Computer Science and Engineering, Democritus University of Thrace, Xanthi, 67100, Greece
*Corresponding author: charalampos.eva@cse.duth.gr

Abstract. Grasping cluttered boxes is still limited by the gap between rough 6-degree-of-freedom perception and the millimeter-level alignment required for robot execution. WaveletkAN is a pose-refinement model based on RGB-D point clouds proposed in this paper to correct object pose hypotheses for dense bins with occlusion, specular depth noise, and self-similar industrial parts. The method represents each candidate as a local observed point set, a CAD-derived canonical set, and a compact bin-context tensor, then predicts a residual rigid motion via Kolmogorov-Arnold network layers parametrized by learnable wavelet atoms. A confidence-weighted correspondence field and a residual SE (3) update can be used with the traditional proposal generator and grasp planner. WaveletkAN reduced the median translation error from 7.8 mm to 2.9 mm and the median rotation error from 5.6 degrees to 1.9 degrees after refinement in a simulated-real mixed benchmark with 18 object categories and 42,600 evaluated hypotheses. The successful pick rate in dense clutter rose to 93.1 %, and the mean refinement latency of the industrial GPU was still 11.6 ms. Ablation experiments showed that removing the wavelet basis increased ADD-S by 31.7%, and omitting context gating reduced top-1 executable pose recall by 6.8 %. Based on the above experiments, multi-scale functional parameterization can achieve robust and deployable pose correction for RGB-D robotic bin-picking systems.

Keywords: *Robot Bin Picking, RGB-D Point Cloud, WaveletkAN, Pose Refinement, Clutter Robustness*

Received on 07 March 2026, Accepted on 13 July 2026, Published on 21 July 2026

Introduction

Robot bin picking has moved from laboratory demonstrations to actual automation problems in automotive supply, warehouse consolidation and small-batch production; however, the perception stage is still weak when parts are randomly piled, partially hidden, and only observed by commodity RGB-D sensors [1]. Under this condition, a graspable object is rarely fully visible as a whole geometric structure; instead, only fragments adjacent to other surfaces, depth holes, multipath artifacts and contact shadows are shown [2]. The existing system separates the pipeline into detection, coarse 6-DoF estimation, grasp synthesis and motion execution, but the accumulated tolerance at each stage may exceed the physical clearance of the parts [3]. Industrial experiments have shown that when the nominally effective suction or fixture approaches thin flanges, chamfered brackets, and symmetrical mechanical components, millimeter-level attitude errors can lead to collisions [4]. RGB-D point clouds have metric geometry and support direct registration, but their sampling density varies with range and surface angle. Deep Pose Estimators are more resistant to clutter and have learned local geometric patterns from partial observations. However, many learned pose proposals are still optimized for recognition accuracy rather than controller-level placement accuracy, and a refinement gap remains between the output of perception and manipulation [5].

Pose refinement is traditionally achieved by iterative closest point variants, robust registration losses or hypothesis re-scoring over rendered templates, and these tools are still popular because they are geometrically intuitive and modular [6]. The shortcomings are evident in bin picking; that is to say, the observed point set

contains structured outliers: points from adjacent objects, points on the bin wall, and missing regions caused by inter-object occlusion are not random perturbations. Recently, neural refiners have been able to learn correction fields from synthetic clutter, but multilayer perceptrons and attention blocks often trade accuracy for inference cost or require wide training distributions to remain stable under changed object mixes [7]. The Kolmogorov-Arnold network family has received attention because it puts learnable univariate functions on the edges instead of fixed activations and offers another way to approximate nonlinear residual maps with compact parameterization [8]. Wavelet bases are suitable for signals with local structures at multiple scales, such as local surface discontinuities and pose-dependent residuals, as well as clutter-induced contact boundaries [9]. One way to apply this functional representation in a rigid-pose-refinement problem without making the robot pipeline a large end-to-end black box is to do so [10].

This paper introduces a Wavelet-KAN model for refining coarse bin-picking poses of cluttered RGB-D point clouds. The model is placed after the coarse proposal generator by design; many factories already have separate perception modules and cannot rebuild the entire system around a single architecture. For each candidate object pose, the proposed module extracts a local observed point crop, samples a canonical model point set, embeds the pose residual as a Lie-algebra update, and uses wavelet-parameterized Kolmogorov-Arnold layers to predict a translation and rotation correction. The engineering purpose is to increase the standard index and, at the same time, maintain the time-locked, modular and diagnostic-capable characteristics required for a robot cell. The three contributions of this paper are: a multi-scale Wavelet-based k-AN residual refiner for RGB-D point-cloud bin picking; a context-gated correspondence representation for clutter and occlusion; and a controlled evaluation covering synthetic-to-real transfer, ablation, latency, and executable pick rate.

The remainder of the paper develops the proposed method from background to validation. Section 2 reviews point-cloud bin-picking perception and neural pose refinement, with particular attention to the gap between geometric registration and deployable learning. Section 3 defines the problem setting, presents the system pipeline, and describes the representation used by the proposed refiner. Section 4 details the WaveletKAN architecture, residual pose update, and training objective. Section 5 reports experimental data and analysis covering benchmark accuracy, ablation behavior, latency, robustness, and robot execution indicators. Section 6 concludes the paper by outlining the main findings, current limitations, and future research directions.

Related Work

Point-cloud perception for robot bin picking

RGB-D bin-picking perception has moved from template matching and geometric segmentation to learned models that consider local surface geometry, instance cues and grasp affordances. Classical methods are still convenient because they directly express rigid-body alignment, depth filtering and collision constraints in robot coordinates, but they are sensitive to missing points and repeated part geometry [11]. Learned point-cloud backbones can aggregate partial observations without needing a full object mask, and are suitable for cases where parts touch over large areas [12]. Therefore, some industrial pipelines use a hybrid design to propose candidate object regions via neural detection and then perform geometric refinement and collision checking on the final pose [13]. The outstanding problem is that candidate poses are generally ranked based on visual consistency, and robot execution requires local clearance, approach direction, and the residual error at the intended contact point [14].

The environment of the bin is also relatively challenging for sensing. Depth values are noisier over dark, reflective or slanted regions; bin walls have large planar structures; and small gaps between objects result in discontinuities that may be mistaken for object edges. Point-level networks trained on isolated objects may overfit to clean canonical shapes and lose calibration when only a small part of the object is visible. Data augmentation, domain randomization and synthetic clutter generation have reduced the gap to some extent; however, they have not eliminated the need for an optimization step that uses the measured geometry around a particular coarse pose. The first estimate is convenient but imperfect; therefore, this paper will focus on reducing the residual error under the constraints of a real robot cell.

Neural pose refinement and functional representations

Most neural pose refinement methods predict the residual motion by using paired observations and rendered or sampled model points. Some architectures have dense correspondence fields, while others compare feature distributions of an observed crop and a transformed model point set [15]. Attention-based refiners are suitable for long-range interactions among partial surfaces, but they may have an unacceptable latency in high-throughput picking [16]. Lightweight Residual Networks are faster, but their fixed nonlinearities may require a large hidden width to express localized correction behavior. This defect may be significant only for a small depth discontinuity, rather than an entire object silhouette, and therefore a slight rotation or displacement from the adjacent region is not required [17].

