Article | Received 27 July 2026; Revised 30 September 2026; Accepted 18 November 2026; Published XX November 2026
https://doi.org/10.55092/rl20260030

Learning-informed perception and executability-aware selection of seven approach directions for robotic tomato harvesting

Zehuan Feng, Lei Yang, Ziyu Ren, Yun Wang, Kun Xu, Li Wen and Tianxue Zhang*

School of Mechanical Engineering and Automation, Beihang University, Beijing, China

* Correspondence author; E-mail: tianxuezhang@buaa.edu.cn.

Highlights:

- Seven-direction selection jointly evaluates corridor occupancy and executability.
- RGB-D point-cloud corridors represent canopy obstacles without semantic labels.
- Local candidate-pose feasibility and system harvesting success reached 85.7% and 82.0%, respectively.

Abstract: Tomato-harvesting robots often use fixed approach directions or select the clearest local path without assessing robotic execution feasibility. We evaluate seven target-relative directions using a fixed-weight ranking of corridor occupancy, joint motion, numerical singularity, principal component analysis (PCA) geometry, and orientation deviation. A standard YOLO11n-seg model, fine-tuned on greenhouse images, supplies ripe fruit masks, and RealSense D405 depth data locate each fruit in three dimensions. On 28 greenhouse red-green-blue-depth (RGB-D) samples, the method found a locally feasible candidate in 24 cases (85.7%), compared with 20 (71.4%) for Forward-only and 14 (50.0%) for Collision-only. The complete system harvested 82 of 100 tomatoes in consecutive greenhouse trials. These results support joint consideration of corridor occupancy and robot executability. Validation was limited to static RGB-D frames and the AUBO C5 platform.

Keywords: greenhouse tomato; harvesting robot; RGB-D vision; seven-direction pose planning; deterministic candidate ranking; robot executability

1. Introduction

Selective tomato harvesting in protected agriculture is labour-intensive. Harvesting robots offer an important means of alleviating labour shortages and improving operational consistency [1,2]. Recent reviews emphasise that crop perception, end-effector design, task planning, and manipulator motion planning must operate within a single execution pipeline [3]. A remaining challenge is choosing an approach direction that is locally clear and reachable without excessive manipulator reconfiguration.

Detecting and localising a ripe fruit does not determine its approach direction. A fixed approach from the camera, or along the line from the current tool centre point (TCP) to the target, can intersect

Copyright©2026 by the authors. Published by ELSP. This work is licensed under Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium provided the original work is properly cited.

local obstacles. This problem is common when fruit clusters overlap, leaves occlude the target, or the main stem is nearby. Figure 1 illustrates a target tomato, the red-green-blue-depth (RGB-D) observation, and local occlusions in a representative dense canopy.

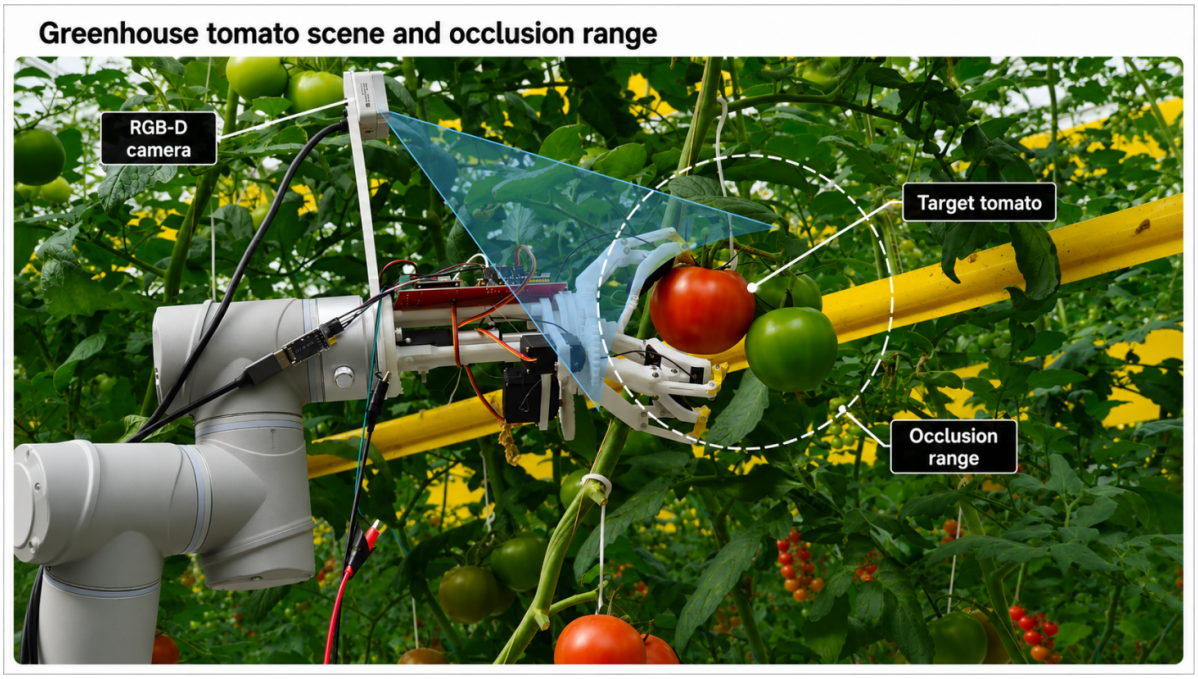


Figure 1. Target tomato, RGB-D camera, and local occlusion region in a dense greenhouse canopy.

In this study, we focus on the pose-planning layer between perception and execution. Tomato harvesting is formulated as selecting an action from several candidate approach directions. Seven target-relative directions constitute a finite and interpretable action space, and each candidate action is represented by features derived from the local point cloud and the current robot state. The deterministic scoring function ranks the candidate actions, and its interface permits the future integration of learning-based ranking methods. This study makes three contributions. First, it links RGB-D ripe-fruit segmentation and localisation to candidate harvesting poses. Second, it represents obstacles along seven approach directions as local point-cloud occupancy. Third, it integrates established inverse-kinematics, joint-motion, singularity, principal component analysis (PCA)-geometry, and orientation-deviation measures in a deterministic ranking and evaluates them under matched offline conditions.

The component parameter, corridor-count threshold, and weights follow geometric constraints and engineering judgement; the ranker is deterministic. Its candidate features could support future learning while controller inverse kinematics (IK) validity and the final joint-motion block remain explicit.

2. Related work

Most existing vision methods prioritise fruit detection or segmentation. Deep-learning methods have markedly improved tomato recognition under variable illumination, occlusion, and ripeness conditions [4,5]. Instance-segmentation, multitask, and Transformer-based models for cherry tomatoes and tomato clusters further improve localisation using RGB-D, peduncle, or main-stem information [6–8]. Previous studies have also validated RGB-D fruit recognition, shape and position reconstruction for occluded fruit, and

integrated harvesting robots for different crops [9–12]. Qi *et al.* combined improved YOLOv8-seg and RGB-D data to measure tomato stem diameter [13]. These advances address perception or system integration but do not directly select a robot-reachable approach direction.

Recent studies have begun to address tomato harvesting-pose estimation and multi-fruit planning. Multi-view sequential planning shows that target order and grasp pose jointly affect efficiency and damage risk in crowded clusters [14]. Chen *et al.* estimated harvesting poses from the spatial relationship between a tomato and its main stem [15]. They constructed an approach direction from the fruit centre and stem axis and validated pose accuracy in a greenhouse. Such methods use agronomic semantics, but the main stem can be occluded, affected by missing depth, or confused with other plant structures. Global motion planners, including improved batch informed trees (BIT*), can jointly consider path length, energy consumption, and collision cost [16]. Recent robot-learning studies have used imitation learning to improve viewpoint planning under occlusion [17] or generated and ranked six-degree-of-freedom grasp candidates directly from point clouds [18]. These findings indicate that approach directions can be constrained by geometric and kinematic priors or ranked through data-driven preferences. For greenhouse tomatoes, the most open geometric direction can still produce excessive joint motion, inverse-kinematics failure, or a near-singular configuration. Obstacle clearance, motion cost, and execution constraints must therefore be considered together [16–20].

Together with related IEEE Transactions on Robotics (T-RO) and IEEE International Conference on Robotics and Automation (ICRA) studies [17,18], this work supports separating local candidate generation from kinematic feasibility checks.

3. Methods

For a selected ripe tomato, the proposed method formulates harvesting-pose planning as a three-stage process comprising candidate generation, constraint evaluation, and ranking. Given an RGB-D image, the corresponding instance mask, the current robot pose, and a local environmental point cloud, the system first estimates the three-dimensional fruit centre. It then generates candidate poses along seven target-relative approach directions. Each candidate is evaluated in terms of local traversability and execution feasibility, after which the pose with the lowest aggregate cost is selected. Figure 2 summarises the overall workflow.

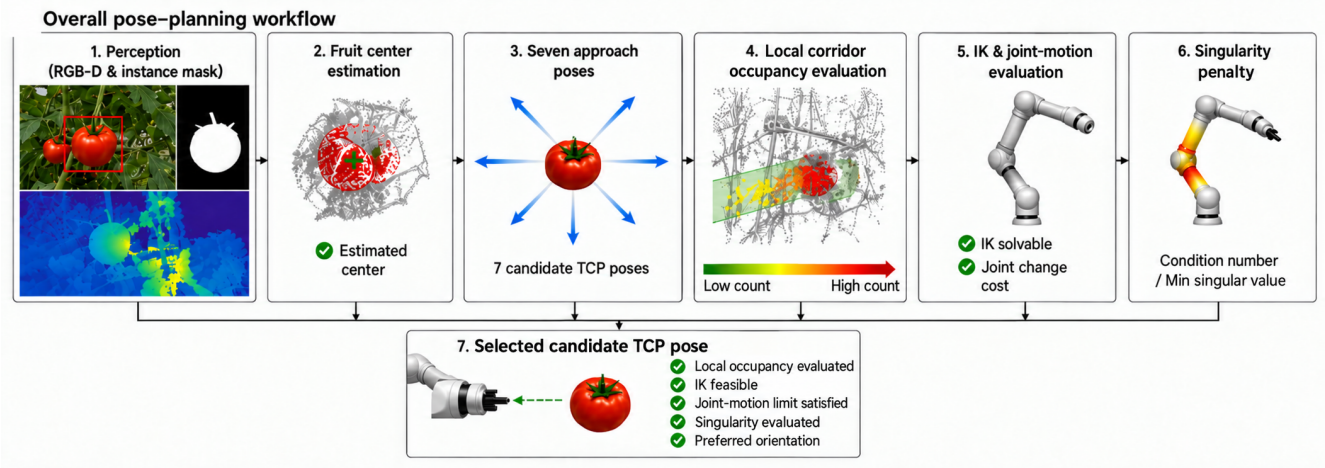


Figure 2. Overall workflow for robot-executability-aware seven-direction harvesting pose planning.

The module checks a local TCP corridor and endpoint IK; it does not verify link swept volumes, self-collision, or a complete configuration-space trajectory. A downstream trajectory planner must validate the selected pose before execution.

3.1. System configuration and notation

The system comprises an RGB-D camera, an AUBO C5 six-degree-of-freedom collaborative manipulator, an instance-segmentation model, and a candidate-pose planning module. A RealSense D405 depth camera captures aligned colour and depth images. The AUBO C5 provides the current TCP pose and six joint angles and executes the planned joint-motion commands. In multi-target scenes, fruit are ordered from left to right by horizontal image position. When horizontal positions are similar, the nearer ripe fruit is prioritised. A candidate target must remain stable briefly before pose planning begins.

To simplify the notation, the manipulator base frame is used as the default coordinate frame. The camera frame is denoted by $\{C\}$, the end-effector frame by $\{G\}$, and the manipulator base frame by $\{B\}$. The target-fruit centre in the camera frame is denoted by p_f^c, and its transformed position in the base frame by p_f. The current TCP position is p_{tcp}, and the current joint configuration is q_0. The set of candidate approach directions is $\mathscr{V} = \{v_k\}_{k=1}^{7}$, where each v_k is a unit vector in the base frame. Unless stated otherwise, all vectors below are expressed in the base frame.

3.2. Ripeness recognition and RGB-D fruit localisation

The colour image is first passed to YOLO11n-seg to obtain object classes, class confidences, bounding boxes, and pixel-level instance masks. Ripeness recognition is formulated as instance-level classification. For each tomato instance i, the model predicts ripeness class $\widehat{c}_i$, confidence s_i, bounding box b_i, and mask M_i. The classes are ripe and unripe tomato. An instance is retained as a harvesting candidate only when $\widehat{c}_i$ denotes ripe tomato and $s_i \geq \tau_r$, where $\tau_r = 0.60$. Unripe tomatoes are excluded to prevent path planning for fruit that are not ready for harvest.

During inference, a low initial confidence threshold of $\tau_0 = 0.15$ and an intersection over union (IoU) threshold of 0.50 are used. This reduces premature suppression of distant, occluded, or partially visible fruit. The system then applies a second class-level filter to ripe tomatoes. The candidate set is

$$\mathscr{T} = \{i \mid \widehat{c}_i = \text{ripe},\ s_i \geq \tau_r,\ z_i \in [z_{\min}, z_{\max}]\}, \tag{1}$$

where z_i is the valid depth near the centre of a candidate instance, $z_{\min} = 0.1$ m, and $z_{\max} = 0.7$ m. The depth constraint rejects instances beyond the close-range harvesting workspace or with anomalous depth measurements. This filtering first separates ripe from unripe You Only Look Once (YOLO) outputs. Instances satisfying the ripeness, confidence, and depth criteria then enter three-dimensional localisation and pose planning.

For each retained ripe-tomato instance, mask moments first provide the two-dimensional centre. The median valid depth within a local neighbourhood suppresses isolated depth spikes. Let $\Omega_i = \{(u,v) \mid M_i(u,v) = 1\}$ be the pixel set of the instance mask. The mask centre is

$$(u_c, v_c) = \frac{\sum_{(u,v)\in\Omega_i} M_i(u,v)(u,v)}{\sum_{(u,v)\in\Omega_i} M_i(u,v)}. \tag{2}$$

Using (u_c, v_c) as the centre of the local window Ω_c, the system retains valid depths and calculates their median:

$$z_c = \text{median}\{D(u,v) \mid (u,v) \in \Omega_c,\ D(u,v) > 0\}. \tag{3}$$

The pixel centre is back-projected into the camera coordinate frame using the camera intrinsics, producing a three-dimensional point near the visible surface. When sufficient valid depth points are available within the mask, a point cloud is sampled and fitted with a sphere by least squares. The fitted sphere centre provides the fruit-centre estimate. The fitted radius is constrained to 2.2–5.0 cm to reject implausible depth noise. Figure 3 shows the localisation procedure from the instance mask and local median depth to sphere fitting or ray compensation.