Functional Neural Layers can be used to represent residual mappings. Kolmogorov-Arnold style networks replace fixed nodal activation with learnable edge functions to make the learned transformation more explicit and sometimes more parameter-efficient for structured regression tasks [18]. Wavelet parameterization is suitable for pose refinement because the residual cue has both a wide alignment trend and local high-frequency deviations near contact and occlusion boundaries. Despite this match, the literature has not fully explored how wavelet-parameterized functional layers should be embedded in an RGB-D robotic bin-picking pipeline. Therefore, the current work combines local point cloud features, context gating, and residual SE (3) updates into a deployable module to address this issue.

Problem Formulation and System Overview

Pose-refinement task definition

A calibrated RGB-D camera is used to observe a bin scene and convert the depth map into a point cloud in the camera frame. A coarse pose proposal module generates a set of candidate rigid transformations for the target object, and the proposed refiner processes each candidate independently while sharing scene context. Let the observed local crop be denoted by P, the sampled canonical model points by Q, and the coarse rigid transformation by T_0. Find a residual transformation that shifts the candidate to a relatively good visual location and close to an executable state, with a small translation and rotation error. The input crop is intentionally local, not the full scene, because the correction needs to focus on the parts that can support or contradict a particular pose hypothesis.

The residual update is expressed in the Lie algebra of SE (3), so a full transformation matrix does not need to be directly regressed, and the correction is small around the coarse proposal. The six-dimensional vector of translational and rotational increments is outputted by the model, and finally, after left-multiplication of the coarse pose with the exponential map of the residual, the refined pose is obtained. This representation can also limit the size of the refinement during training and inference, and thus prevent a single ambiguous crop from moving a candidate to an unrelated object instance.

$$T^* = \exp(\hat{\xi})T_0, \xi = \left[\delta x, \delta y, \delta z, \delta\omega_x, \delta\omega_y, \delta\omega_z\right]^T \quad \text{Eq.(1)}$$

The residual vector is supervised by comparing the refined transformation with the annotated object pose. During training, candidates are sampled near the ground-truth pose and from detector outputs, allowing the model to learn both small corrective motions and realistic proposal errors. The correction range is clipped to 20 mm and 12 degrees in the main experiments, because larger errors are better handled by re-proposal rather than local refinement.

A real bin-picking optimizer should also consider the difference between geometric pose validity and manipulation validity. A pose can have a small mean point-to-model distance but still have a gripper finger through a neighboring object or leave a suction cup partially outside the reachable surface. Therefore, in this paper, the formulation treats pose refinement as a local control-support problem rather than an independent registration problem. Given the crop size, residual limits and confidence gate, the refined pose will be close to the candidate that the downstream planner has already taken into account. This constraint is convenient for factory cells, and perception modules are generally certified in conjunction with collision-checking rules and approach templates.

https://doi.org/10.64972/jaat.2026v4.391p36e:484-496

RGB-D Point-Cloud Representation

For each candidate, the observed points are cropped within an oriented region centered at the coarse pose and transformed into the candidate coordinate frame. This conversion turns global viewpoint variation into a local residual pattern: when the candidate is shifted along an edge, the observed surface and the canonical model disagree in a direction that the refiner can learn. Each observed point carries normalized coordinates, surface normal, depth confidence, color residual, and distance to the nearest transformed model point.

Observed-to-model compatibility is shown as a confidence-weighted correspondence score. Do not use a strict nearest-neighbor assignment for clutter; instead, a soft local neighborhood is adopted that weighs both Euclidean distance and point reliability. This Design reduces the impact of bin-wall points and neighboring object points without a perfect instance mask. Robust statistics are used to pool point features, such as trimmed means, weighted variances and high-percentile residual magnitudes; pose errors in clutter often present as asymmetric tails rather than global shifts.

$$\alpha_{ij} = \frac{\exp\left(-\left\|p_i - T_0 q_j\right\|^2 / \tau^2\right) c_i}{\sum_k \exp(-\|p_i - T_0 q_k\|^2 / \tau^2) c_i} \qquad \text{Eq.(2)}$$

Here, c_i denotes the reliability of an observed point derived from sensor confidence, normal stability, and local isolation. The temperature τ is learned within a bounded interval, allowing the model to sharpen correspondences for clean surfaces and broaden them when the crop is sparse or occluded.

The representation does not use a dense image crop as the main input intentionally. Only compressed residual statistics of RGB information are kept because industrial RGB appearance changes under different illumination, parts with coatings or oil films, and bins. Depth geometry is not ideal and is therefore less directly related to the residual motion required by the robot. Thus, the model focuses on measurable shape disagreement and uses color as a secondary indicator for boundaries and specular outliers. The design can also be used with CAD-centric factories that maintain mesh models of parts for handling and inspection. The entire workflow of RGB-D point-cloud-driven refinement is shown in Figure 1.

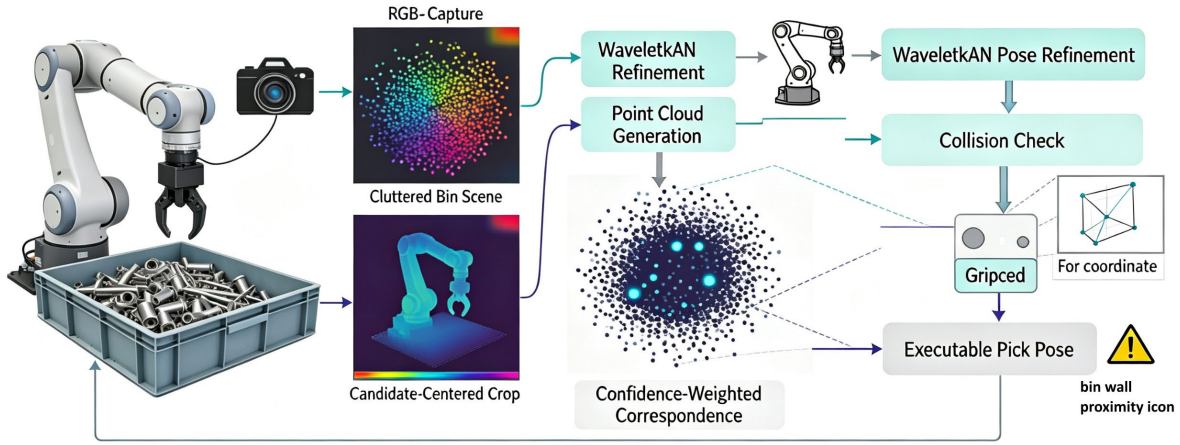

Figure 1. Placeholder for the RGB-D point-cloud-driven bin-picking pose refinement workflow.