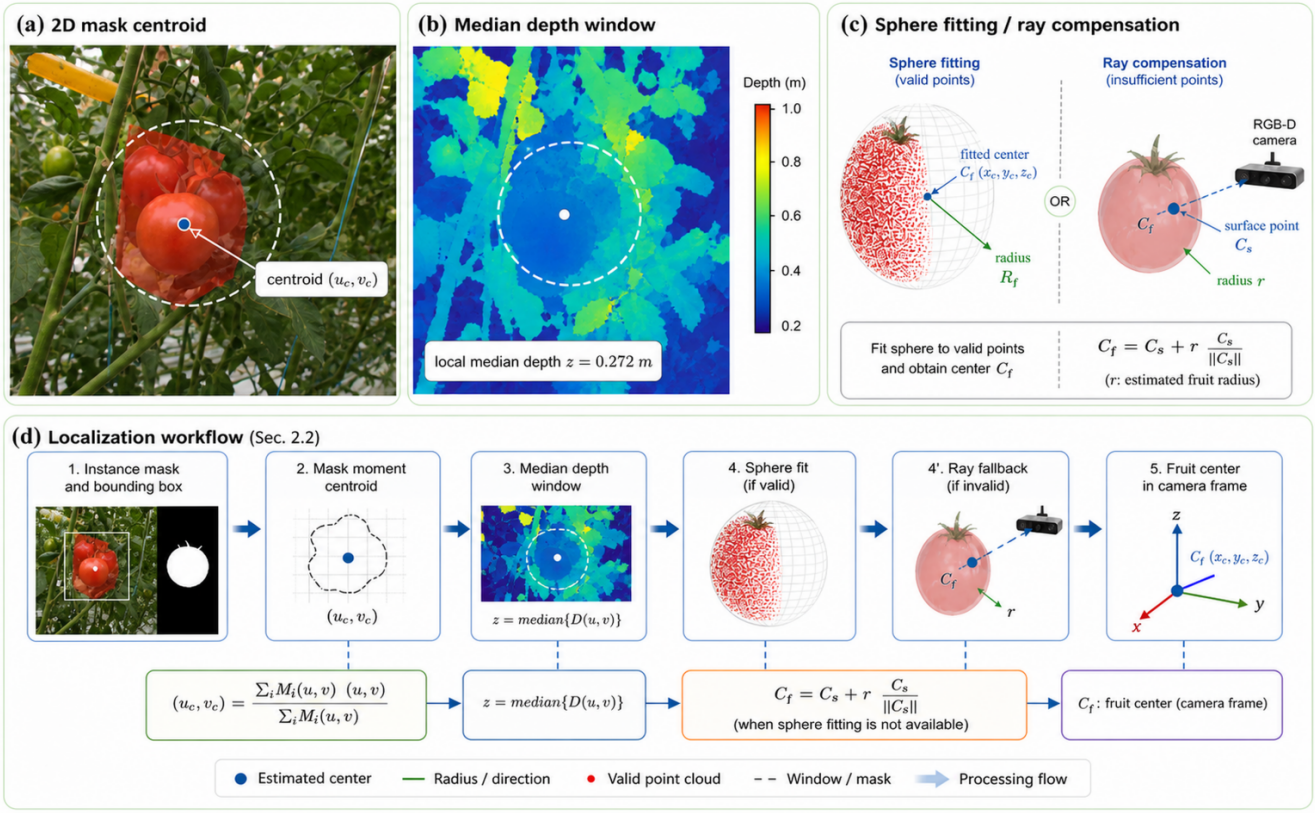


Figure 3. RGB-D fruit-centre localisation using sphere fitting or ray compensation.

When too few point-cloud samples are available or sphere fitting fails, a compensated radius r combines the pixel radius with the historical valid radius. The visible surface point p_s^c is then shifted along the camera ray to estimate the fruit centre:

$$p_f^c = p_s^c + r\frac{p_s^c}{\| p_s^c \|_2}. \tag{4}$$

This compensation maintains planning continuity under occlusion, reflection, and missing depth. Its centre estimate is generally less accurate than sphere fitting with sufficient valid points. After obtaining p_f^c, the fruit centre is mapped to the manipulator base frame using the camera-to-end-effector and end-effector-to-base extrinsic transformations:

$$p_f = R_{BG}(R_{GC}p_f^c + t_{GC}) + t_{BG}. \tag{5}$$

where R_{GC} and t_{GC} are the rotation and translation from the camera to the end-effector frame. R_{BG} and t_{BG} are obtained from the current TCP pose.

3.3. Generation of seven candidate approach vectors

Approaching along the line from the current TCP to the fruit centre assumes both a clear local corridor and a robot configuration that needs little adjustment. In greenhouse environments, however, this line may intersect leaves, the main stem, or neighbouring fruits even when the target remains visible. Moreover, a direct approach may require a substantial change in the manipulator configuration. Let $\hat{f} = (p_f - p_{tcp})/\|p_f - p_{tcp}\|_2$. Projecting the base-frame vertical axis onto the plane normal to $\hat{f}$ and normalising gives $\hat{y}$; if degenerate, we use the base-frame y axis. Set $\hat{x} = \hat{y} \times \hat{f}$. The local orthonormal basis is $(\hat{x}, \hat{y}, \hat{f})$. With $c = \cos\theta$, $s = \sin\theta$, and $\eta = (c^2 + 2s^2)^{1/2}$, the seven unit vectors are

$$\begin{aligned} v_{Fx} &= \hat{f}, \quad v_L = c\hat{f} - s\hat{x}, \quad v_R = c\hat{f} + s\hat{x}, \\ v_{TL} &= \frac{c\hat{f} - s\hat{x} - s\hat{y}}{\eta}, \quad v_{TR} = \frac{c\hat{f} + s\hat{x} - s\hat{y}}{\eta}, \\ v_{BL} &= \frac{c\hat{f} - s\hat{x} + s\hat{y}}{\eta}, \quad v_{BR} = \frac{c\hat{f} + s\hat{x} + s\hat{y}}{\eta}. \end{aligned} \tag{6}$$

The seven directions are task-space choices, not a consequence of the AUBO C5's six degrees of freedom. The controller's numerical IK solver maps each six-degree-of-freedom target TCP pose to joint space. Candidates that fail IK are discarded; wrapped joint motion is used for subsequent filtering and ranking.

The seven-direction design balances directional coverage against real-time computation. Forward preserves the most direct approach corridor, while Left and Right provide the principal lateral detours. TopLeft, TopRight, BottomLeft, and BottomRight add combined upper and lower detours without requiring a computationally expensive continuous global search. Here, θ is a component parameter, not the polar angle of every vector. Two constraints determine this parameter: geometric separation and forward-component dominance. First, the maximum allowable fruit diameter and the collision-corridor radius define the minimum effective separation for the side vectors:

$$d \sin\theta \geq \max(2R_{\max}, r_c), \tag{7}$$

where $R_{\max}$ is the upper bound on the fitted fruit radius, and r_c is the collision-corridor radius. Substituting $d = 0.20\,\mathrm{m}$, $R_{\max} = 0.05\,\mathrm{m}$, and $r_c = 0.06\,\mathrm{m}$ gives the lower bound on the component parameter:

$$\theta \geq \arcsin\left[\frac{\max(2R_{\max}, r_c)}{d}\right] = 30^\circ. \tag{8}$$

Second, diagonal candidates such as TopLeft must retain a dominant forward component:

$$\cos\theta \geq \sqrt{2}\sin\theta \quad \Rightarrow \quad \theta \leq 35.3^\circ. \tag{9}$$

Combining Equation (8) and Equation (9) gives the feasible interval:

$$30^\circ \leq \theta \leq 35.3^\circ. \tag{10}$$

For Left and Right, the actual polar angle from $\hat{f}$ is $\phi = \theta$. For the four diagonal vectors, normalisation gives

$$\phi = \arctan\left(\sqrt{2}\tan\theta\right). \tag{11}$$

At $\theta = 30^\circ$, the diagonal polar angle is $\phi = 39.2^\circ$. The interval $30^\circ \leq \theta \leq 35.3^\circ$ applies to the component construction, with 30° as its lower bound rather than an empirical optimum. At the standoff distance $d = 0.20\,\mathrm{m}$, the lateral displacement of the Left and Right candidates is:

$$\Delta_\perp = d\sin\theta = 0.10\,\mathrm{m}. \tag{12}$$

The lateral displacement of 0.10 m given by Equation (12) satisfies the minimum effective separation. It moves the centre lines of the Left and Right candidates outside the core Forward corridor at the standoff position. The four diagonal candidates provide combined upper and lower detours while retaining a dominant forward component. For each candidate direction v_k, the target TCP position is:

$$p_k = p_f - dv_k, \tag{13}$$

where $d = 0.20\,\mathrm{m}$ is the end-effector standoff distance. The 0.20 m value equals the installed end effector's axial length. It depends on tool geometry, not the manipulator's degrees of freedom, and was not fitted to the experiments. A different tool length or TCP definition requires recalculating d and the geometric angle bound. The target orientation uses an align-z-keep-twist mode. The tool-frame z-axis is aligned with the candidate approach direction while twist is retained as closely as possible relative to the current end-effector orientation. The same rule constructs every target pose, preventing candidate-specific orientation adjustments from conferring an additional advantage.