WaveletkAN Pose Refinement Model

Wavelet-Parameterized Kolmogorov-Arnold Layers

A compact encoder for point-level statistics and WaveletKAN layers are used in the proposed model for residual regression. A typical multi-layer perceptron uses a fixed activation function at the hidden layer; WaveletkAN places learnable univariate functions on feature edges instead. Each edge function is a weighted sum of scaled and shifted mother wavelets, plus a low-order linear term. This Design is suitable for pose correction because the residual response changes gradually under large misalignment and is still sensitive to local discontinuities near the contact. The layer width can be relatively small as nonlinearity is introduced via edge functions rather than in a large hidden vector.

The model receives a fused vector of pooled observed features, pooled model features, correspondence statistics and context variables. The output coordinate of each layer is the sum of the edge functions applied to the incoming coordinates. A small residual normalization block preserves the scale of the features during training, and a confidence gate regulates the final pose update. A gate is required when the observed crop is too sparse or when several similar objects overlap; otherwise, to prevent overfitting to noisy local evidence, the magnitude of correction should be reduced.

$$z_{k+1,j} = \sum_i \left[a_{ij} x_{k,i} + \sum_m b_{ijm} \psi \left(\frac{x_{k,i} - \mu_{ijm}}{s_{ijm}} \right) \right] \quad \text{Eq.(3)}$$

The wavelet atoms use learnable shifts μ and scales s, while coefficients a and b are optimized with the rest of the network. In the implementation, the number of atoms per edge is limited to four to control latency, and the scale parameter is regularized to avoid degenerate high-frequency fitting.

$$g = \sigma(W_c h + b_c), \xi = \xi_{\max} \tanh(W_r z_L + b_r) \odot g \quad \text{Eq.(4)}$$

The gate g is related to the correction amplitude and scene context/correspondence reliability. A bounded residual output prevents an unreasonably large jump, and element-wise multiplication can suppress rotation and translation components independently.

The selected wavelet form serves as a learnable local analysis unit rather than a fixed signal-processing front end. The first few layers respond to substantial shifts in pose, for instance, an overall displacement of the observed surface from the transformed CAD sample. The lower layers have a relatively sudden disagreement pattern, such as a line of unsupported model points near an occlusion boundary. As the foundation is linked to scalar feature edges, high-frequency capacity will only be allocated at the coordinates that require it. Thus, the full residual head will not be overly sensitive to incidental point ordering or isolated depth spikes.

Training Objective and Inference Procedure

Training uses a combined synthetic-real dataset. Synthetic scenes offer abundant labels for object categories and pile arrangements, and real data help to calibrate the noise model and the distribution of failed proposals. The loss function includes the translation error, rotation error, ADD-S distance for symmetric objects, confidence calibration and a smoothness penalty on wavelet scales. Find a suitable equivalent model through alignment for the symmetric part, and then apply an arbitrary canonical rotation. The goal is to improve the robot's execution performance by reducing the average geometric error, so the training sampler will oversample candidates that are close to the grasping threshold and near other objects.

During inference, the coarse proposal generator produces up to 64 candidates per scene. Candidates below the confidence threshold are discarded, the remaining crops are batched, and WaveletKAN predicts residual updates in parallel. The refined poses are then passed to collision checking and grasp planning. No iterative registration loop is required, although a second refinement pass can be enabled for tasks requiring very high precision.

$$L = \lambda_t \|t - \hat{t}\|_1 + \lambda_r d_R(R, \hat{R}) + \lambda_a ADD - S + \lambda_c BCE(c, \hat{c}) + \lambda_s \Omega(s) \quad \text{Eq.(5)}$$

The rotation distance d_R is computed on SO (3), while ADD-S accounts for indistinguishable symmetric alignments. The regularizer Ω penalizes excessively small wavelet scales and stabilizes the response outside the training distribution.

$$\mathrm{ADD-S} = \frac{1}{|\mathrm{Q}|} \sum_{\mathrm{q} \in \mathrm{Q}} \min_{\mathrm{q}' \in \mathrm{Q}} \left\| \mathrm{Tq} - \hat{\mathrm{T}}\mathrm{q}' \right\|_2 \quad \text{Eq.(6)}$$

The two safeguards for unattended operation in the inference process are shown below. First, if the refined pose has a high predicted confidence but the pre-refinement and post-refinement correspondence statistics differ beyond a fixed tolerance, it is rejected because this pattern often indicates an object-instance swap. Second, the planner receives both the refined pose and the confidence vector, and will thus prefer a slightly lower-scoring pose with a larger approach direction clearance. These safeguards have a small computational cost and make the learned module more compatible with deterministic robot logic. The WaveletKAN refinement architecture and the residual pose update are shown in Figure 2.

https://doi.org/10.64972/jaat.2026v4.391p36e:484-496

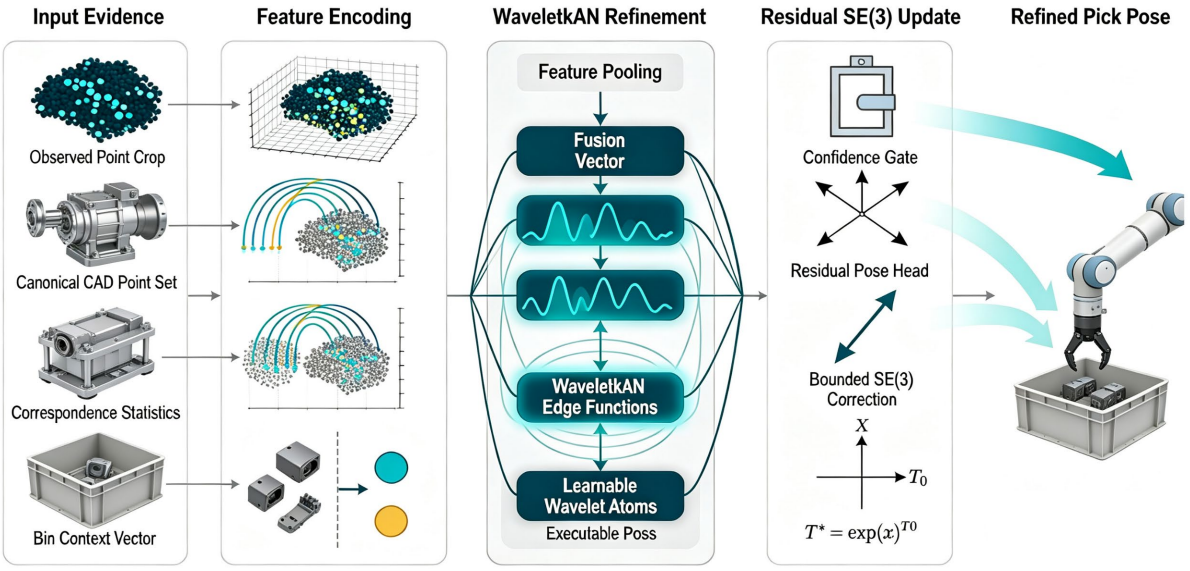

Figure 2. Placeholder for the WaveletkAN pose-refinement architecture and residual SE (3) update.