3.4. Local point-cloud corridor collision evaluation

After generating the candidates, the system determines whether each direction intersects leaves, the main stem, neighbouring fruit, or support structures. Obstacle semantics are not required; the current depth frame directly provides a local environmental point cloud $\mathscr{P}_e$. A cylindrical approach corridor is constructed between the fruit centre and the standoff position. Points within this corridor are counted as potential collisions:

$$N_k = \sum_{p_j \in \mathscr{P}_e} \mathbb{I}\left(-d < h_{jk} < 0,\ \rho_{jk} < r_c\right), \tag{14}$$

where h_{jk} and ρ_{jk} are the axial projection and radial distance of point p_j relative to candidate direction v_k, respectively. The corridor radius is $r_c = 0.06$ m, and $\mathbb{I}(\cdot)$ is the indicator function. A higher collision count indicates more sampled depth points in the local approach corridor. The threshold $N_k < 30$ was chosen empirically by visually judging approach-corridor feasibility in offline samples. In the candidate records, low counts range from 9 to 29 and high counts from 112 to 204; 30 was placed in this gap. Its filtering effectiveness has not been independently validated. It is specific to the depth-sampling and downsampling settings and serves as a conditional candidate filter. If no IK-valid candidate satisfies it, all IK-valid candidates remain eligible for ranking.

Compared with semantic methods based only on main-stem or calyx relationships, point-cloud corridor evaluation is more general. It treats leaves, main stems, unripe fruit, ripe fruit, and support structures uniformly as local spatial occupancy. It therefore does not require precise obstacle classification and is suitable when semantic detection is unreliable.

3.5. Robot-executability-aware aggregate scoring

Selecting only the direction with the lowest collision count does not necessarily yield an executable harvesting pose. Some lateral or downward directions are geometrically open but require large joint rotations, approach joint limits, or cause inverse-kinematics failure. Therefore, we formulate candidate evaluation as a multi-objective ranking problem rather than single-objective obstacle avoidance.

For each direction, the system constructs a target pose and calls the numerical inverse-kinematics interface in the AUBO C5 controller software development kit (SDK). The current joint configuration q_0 is used as the initial solution. IK-related solution metrics are obtained where available and incorporated into candidate evaluation. For each candidate with valid IK, the joint solution is mapped to the equivalent angles nearest the current configuration. Each equivalent joint angle is mapped preferentially to the configured $[-2\pi, 2\pi]$ interval; rejection depends on controller IK validity. A 6×6 numerical Jacobian uses forward-kinematics differences with a joint step of 10^{-4} rad. Its translational block uses base-frame position differences in m/rad; its rotational block uses rotvec$(R^\top R_j)$ in rad/rad, expressed in the unperturbed tool frame, where R and R_j are the tool-to-base rotations before and after perturbation. No characteristic-length scaling is applied. The condition number and minimum singular value are computed from this matrix.

The fruit-mask point cloud also provides a weak geometric prior. PCA is applied to the camera-frame tomato point cloud after outlier removal. Let $\lambda_{\min}$ and $\lambda_{\max}$ be the smallest and largest eigenvalues, and let $e_{\min} = (e_x, e_y, e_z)^\top$ be the minimum-eigenvalue eigenvector. We compute

$$r = \frac{\lambda_{\min}}{\lambda_{\max}}, \qquad a = R_{BC} \frac{(e_x, e_y, 0.5e_z)^\top}{\|(e_x, e_y, 0.5e_z)^\top\|_2}, \qquad A_k = 1 - \left|v_k^\top a\right| .A_k = 1 - \left|v_k^\top a\right|. \tag{15}$$

The factor 0.5 down-weights the camera depth-axis component before normalisation; R_{BC} rotates the resulting direction into the base frame used for v_k. The threshold $r \leq 0.75$ is an engineering cutoff requiring at least 25% normalised separation between $\lambda_{\min}$ and $\lambda_{\max}$. For $r > 0.75$, the Forward vector replaces a, suppressing PCA-based direction preferences for near-isotropic observed point clouds. The ratio describes the visible point cloud and depends on fruit geometry, occlusion, and depth sampling.

The aggregate score comprises collision, PCA-alignment, joint-motion, singularity, orientation-deviation, and large-joint-motion penalties:

$$S_k = w_C C_k + w_A A_k + w_D D_k + w_G G_k + w_O O_k + w_L L_k. \tag{16}$$

The score components required to reproduce the implemented ranking are defined one per line as

$$
\begin{aligned}
\delta q_k &= \operatorname{atan2}[\sin(q_k - q_0), \cos(q_k - q_0)], \\
C_k &= N_k/30, \\
D_k &= \|\delta q_k\|_2, \\
D_k^{\max} &= \|\delta q_k\|_\infty, \\
G_k &= \operatorname{clip}_{[0,4]}(\log_{10} \max(\kappa_k, 1)) + \mathbb{I}(\sigma_k^{\min} \le 10^{-4}) + \mathbb{I}(\kappa_k > 1200), \\
O_k &= \|\delta r_k\|_2/30^\circ, \\
L_k &= \max(0, D_k^{\max} - 120^\circ)/120^\circ.
\end{aligned}
\tag{17}
$$

Here, N_k is a point count; D_k and $D_k^{\max}$ are in radians; κ_k and $\sigma_k^{\min}$ are the Jacobian condition number and minimum singular value; and δr_k is the soft-lock Euler offset in degrees. The clip operator bounds only the logarithmic term. The experimental align-z-keep-twist mode gives $O_k = 0$. A candidate with unavailable Jacobian metrics receives $G_k = 0$ and remains eligible during singularity filtering. The fixed weights are $w_C = 1.6$, $w_A = 0.55$, $w_D = 1.2$, $w_G = 0.45$, $w_O = 0.15$, and $w_L = 1.0$.

Algorithm 1 Candidate-pose selection

Require: Seven candidates in the fixed direction order, initial joints q_0

$V \leftarrow$ candidates with valid IK
if $V = \varnothing$ **then**
    Retry the minimum-N_k target pose once with the same q_0, allowing a larger number of solver iterations
    $k^\star \leftarrow$ retry candidate
    $k^\star \leftarrow$ retry candidate
else
    if $\exists k \in V : N_k < 30$ **then**
        $V \leftarrow \{k \in V : N_k < 30\}$
    end if
    $U \leftarrow \{k \in V :$ Jacobian unavailable or $(\kappa_k \le 1200 \wedge \sigma_k^{\min} \ge 10^{-4})\}$
    if $U \neq \varnothing$ **then**
        $V \leftarrow U$
    end if
    if $\exists k \in V : D_k^{\max} \le 160^\circ$ **then**
        $V \leftarrow \{k \in V : D_k^{\max} \le 160^\circ\}$
    end if
    Sort V by (S_k, N_k), preserving the fixed direction order for ties
    $k^\star \leftarrow$ first candidate in V
end if
if IK is invalid or a finite six-joint solution is unavailable **then**
    Return failure
end if
Map the solution to the nearest equivalent angles and compute $D_{k^\star}^{\max}$ from q_0
if $D_{k^\star}^{\max} > 160^\circ$ **then**
    Return $k^\star$ as blocked
else
    Return $k^\star$
end if

IK failure and the final 160° joint-motion block are hard conditions on both the normal and retry paths. The $N_k < 30$ and singularity tests are conditional gates with fallback, whereas the six terms in Equation (16) rank candidates within the retained pool.