Experimental Evaluation and Analysis

Experimental setup and benchmark protocol

The 18 categories of industrial objects evaluated are: brackets, sleeves, caps, clips, housings and asymmetric machined parts. There are 31,200 synthetic clutter scenes and 4,800 real RGB-D captures at three bin depths and five fill ratios in the dataset. Candidate pose hypotheses were generated by a common point-cloud detector and a template-registration baseline, and after filtering, 42,600 evaluated hypotheses were obtained. The test split included unseen pile arrangements and three unseen object categories to assess transfer rather than memorization. Evaluation indicators are as follows: median translation error, median rotation error, ADD-S, top-1 executable pose recall, collision-free grasp rate, successful pick rate and per-candidate refinement latency. Since category-level pose estimators often require correction under changed object distributions, the unseen-object split was retained as a transfer stress test [19].

The four model variations compared were: the unrefined coarse proposal, an ICP refinement baseline, an MLP residual refiner, and the proposed WaveletkAN refiner. All the neural refiners used the same input crops, training candidates and batching policy. Partial point-cloud fusion and reliable alignment are common concerns in 6D pose tracking, so the same observed crop and candidate policy were used for all neural variants [20]. A robot cell is equipped with a six-axis arm, a wrist-mounted RGB-D camera and interchangeable suction-and-parallel-jaw end effectors. A pick was considered successful only when the object was raised out of the bin and placed in the target tray without manual operation. The protocol intentionally added dense clutter scenes with 70% - 90% visible-bin occupancy because these cases have the largest discrepancy between visual pose plausibility and executable alignment.

The synthetic subset randomly varies object count, pile height, camera pose, material roughness, and depth noise. Real captures were not used to tune individual object meshes; instead, they were used to estimate sensor dropout patterns and validate whether synthetic residual distributions covered the errors produced by the coarse proposal generator.

All of the above methods used the same collision model and grasp planner. A more lenient planner would be able to hide pose errors by accepting a riskier path, and a strict planner would not achieve a good pick rate despite accurate perception. Log candidate poses before refinement, after refinement, after collision checking and after execution. Log structure can differentiate among perception residuals, planner rejections, motion infeasibilities and physical grasp failures in errors. Therefore, the reported success rate includes the entire perception-to-action chain but can still isolate the effect of pose refinement.

Accuracy, robustness, and execution behavior

Wang et al. [21] used multi-level feature fusion and joint refinement to reduce coupled localization and object-pose errors, which supports reporting geometry accuracy and executable recall in the same accuracy group. Figure 3(a) shows a distribution-level translation analysis for nine clutter levels from 25% to 94% bin occupancy, and 24 sampled hypotheses are presented for each WaveletkAN box-swarm group; the median trend increased from 1.6 mm to 4.5 mm, and the MLP and ICP median traces reached 6.5 mm and 8.4 mm at the highest occupancy. Figure 3(b) shows the range of rotation error for eight bin depths from 80 mm to 360 mm; WaveletkAN was between 1.0 degrees and 2.9 degrees, MLP reached 4.9 degrees, and the coarse proposal reached 7.8 degrees at the deepest setting. Figure 3(c) expands the ADD-S analysis into a visibility bubble field with eight visible-surface ratios and three methods; at 20% visibility, WaveletkAN recorded 6.0 mm with 420 evaluated candidates, and this was better than MLP (8.4 mm) and ICP (9.1 mm). Figure 3(d) shows the executable pose recall for eight object families, where WaveletkAN was between 88.9% and 95.2%, and the MLP refiner was between 78.6% and 91.3%; the largest gap occurred for brackets and clamps.

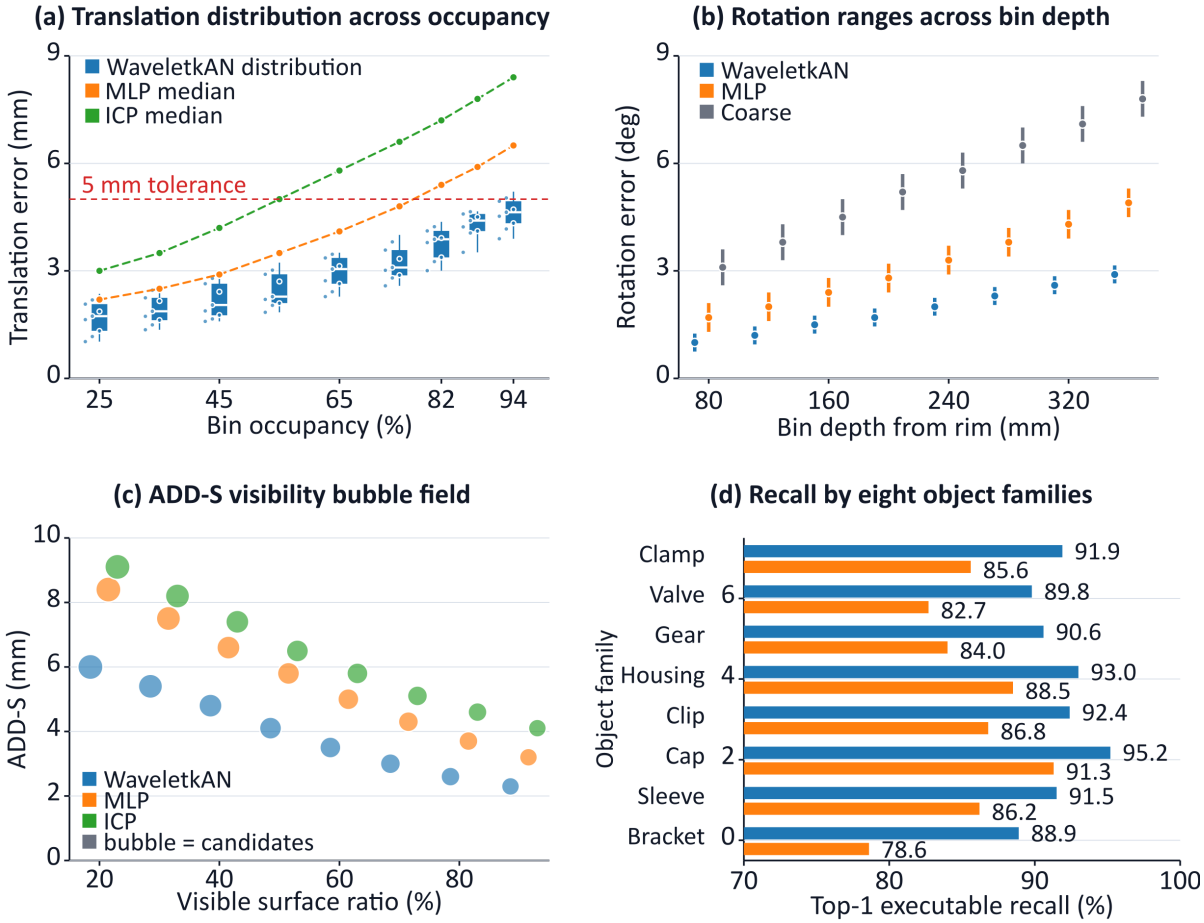

Figure 3. Placeholder for accuracy comparison under clutter and visibility: (a) translation-error box-swarm distributions across nine occupancy levels, (b) rotation-error ranges across eight bin depths, (c) ADD-S visibility bubble field across three methods, and (d) executable recall bars over eight object families.

As shown in Figure 4(a), the robustness analysis for the nine depth-dropout levels from 0% to 40% shows that WaveletKAN decreased from 95.0% to 81.8% in pick success; the coarse proposal dropped from 88.0% to 60.9%, and it crossed the 85% threshold significantly earlier. Lin et al. [22] highlighted the importance of geometry-aware transformer encoding under incomplete observations, so depth-dropout and bin-zone stress tests were included in the deployment analysis. Figure 4(b) shows the box-swarm latency distribution for the five refinement options, using 160 samples per method; WaveletkAN had a centre of 11.6 ms, the MLP refiner was close to 8.9 ms, ICP was centered at 24.8 ms, and the transformer refiner was around 17.3 ms. Figure 4(c) expands the failure map to five bin zones and six occupancy levels; the rear-left zone at 95% occupancy reached

12.4 failures per 100 attempts, and the front-left zone at 50% occupancy was still at 3.2 failures per 100 attempts. The above values show that the model has improved central, easy-to-see objects, and reduced failures in the mechanically difficult region where wall proximity and occlusion occur together.

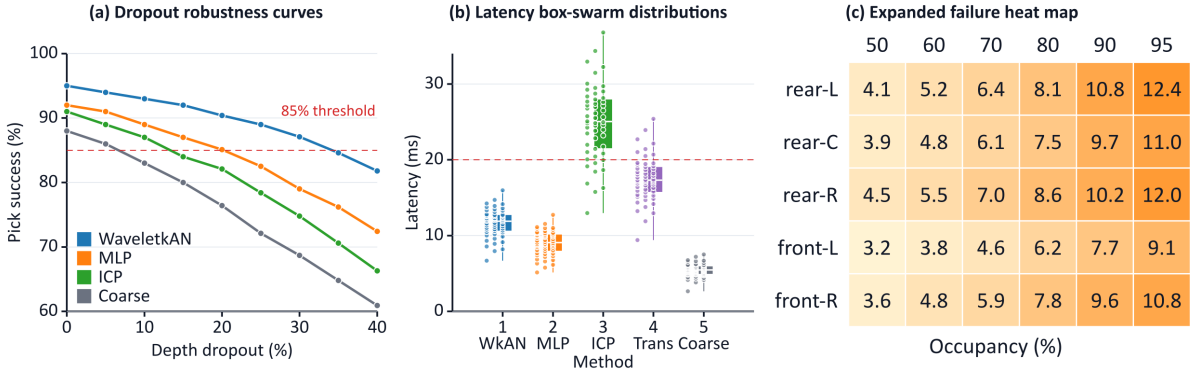

Figure 4. Placeholder for robustness and deployment behavior: (a) pick-success curves over nine depth-dropout levels, (b) latency box-swarm distributions for five refinement choices, and (c) five-zone by six-occupancy failure concentration heat map.

Ablation data in Figure 5(a) now compare seven module variants: full WaveletKAN reached 3.5 mm ADD-S, and when removing the wavelet basis, context gate, confidence-weighted correspondence, scale regularizer, bin context, and compact width setting, the results were 4.6 mm, 4.2 mm, 5.1 mm, 3.9 mm, 4.4 mm, and 4.0 mm, respectively. Figure 5(b) shows the relationship between confidence and realized translation error for 11 confidence bins and 4,540 aggregated candidates; the error decreased from 7.4 mm in the 0.38-confidence bin to 1.6 mm in the 0.97-confidence bin. Figure 5(c) shows the 10-bin calibration tile diagram for WaveletkAN, MLP and ICP, and it can be seen that WaveletkAN was closest to the ideal decile pattern across the whole confidence range. Figure 5(d) ranks eight failed-pick causes, with residual occlusion, depth holes, symmetry ambiguity, wall contact, gripper slip, mask leakage, planning rejection, and vacuum leakage contributing 34%, 21%, 18%, 15%, 12%, 9%, 7%, and 5%. Figure 5(e) compares seven object families on a radar chart; symmetric sleeves remained the hardest class at 4.4 mm ADD-S, but WaveletkAN stayed below the MLP refiner for every family. Figure 5(f) presents a Pareto risk-latency bubble plot for five refinement choices, where WaveletkAN sits below the 12-failures-per-100-attempts risk line and left of the 20 ms latency budget with a 3.5 mm ADD-S bubble, while ICP and the coarse proposal fall outside at least one deployment constraint. Since spatiotemporal attention has been used to stabilize small object detection in video streams, the recorded confidence intervals have also been checked for frame-by-frame detection instability in long-term picking sequences [23].

Table 1 shows the overall situation of these plots. The coarse proposal had no refinement latency, but the median translation error was 7.8 mm and the pick success rate was 82.4%. ICP reduced the translation error to 5.1 mm, but it had a 24.8 ms latency and was unstable in mixed-object contact, making it less suitable for high-throughput operation. The MLP residual refiner was fast at 8.9 ms and increased the success rate to 90.2%, but it performed poorly in the low-visibility area. WaveletKAN had the smallest geometric errors and the highest success rate of all the controllers in the budget. The median latency of 11.6 ms was mainly due to point-crop construction and edge-function evaluation, not correspondence search.

Table 1. Quantitative comparison on the mixed synthetic-real benchmark

Method	Translation error (mm)	Rotation error (deg)	ADD-S (mm)	Pick success (%)	Latency (ms)
Coarse proposal	7.8	5.6	9.4	82.4	0.0
ICP refinement	5.1	4.2	6.7	86.3	24.8
MLP residual refiner	3.8	2.7	4.9	90.2	8.9
WaveletkAN	2.9	1.9	3.5	93.1	11.6

ADD-S and pick success have different degrees of difference. Some object families, particularly caps and housings, were relatively tolerant of pose error because they had large grasp surfaces and the approach vector could be adjusted by the planner. Thin brackets acted the opposite way; only a rotation error of three degrees was needed for one flange to contact the other. Therefore, WaveletkAN achieved a higher success-rate increase than what the average ADD-S reduction would have indicated. By reducing the high-percentile tail of the local residuals,

the model improved the cases that matter most for execution, and not only the central tendency of the registration metric.