For within-sample diagnostics, the relative occupancy score is defined as

$$Q_k = \begin{cases} \dfrac{N_{\max} - N_k}{N_{\max} - N_{\min}}, & N_{\max} > N_{\min}, \\ 1, & N_{\max} = N_{\min}, \end{cases} \tag{18}$$

where $N_{\min}$ and $N_{\max}$ are the minimum and maximum counts among the seven directions for the same sample. Q_k is a within-sample relative occupancy score; it does not represent physical clearance or collision safety.

4. Experiments and results

This section is organized to mirror the experimental workflow used in this study. We first describe the training dataset, offline RGB-D samples, AUBO C5 robotic platform, and evaluation metrics. We then report instance-segmentation performance. Next, we present an offline ablation across four component settings and compare the Forward-only, Collision-only, and proposed planning strategies. Finally, we analyze approach-direction selection and offline failure modes before reporting the results of 100 consecutive harvesting trials.

4.1. Experimental data, platform, and evaluation protocol

The instance-segmentation model was trained on the Tomato-Harvest.v3i.yolov11 dataset. Images were collected in greenhouse tomato rows at Beijing Shounong Cuihu Factory, Beijing, China. The data cover mixed ripe and unripe fruit, individual fruit and clusters, leaf occlusion, crossing branches, suspended vines, and support structures. They also include close-range views from the manipulator workspace. After annotation, the dataset was exported in the YOLOv11 instance-segmentation format, and all images were resized to 640×640 pixels. The annotated classes were ripe-tomato and unripe-tomato. Only ripe-tomato instances were considered harvesting candidates.

The dataset contains 1310 images and 8898 annotated instances. Of these images, 1259 were used for training, 35 for validation, and 16 for testing. We used the standard YOLO11n-seg architecture without structural modification. Starting from pretrained weights, the model was fine-tuned for 100 epochs with an input size of 640 and a batch size of 8.

We chose YOLO11n-seg to limit perception overhead and support future embedded deployment. All three strategies use the same masks, which keeps the perception input fixed across the planning comparison. The present evaluation covers the nano variant only.

The offline comparison used 28 RGB-D samples collected during one afternoon at Beijing Shounong Cuihu Factory. Each sample included a colour image, raw depth array, depth visualisation, and metadata. Figure 4 shows shaded, mixed-light, and bright-background views with leaf and branch occlusion, overlapping fruit, and support structures. The samples capture within-session variation, not controlled time-of-day or dynamic-scene effects.

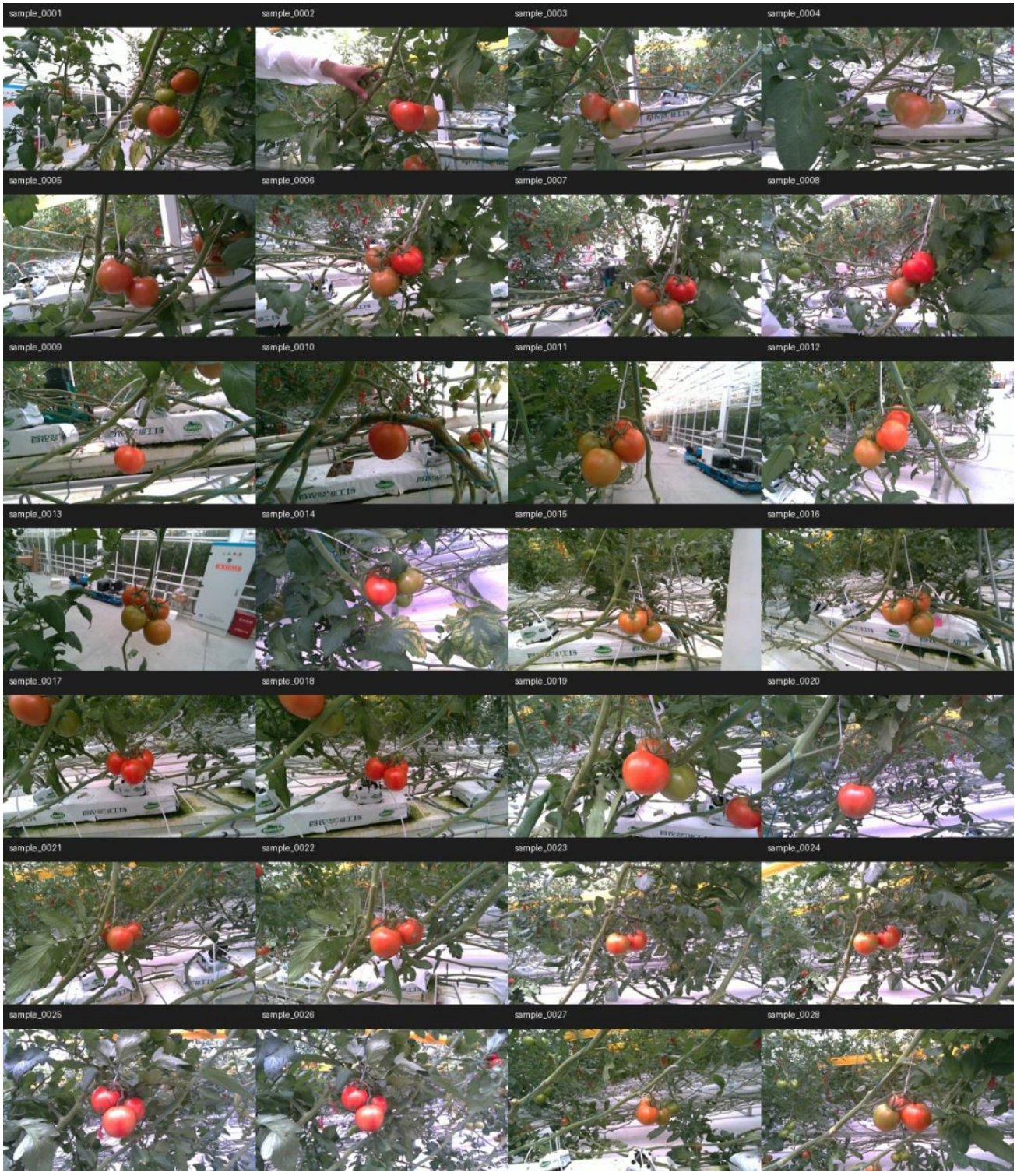


Figure 4. The 28 greenhouse RGB-D samples collected during a single afternoon field session. The montage includes shaded, mixed-light, and bright-background views, as well as leaf and branch occlusion, overlapping fruit, and support structures.

The three strategies were evaluated on the same RGB-D samples. Forward-only always selected the forward direction from the current TCP to the target tomato. Collision-only selected the direction with the lowest point-cloud collision count from the seven candidates. The proposed method selected an approach by combining corridor occupancy, IK-related metrics, joint-motion cost, singularity, and directional score.

We recorded the proportion of samples containing at least one IK-valid candidate, local candidate-pose feasibility, the within-sample relative occupancy score, and maximum joint displacement. For the selected candidate, the relative-occupancy pass was $g_k = \mathbb{I}(N_k \leq N_F)\mathbb{I}(Q_k \geq 0.5)$, where N_F is the Forward count. Local feasibility required $g_k = 1$, controller IK return code 0, and $D_k^{\max} \leq 160^\circ$; complete collision-free motion remains subject to downstream trajectory validation. The 28 samples were analysed as one natural-greenhouse dataset because low-, medium-, and high-occlusion labels were unavailable.