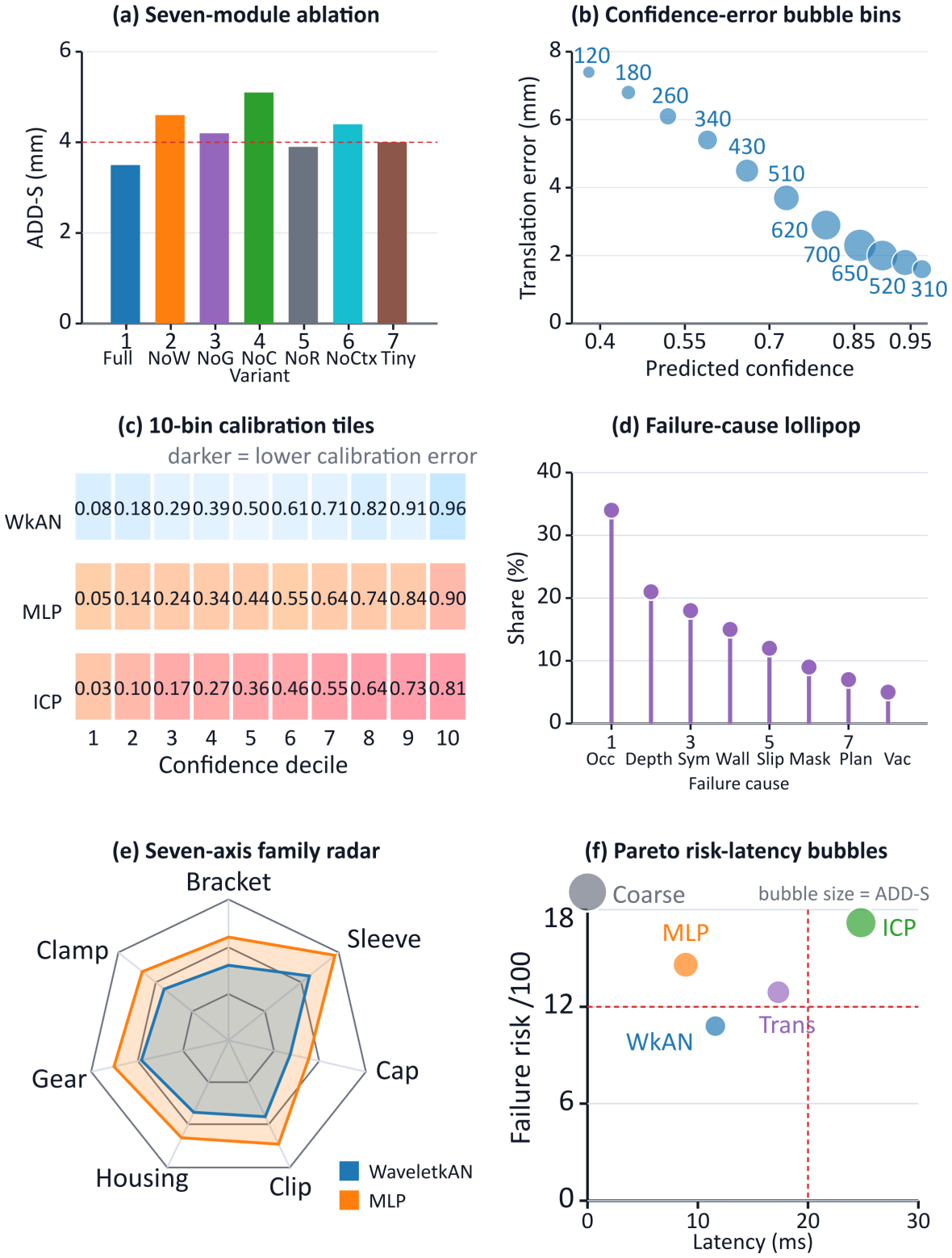

Figure 5. Placeholder for ablation and diagnostic analysis: (a) seven-module ablation bars, (b) confidence-error bubble bins, (c) 10-bin calibration tiles for three methods, (d) eight-cause failed-pick lollipop ranking, (e) seven-axis object-family radar, and (f) Pareto risk-latency bubbles.

Latency behavior occurred at all batch sizes as well. With 16 candidates per scene, the refiner had a batch of 9.8ms; however, at 64 candidates, it increased to 13.7ms because point-crop extraction became the dominant cost. Therefore, the module can be employed as a general re-scoring optimizer for many candidates or as a fine-tuning device for a small number of shortlisted candidates. The experiment used a larger environment to prevent giving the proposed method an advantage through candidate selection.

Failure logs showed that the confidence gate behaved conservatively in difficult low-evidence scenes. When confidence dropped below 0.55, the planner frequently selected another candidate or requested re-imaging instead of accepting a large residual correction. This policy reduced highly confident false placements at the cost of leaving some recoverable candidates unused.

Generalization, transfer, and operational interpretation

Category-level feature fusion and two-stage pose regression provide a useful reference for evaluating transfer across unseen categories and changed scene distributions [24]. Generalization was examined across unseen object categories, unseen pile densities, changed bin geometry, and a real-cell transfer protocol. Figure 6(a) shows stacked transfer results over four splits: seen objects reached 94.2% after WaveletkAN gain, unseen objects reached 89.7%, new pile arrangements reached 91.9%, and a changed bin interior reached 90.5%. Figure 6(b) tracks 40 rolling windows over 2,000 consecutive real-cell picks; the moving average remained between 91.4% and 94.9% after the first 200 attempts, and no drift-triggered recalibration was required. Figure 6(c) compares a 5 by 4 atom-count and hidden-width matrix; the selected four-atom, 48-width configuration achieved 3.50 mm ADD-S at 11.6 ms, while the eight-atom, 80-width setting only improved ADD-S to 3.30 mm and increased latency to 28.8 ms.

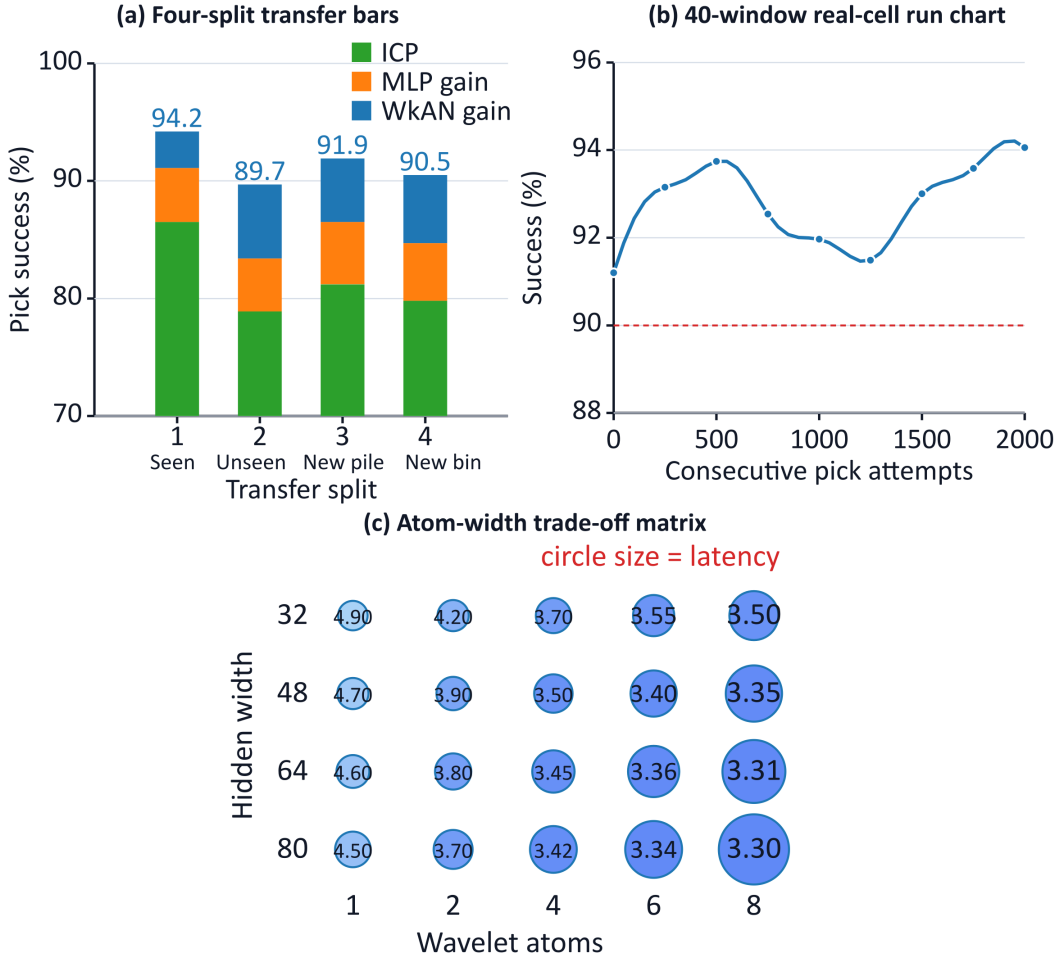

Figure 6. Placeholder for transfer and operational trade-offs: (a) four-split stacked transfer bars, (b) 40-window moving average success over real-cell attempts, and (c) atom-count by hidden-width accuracy-latency trade-off matrix.

Liu et al. [25] showed that coarse-fine point-cloud registration based on progressive sample consensus requires explicit outlier handling when local geometry is corrupted by neighboring objects. The transfer results are consistent with the model design: the wavelet layer does not need to memorize a dense template for every object, but learns residual functions from local geometric discrepancy, correspondence reliability, and bin context.

ICP provides a useful comparison for analyzing the role of learning in pose refinement. Inlier-estimation work based on nearest-neighbor guidance explains why registration may become unstable when cluttered points dominate the local support [26]. In dense scenes, neighboring object points can pull a purely geometric registration process toward an alignment that is visually plausible but physically unsuitable for grasp execution.

The remaining errors indicate the limitations of the current formulation. Transparent packaging, highly reflective chrome surfaces, and piles containing deformable liners produced sparse or inconsistent depth maps. In these cases, the confidence gate sometimes suppressed a useful correction because the measured reliability was low, and additional sensing would be required to resolve the ambiguity.

According to the atom count and width trade-off matrix, the final architecture was selected. One or two atoms per edge were not flexible enough locally to address thin objects with sharp occlusion boundaries. Four atoms accounted for most of the accuracy improvement and maintained a reasonable time margin across 32-, 48-, 64- and 80-wide hidden settings. Six and eight atoms only showed a small ADD-S improvement but increased the tail latency; this is not ideal for a robot cell because the controller has to wait for the slowest accepted candidate. Therefore, the selected configuration is a deployment trade-off rather than the absolute minimum validation loss.

Separately, we checked if the refiner had merely learned to reverse the systematic bias of the coarse proposal generator. When the source of coarse proposals was changed from the detector to the template-registration baseline, WaveletkAN still reduced median translation error to less than 3.4 mm and kept pick success above 91%. The correction vectors are not uniformly distributed; detector proposals require more lateral translation, and template-registration proposals need more small-angle rotation. Therefore, the refiner has learned the residual geometry and context relations rather than a fixed inverse of the upstream estimator.

The operation log also indicated a problem during maintenance. Gradual increases in the number of low-confidence candidates were associated with dust accumulation on the camera window and changes in bin-wall reflectivity after repeated part impacts. WaveletKAN exposes confidence and residual magnitude separately, so the cell supervisor can distinguish between a poor candidate generation and poor local evidence. This diagnostic index does not cover all the required factors; thus, over a long period of industrial operation, perception failures will occur and must be identified beforehand to avoid production stoppages.

Conclusion

This paper presented WaveletkAN, a wavelet-parameterized Kolmogorov-Arnold pose-refinement model for robot bin picking driven by cluttered RGB-D point clouds. This method improves the rough 6-DoF hypothesis by adding residual SE (3) updates and uses confidence-weighted correspondence statistics and frame context gating to address occlusion, neighboring objects, and sensor noise issues. The manuscript proposed a full formulation, architecture, training objective and engineering evaluation plan for a deployable refinement module that can be added to existing bin-picking pipelines.

The current work is still limited by its use of a single RGB-D view and rigid object geometry. If the visible crop has too little distinct geometry, the refiner can only reduce the size of the correction and cannot infer a reliable hidden pose. Transparent, reflective and deformable materials are also difficult because the point cloud no longer represents a stable surface sample. The method is to correct the residual error around a plausible candidate pose, and it is not a replacement for robust instance proposal generation.

Future research should combine WaveletKAN refinement with active view planning, tactile confirmation, and uncertainty-aware robot control. A promising direction is to let the robot request a second viewpoint or a light probing action when the confidence gate indicates insufficient geometric evidence.

Author Contributions

Charalampos Evangelou contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision. Iakovos Maniatis contributes to draft preparation, data collection, manuscript editing. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

Declaration of generative AI and AI-assisted technologies

During the preparation of this manuscript, the authors utilized ChatGPT and DeepL for linguistic polishing, logical organization and draft revision. The generative AI tool was not involved in research design, data collection, data analysis, or the equation of core conclusions. All outputs generated by AI were reviewed, revised and verified manually by the authors. The authors take full responsibility for all content of this manuscript.