Figure 4 provides an overview of the 28 RGB-D samples, in which target fruit frequently overlap with leaves, branches, and support structures. Figure 5 shows the manipulator platform and greenhouse used for data collection and validation. Testing each strategy on the same targets isolates the planning layer and allows corridor occupancy and robot executability to be compared directly.



Figure 5. Manipulator platform and greenhouse experimental environment at Beijing Shounong Cuihu Factory.

4.2. Perception-model training results

The YOLO11n-seg model was trained for 100 epochs. Using mask mean average precision at IoU = 0.50 as the selection criterion, the best checkpoint occurred at epoch 29. At this checkpoint, mask precision was 0.8852, recall was 0.8520, mAP50 was 0.9227, and mAP50-95 was 0.6161. When mAP50-95 was used instead, the maximum occurred at epoch 80, with mAP50 of 0.9084 and mAP50-95 of 0.6351. At the final epoch, mask mAP50 and mAP50-95 were 0.8859 and 0.6085, respectively.

A post hoc reanalysis of the 28 RGB-D frames estimated fruit-centre sensitivity to point sampling using the saved segmentation checkpoint. In each frame, we selected the leftmost ripe-fruit mask satisfying the confidence and depth criteria. Sphere fitting met the 22–50 mm radius criterion in 27 frames; the remaining frame used ray compensation. For each accepted fit, we resampled occupied 8×8-pixel blocks 1000 times, holding the mask fixed and refitting the sphere. The root mean square (RMS) centre displacement from the original fit was 2.70 ± 3.18 mm across 27 frames (mean $\pm$ sample standard deviation; range, 0.58–12.10 mm). The per-frame 95th percentiles ranged from 0.96 to 24.02 mm. These values quantify conditional fitting uncertainty; absolute localisation accuracy additionally depends on segmentation, surface visibility, depth bias, and camera calibration.

4.3. Offline ablation over four component-angle settings

We evaluated $\theta = 25°$, $30°$, $35°$, and $40°$ on the same 28 RGB-D samples. The corresponding diagonal polar angles ϕ were 33.4°, 39.2°, 44.7°, and 49.9°. The 25° and 40° cases were out-of-range sensitivity settings, and all other candidate-evaluation procedures were unchanged.

Table 1. Offline ablation of the proposed strategy at four component-angle settings ($n = 28$; paired samples).

θ	IK-valid samples/Local feasibility	Within-sample relative occupancy score Q	Max. joint change (°)	Time (ms)
25°	22/28 (78.6%)/21/28 (75.0%)	0.9423 ± 0.0779	66.3 ± 15.8	8.86 ± 2.85
30°	24/28 (85.7%)/24/28 (85.7%)	0.8516 ± 0.1214	58.1 ± 9.4	8.66 ± 1.93
35°	22/28 (78.6%)/22/28 (78.6%)	0.9184 ± 0.0916	69.0 ± 16.3	8.68 ± 2.79
40°	18/28 (64.3%)/18/28 (64.3%)	0.9248 ± 0.0905	72.6 ± 17.4	9.19 ± 3.00

Note: θ is the construction parameter in Equation (6); the corresponding polar angle of the diagonal vectors is $\phi = \arctan(\sqrt{2}\tan\theta)$, giving $\phi \approx 39.2°$ for $\theta = 30°$. Values after $\pm$ are sample standard deviations.

The 30° setting achieved the highest local candidate-pose feasibility rate: 24/28 (85.7%; Wilson 95% CI, 68.5%–94.3%). At 25°, 35°, and 40°, the rates were 21/28 (75.0%), 22/28 (78.6%), and 18/28 (64.3%), respectively. Mean maximum joint displacement at 30° was $58.1° \pm 9.4°$, the lowest of the four settings.

The mean Q values describe the relative rank of selected directions within individual samples and are not measurements of physical clearance.

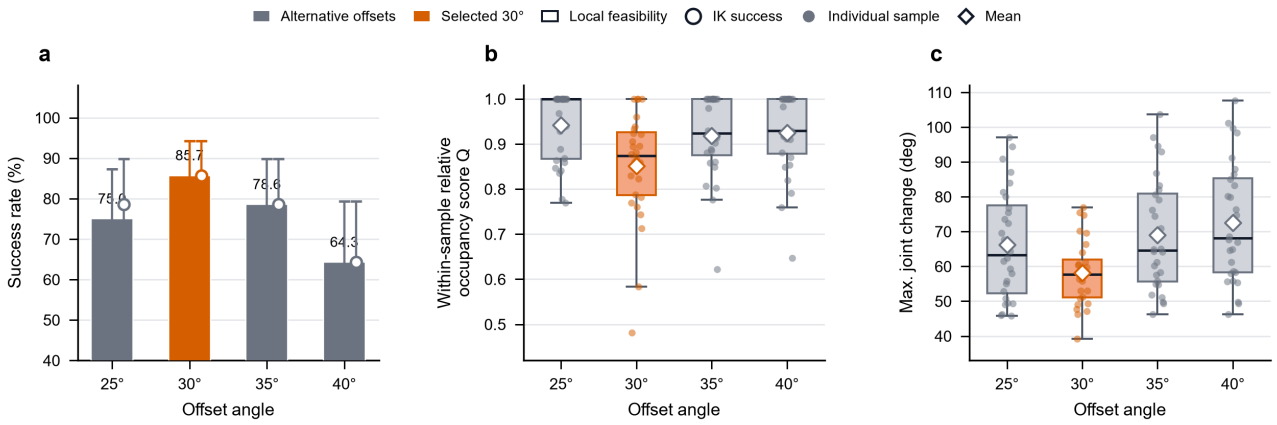


Figure 6. Offline ablation of the proposed strategy at four component-angle settings. **(a)** Samples containing at least one IK-valid candidate and local candidate-pose feasibility; **(b)** Within-sample relative occupancy scores Q; larger values denote lower occupancy relative to the other directions in the same sample, not greater physical clearance; **(c)** Maximum joint displacement. Points denote the same paired samples; boxes show interquartile ranges, centre lines indicate medians, and diamonds indicate means.

4.4. Offline comparison of three approach strategies

Supplementary Table S1 reports the selected candidates' raw occupancy counts and the proportion satisfying $N_k < 30$ for each strategy and angle.

Table 2 reports the results for the three approach strategies. Figure 7B compares their mean within-sample relative occupancy scores, the proportions of samples containing at least one IK-valid candidate, and local candidate-pose feasibility rates. Figure 7C shows maximum joint-displacement distributions across the samples, and Figure 7D shows the directions selected by each strategy. Forward-only always approached from the current TCP towards the fruit centre. Its mean within-sample relative occupancy score was 0.5526. In 78.6% of the samples at least one IK-valid candidate was found, its local feasibility rate was 71.4%, and mean maximum joint displacement was 55.7°.

Table 2. Offline local candidate-pose comparison of the three approach strategies.

Strategy	***n***	**Mean within-sample relative occupancy score *Q***	**IK-valid samples**	**Local feasibility**	**Mean maximum joint change**
Forward-only	28	0.5526	78.6%	71.4%	55.7°
Collision-only	28	1.0000	50.0%	50.0%	82.5°
Proposed	28	0.8516	85.7%	85.7%	58.1°

Collision-only obtained a mean within-sample relative occupancy score of 1.0000 by selecting the minimum-count direction in each sample. This value indicates relative rank by construction and does not establish physical clearance or collision safety. Its samples contained at least one IK-valid candidate in 50.0% of cases, its local feasibility rate was 50.0%, and mean maximum joint displacement increased

to 82.5°. Figure 7B,C shows that the lowest-occupancy direction was not always the easiest for the manipulator to reach.

The proposed method balanced within-sample occupancy rank with kinematic executability. Its mean within-sample relative occupancy score was 0.8516. Both the proportion of samples with at least one IK-valid candidate and the local feasibility rate were 85.7%, and mean maximum joint displacement was 58.1°. Local feasibility increased by 14.3 percentage points over Forward-only and by 35.7 percentage points over Collision-only. Relative to Collision-only, mean maximum joint displacement decreased by 24.4°.