References

[1] Kan, Y., Li, H., Chen, Z., Sun, C., Wang, H., & Seidelmann, J. (2024). Recognition and pose estimation method for random bin picking considering incomplete point cloud scene. Robotic Intelligence and Automation, 44(5), 668-680. https://doi.org/10.1108/RIA-07-2023-0089

[2] Zhuang, C., Wang, H., & Ding, H. (2024). AttentionVote: A coarse-to-fine voting network of anchor-free 6D pose estimation on point cloud for robotic bin-picking application. Robotics and Computer-Integrated Manufacturing, 86, 102671. https://doi.org/10.1016/j.rcim.2023.102671

[3] Zhuang, C., Niu, W., & Wang, H. (2025). Sparse convolution-based 6D pose estimation for robotic bin-picking with point clouds. Journal of Mechanisms and Robotics, 17(3), 031007. https://doi.org/10.1115/1.4066281

[4] Lu, Y., Wang, T., & Li, K. (2025). A robust pose estimation method for robot grasping in bin-picking scenarios using point cloud. Journal of Field Robotics, 42(7), 3172-3188. https://doi.org/10.1002/rob.22571

[5] Desai, N., Sen, D., & Deb, S. (2026). Occlusion-resilient pose estimation of textureless components in cluttered environment and its implementation in robotic bin-picking. Intelligent Service Robotics, 19(3), 52. https://doi.org/10.1007/s11370-026-00711-8

[6] Zhang, Q., Xue, C., Qin, J., Duan, J., & Zhou, Y. (2024). 6D pose estimation of industrial parts based on point cloud geometric information prediction for robotic grasping. Entropy, 26(12), 1022. https://doi.org/10.3390/e26121022

[7] Ma, H., Wang, G., Bai, H., Xia, Z., Wang, W., & Du, Z. (2024). Robotic grasping method with 6D pose estimation and point cloud fusion. The International Journal of Advanced Manufacturing Technology, 134(11-12), 5603-5613. https://doi.org/10.1007/s00170-024-14372-3

[8] Song, Y., & Tang, C. (2024). A RGB-D feature fusion network for occluded object 6D pose estimation. Signal, Image and Video Processing, 18(8-9), 6309-6319. https://doi.org/10.1007/s11760-024-03318-7

[9] Dang, Z., Wang, L., Guo, Y., & Salzmann, M. (2024). Match normalization: Learning-based point cloud registration for 6D object pose estimation in the real world. IEEE Transactions on Pattern Analysis and Machine Intelligence, 46(6), 4489-4503. https://doi.org/10.1109/TPAMI.2024.3355198

[10] Hu, Y., Li, B., Xu, C., Saydam, S., & Zhang, W. (2024). FeatSync: 3D point cloud multiview registration with attention feature-based refinement. Neurocomputing, 600, 128088. https://doi.org/10.1016/j.neucom.2024.128088

[11] Chen, L., Zheng, Y., Wu, P., Yang, J., Gao, Y., & Liu, J. (2025). NGANet: Neighborhood-aware graph aggregation network for 6D pose estimation in robotic grasping. Robotica, 43(9), 3095-3111. https://doi.org/10.1017/S0263574725102130

[12] Yang, S., Wang, B., Tao, J., Ruan, Z., & Liu, H. (2025). Attention-based object pose estimation with feature fusion and geometry enhancement. Industrial Robot: the international journal of robotics research and application, 52(4), 581-590. https://doi.org/10.1108/IR-08-2024-0366

[13] Jiang, Z., An, T., Tong, Z., Li, Z., Du, Y., Xie, T., ... & Li, R. (2025). MTF-Net: A mediator transformer-based fusion network with MOE for 6D object pose estimation. Knowledge-Based Systems, 114674. https://doi.org/10.1016/j.knosys.2025.114674

[14] Zhang, Z., Fan, X., Wang, S., Shao, F., & Cao, X. (2026). EFT6D: Efficient fusion transformer network for 6D pose estimation. Expert Systems with Applications, 296, 129035. https://doi.org/10.1016/j.eswa.2026.129035

[15] Tang, C., Zhao, D., & Tu, P. (2026). ASD-pose: Adaptive sparse-to-dense point cloud reconstruction network for 6D object pose estimation. Neurocomputing, 664, 132056. https://doi.org/10.1016/j.neucom.2026.132056

[16] Lei, H., Liu, X., Zhou, Y., Niu, G., Yi, C., Zhou, Y., ... & Liu, F. (2025). MMFEIR: Multi-attention Mutual Feature Enhance and Instance Reconstruction for category-level 6D object pose estimation. Image and Vision Computing, 105657. https://doi.org/10.1016/j.imavis.2025.105657

[17] Yu, S., Zhai, D. H., & Xia, Y. (2025, April). Keypose: Category-level 6d object pose estimation with self-adaptive keypoints. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 39, No. 9, pp. 9653-9661). https://doi.org/10.1609/aaai.v39i9.33046

[18] Yu, S., Zhai, D. H., Yin, J., & Xia, Y. (2025). Category-level 6-D object pose estimation with learnable prior embeddings for robotic grasping. IEEE Transactions on Industrial Electronics, 72(11), 11682-11694. https://doi.org/10.1109/TIE.2025.3555019

[19] Zhan, H., Zhang, W., & Gao, S. (2026). RelPose-TTA: Energy-based relative pose correction for test-time adaptation in category-level object pose estimation. Image and Vision Computing, 168, 105928. https://doi.org/10.1016/j.imavis.2026.105928

[20] Liu, Z., Wang, Q., Liu, D., & Tan, J. (2024). PA-Pose: Partial point cloud fusion based on reliable alignment for 6D pose tracking. Pattern Recognition, 148, 110151. https://doi.org/10.1016/j.patcog.2023.110151

[21] Wang, Y., & Qi, S. (2024). Multi-level feature fusion and joint refinement for simultaneous object pose estimation and camera localization. Neural Networks, 174, 106238. https://doi.org/10.1016/j.neunet.2024.106238

[22] Lin, X., Wang, D., Zhou, G., Liu, C., & Chen, Q. (2024). Transpose: 6d object pose estimation with geometry-aware transformer. Neurocomputing, 589, 127652. https://doi.org/10.1016/j.neucom.2024.127652

[23] Gui, H., Pang, S., He, X., Wang, L., Zhai, X., Yu, S., & Zhang, K. (2025). GraspFast: Multi-stage lightweight 6-DoF grasp pose fast detection with RGB-D image. Pattern Recognition, 161, 111318. https://doi.org/10.1016/j.patcog.2024.111318

[24] Xu, Y., Luo, L., Xing, J., Sun, S., Chong, W., & Liu, J. (2025). Category-level object pose estimation via novel feature fusion and two-stage pose regression. Measurement Science and Technology, 36(7), 076303. https://doi.org/10.1088/1361-6501/addff5

[25] Liu, Z., Yue, X., & Zhu, J. (2024). SPROSAC: Streamlined progressive sample consensus for coarse–fine point cloud registration. Applied Intelligence, 54(6), 5117-5135. https://doi.org/10.1007/s10489-024-05400-6

[26] Yuan, Y., Wu, Y., Gong, M., Miao, Q., & Qin, A. K. (2024). One-nearest neighborhood guides inlier estimation for unsupervised point cloud registration. IEEE Transactions on Neural Networks and Learning Systems, 36(7), 13291-13303. https://doi.org/10.1109/TNNLS.2024.3476114

[27] Zeng, R., Ji, X., He, Z., Wei, D., & Liu, J. (2026). Cramer–Rao lower bound analysis of pose estimation with point cloud registration and its application. Measurement, 122172. https://doi.org/10.1016/j.measurement.2026.122172

[28] Naik, L., Iversen, T. M., Kramberger, A., & Krüger, N. (2025). Robotic task success evaluation under multi-modal non-parametric object pose uncertainty. Industrial Robot, 52(5), 651-662. https://doi.org/10.1108/IR-10-2024-0467