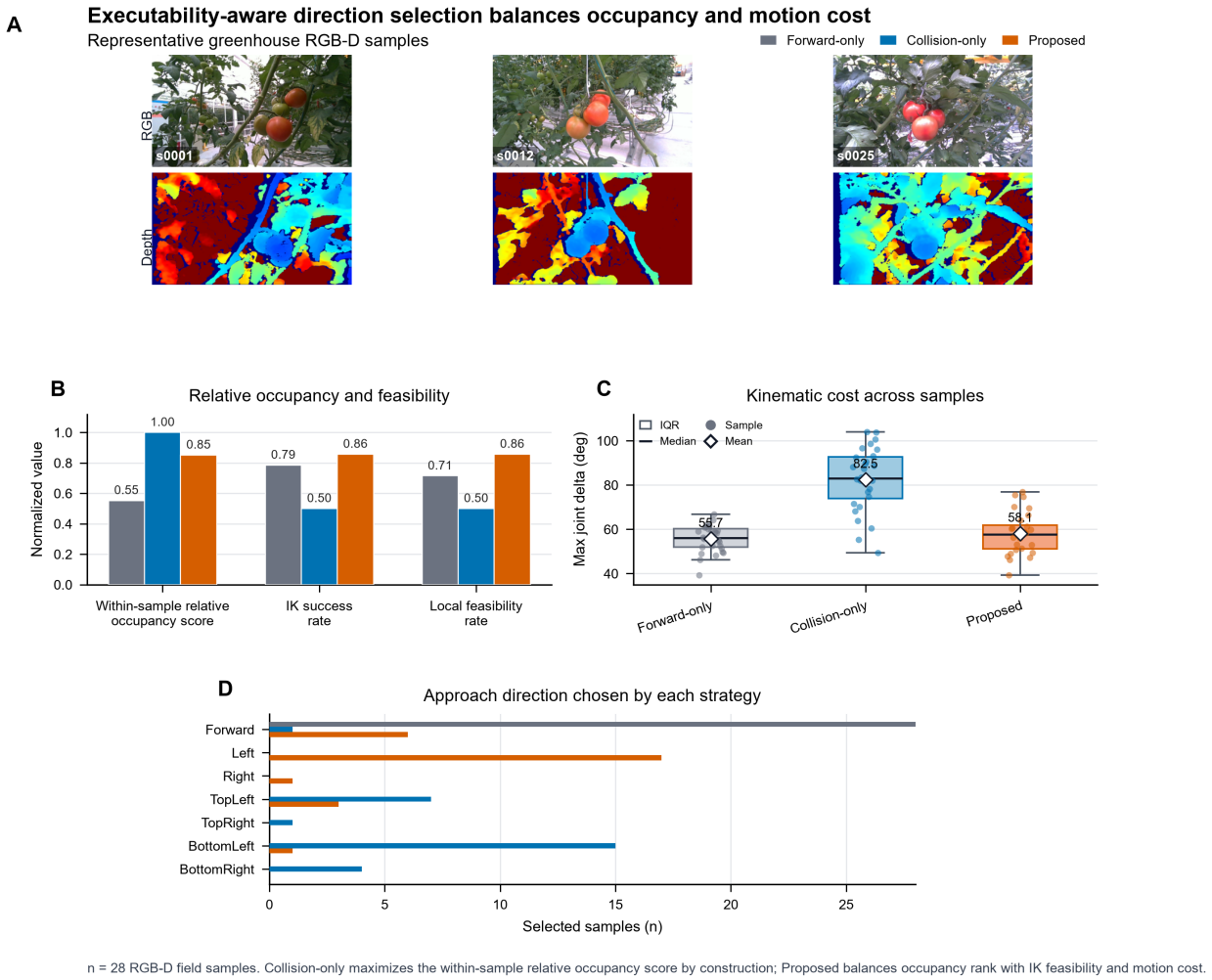


Figure 7. Offline local candidate-pose comparison of the three approach strategies. Q is a within-sample relative occupancy score and does not represent physical clearance or collision safety.

4.5. Direction selection and offline failure analysis

Occlusion type and target position were not labelled, so their effects could not be assessed.

The proposed method does not minimise the collision count for every sample. The ranking also accounts for manipulator reachability. This distinction matters in greenhouse harvesting because failed motion, near-singular approaches, and large swings increase operational risk.

Four of the 28 offline samples had no IK-valid candidate among the seven directions: sample_0009, sample_0013, sample_0020, and sample_0023. The other 24 samples contained at least one IK-valid candidate. Supplementary Table S2 lists the stage-by-stage decision records for the four failures.

4.6. Consecutive greenhouse harvesting trials

To further evaluate integrated system performance, we conducted consecutive trials on 100 ripe tomatoes at Beijing Shounong Cuihu Factory. A new RGB-D frame was acquired for each tomato; after target locking, the selected pose was held fixed until the picking attempt ended. A trial was successful when the fruit was detached intact and transferred to the collection position. The system succeeded in 82 of 100 trials, giving a harvesting success rate of 82.0% (Wilson 95% confidence interval, 73.3%–88.3%). The 18 unsuccessful trials were assigned to their primary recorded stages: six perception failures, including four fruit-centre offsets and two scene-change failures under the static-frame assumption; four planning failures with no feasible pose; and eight detachment failures in which thick or tough peduncles could not be cut or twisted free. These labels record occurrences rather than measured error magnitudes. Mean post-locking execution time was 2.7 s per fruit, measured from target locking until end-effector retraction and readiness for the next attempt; it therefore does not represent the complete perception-to-harvest cycle. Successfully harvested fruit were inspected after 24 h at room temperature. Visible skin breakage, cracking, or substantial bruising was classified as damage. Four of the 82 successfully harvested fruit were damaged, corresponding to a fruit-damage rate of 4.9%.

5. Discussion

5.1. Comparison with recent methods

The main value of the proposed method is its conversion of tomato detections into candidate actions. Conventional fixed-direction strategies treat perception and motion separately and assume that a detected target can be approached along a predefined direction. The seven-direction set provides local adjustment, while aggregate scoring avoids a purely geometric or purely kinematic bias.

Compared with the fruit-stem spatial-relation method [15], our method emphasises general point-cloud occupancy and robot execution constraints. Fruit-stem methods are agronomically interpretable but require the main stem to be visible, segmentable, and correctly associated with the target fruit. Our method does not require obstacle classes to be identified in advance. It incorporates main stems, leaves, neighbouring fruit, and supports within the same corridor evaluation. Improved BIT* [16] searches complete motion paths in continuous configuration space and optimises global distance, energy, and collision cost in a known environment. Our seven-direction module instead generates an upstream local target pose. Its output TCP pose can be passed to BIT* or another global planner, making the two methods complementary stages of a planning pipeline.

Among learning-based methods, Li and Kasaei used imitation learning to improve harvesting-view planning under occlusion [17]. AnyGrasp generates and ranks large sets of six-degree-of-freedom grasp candidates from point clouds [18]. These methods provide scene adaptation and nonlinear representations but require demonstrations, grasp labels, or large-scale pretraining. Our method uses seven discrete directions and explicit robot constraints, reducing data requirements and making the decision chain interpretable and easy to debug on site. However, fixed weights cannot adapt automatically to canopy density, robot state, or end-effector changes. A learned ranker could replace Equation (16) for candidates in the retained pool, while controller IK validity and the final joint-motion block remain rule-based [17,18].

5.2. *Learning-based extensions and limitations*

Candidate ranking currently uses fixed geometry- and robot-based weights. A future pairwise-preference or imitation model could learn from corridor occupancy, PCA anisotropy, joint displacement, condition number, and direction encoding paired with expert preferences or logged outcomes. Controller IK validity and joint-motion blocking would remain rule-based.

At the 0.20 m standoff distance, a 1° approach-direction error displaces the constructed TCP target by 3.49 mm. The resampling analysis quantifies one localisation contribution to this error budget. Absolute fruit-centre error, approach-direction error, TCP tracking error, and perception-to-contact scene displacement require synchronised reference measurements.

Validation used the current sensor and canopy setting, the AUBO C5, and the current end effector. Cross-sensor, cross-canopy, and cross-platform transfer of the scoring weights, N_k gate, and PCA threshold remains for future study. The 18 primary stage labels describe system-level outcomes rather than isolated module causes. Future trials will log stage-specific timestamps and outcome codes.

6. Conclusions

We combined learned tomato perception with executability-aware selection among seven local approach directions. Starting from ripe-tomato instance segmentation, the method estimates the three-dimensional fruit centre and generates seven candidate approach directions. Candidate poses are then ranked with fixed weights applied to local point-cloud corridor occupancy, joint displacement, singularity, and PCA geometry. This structure organises learned perception outputs into an interpretable candidate-level decision interface. It also allows a future learning-based ranker to replace the fixed weights.

In offline comparisons on 28 real greenhouse RGB-D samples, the proposed method achieved an 85.7% local candidate-pose feasibility rate.This rate exceeded those of the fixed-forward and collision-only strategies, although local feasibility does not establish complete link-level trajectory feasibility. The results indicate that tomato harvesting in dense canopies requires both local spatial traversability and manipulator execution feasibility. In 100 consecutive greenhouse trials, the complete system succeeded in 82 cases, with a mean post-locking execution time of 2.7 s per fruit. Four of the 82 successfully harvested fruits (4.9%) showed damage after 24 h. These results characterise the complete harvesting pipeline. Future work will extend validation across tomato cultivars, canopy densities, and operating conditions. It will also examine failure causes and damage to fruit and plants in greater detail.

Data availability statement

A supplementary file is provided with this article. Supplementary Tables S1 and S2 provide raw occupancy statistics and stage-by-stage decision records for the four offline IK failures. The data supporting the findings of this study comprise the 28 RGB-D samples and candidate-level planning results used in the offline comparison, model-training records, figure source data, and the summary outcomes reported for the 100 harvesting trials. These data, together with the offline-analysis and harvesting-control code and trained model weights, are available from the corresponding author upon reasonable request.

Declaration of generative AI and AI-assisted technologies

During preparation of this manuscript, the authors used OpenAI Codex to translate the Chinese manuscript and improve language and readability. The authors reviewed and edited the output and take full responsibility for the content of the manuscript.

Acknowledgments

Prof. Yun-hui Liu's pioneering research in robotic grasping and visual servoing has provided both a theoretical foundation and methodological inspiration for this work. He has long focused on enabling robots to adapt in unstructured, dynamic settings. Our work embodied Prof. Liu's core principle of "making robots work adaptively in complex real-world environments."Funding: This work was supported by the Young Elite Scientists Sponsorship Program of the Beijing High Innovation Plan (Grant No. 20250814). This work was supported by the National Agricultural Science and Technology Major Project under Grant XX202315020106.

Authors' contribution

Z.F.: conceptualization, methodology, software, validation, formal analysis, investigation, data curation, visualization, and writing—original draft; L.Y.: experiments and end-effector design; Z.R.: investigation; Y.W.: investigation. K.X.: Investigation; L.W.: investigation. T.Z.: supervision. All authors have read and agreed to the published version of the manuscript.

Conflicts of interest

The authors declare no known financial interests or personal relationships that could have influenced the work reported in this paper.

References

[1] Kootstra G, Wang X, Blok PM, Hemming J, Van Henten E. Selective harvesting robotics: current research, trends, and future directions. *Curr. Robot. Rep.* 2021, 2:95–104.

[2] Zhou H, Wang X, Au W, Kang H, Chen C. Intelligent robots for fruit harvesting: recent developments and future challenges. *Precis. Agric.* 2022, 23:1856–1907.

[3] Droukas L, Doulgeri Z, Tsakiridis NL, Triantafyllou D, Kleitsiotis I, *et al.* A survey of robotic harvesting systems and enabling technologies. *J. Intell. Robot. Syst.* 2023, 107:21.

[4] Yang Y, Han Y, Li S, Yang Y, Zhang M, *et al.* Vision based fruit recognition and positioning technology for harvesting robots. *Comput. Electron. Agric.* 2023, 213:108258.

[5] Xu Z, Jia Y, Liu X, Zhao Y, Sun G. Fast method of detecting tomatoes in a complex scene for picking robots. *IEEE Access* 2020, 8:55289–55299.

[6] Xu P, Fang N, Liu N, Lin F, Yang S, *et al.* Visual recognition of cherry tomatoes in plant factory based on improved deep instance segmentation. *Comput. Electron. Agric.* 2022, 197:106991.

[7] Li Y, Feng Q, Liu C, Xiong Z, Sun Y, *et al.* MTA-YOLACT: Multitask-aware network on fruit bunch identification for cherry tomato robotic harvesting. *Eur. J. Agron.* 2023, 146:126812.

[8] Wang C, Yang G, Huang Y, Liu Y, Zhang Y. A transformer-based mask R-CNN for tomato detection and segmentation. *J. Intell. Fuzzy Syst.* 2023, 44(5):8585–8595.

[9] Yoshida T, Kawahara T, Fukao T. Fruit recognition method for a harvesting robot with RGB-D cameras. *ROBOMECH J.* 2022, 9(1):15.

[10] Gong L, Wang W, Wang T, Liu C. Robotic harvesting of the occluded fruits with a precise shape and position reconstruction approach. *J. Field Robot.* 2022, 39(1):69–84.

[11] Xiong Y, Ge Y, Grimstad L, From PJ. An autonomous strawberry-harvesting robot: design, development, integration, and field evaluation. *J. Field Robot.* 2020, 37(2):202–224.

[12] Rong J, Hu L, Zhou H, Dai G, Yuan T, *et al.* A selective harvesting robot for cherry tomatoes: design, development, field evaluation analysis. *J. Field Robot.* 2024, 41(8):2564–2582.

[13] Qi Z, Hua W, Zhang Z, Deng X, Yuan T, *et al.* A novel method for tomato stem diameter measurement based on improved YOLOv8-seg and RGB-D data. *Comput. Electron. Agric.* 2024, 226:109387.

[14] Dai N, Fang J, Yuan J, Liu X. 3MSP2: sequential picking planning for multi-fruit congregated tomato harvesting in multi-clusters environment based on multi-views. *Comput. Electron. Agric.* 2024, 225:109303.

[15] Chen J, Yu T, Lou M, Xu X, Jiang H, *et al.* Tomato harvesting pose estimation based on fruit-stem spatial relationship. *Comput. Electron. Agric.* 2026, 250:111913.

[16] Ma P, Zhu A, Chen Y, Tu Y, Mao H, *et al.* Multi objective motion planning of fruit harvesting manipulator based on improved BIT* algorithm. *Comput. Electron. Agric.* 2024, 227:109567.

[17] Li L, Kasaei H. Enhanced view planning for robotic harvesting: tackling occlusions with imitation learning. In *Proceedings of 2025 IEEE International Conference on Robotics and Automation (ICRA)*, Vienna, Austria, June 1–6, 2025 pp. 13146–13152.

[18] Fang HS, Wang C, Fang H, Gou M, Liu J, *et al.* AnyGrasp: Robust and efficient grasp perception in spatial and temporal domains. *IEEE Trans. Robot.* 2023, 39(5):3929–3945.

[19] Wang Y, Liu D, Zhao H, Li Y, Song W, *et al.* Rapid citrus harvesting motion planning with pre-harvesting point and quad-tree. *Comput. Electron. Agric.* 2022, 202:107348.

[20] Kurtser P, Edan Y. Planning the sequence of tasks for harvesting robots. *Robot. Auton. Syst.* 2020, 131:103591.