The following publication H. Chen et al., "PSO-Based Optimal Coverage Path Planning for Surface Defect Inspection of 3C Components With a Robotic Line Scanner," in IEEE Transactions on Instrumentation and Measurement, vol. 74, pp. 1-12, 2025, Art no. 7505012 is available at https://doi.org/10.1109/TIM.2025.3552466.

PSO-Based Optimal Coverage Path Planning for Surface Defect Inspection of 3C Components with a Robotic Line Scanner

Hongpeng Chen, *Graduate Student Member, IEEE,* Shengzeng Huo, Muhammad Muddassir, *Member, IEEE,* Hoi-Yin Lee, *Member, IEEE,* Yuli Liu, Junxi Li, Anqing Duan, *Member, IEEE,* Pai Zheng, *Senior Member, IEEE,* and David Navarro-Alarcon, *Senior Member, IEEE*

***Abstract*—The automatic inspection of surface defects is an essential task for quality control in the computers, communications, and consumer electronics (3C) industry. Traditional inspection mechanisms (viz. line-scan sensors) have a limited field of view, thus, prompting the necessity for a multifaceted robotic inspection system capable of comprehensive scanning. Optimally selecting the robot's viewpoints and planning a path is regarded as coverage path planning (CPP), a problem that enables inspecting the object's complete surface while reducing the scanning time and avoiding misdetection of defects. In this paper, we present a new approach for robotic line scanners to detect surface defects of 3C free-form objects automatically. A two-stage region segmentation method defines the local scanning based on the random sample consensus (RANSAC) and K-means clustering to improve the inspection coverage. The proposed method also consists of an adaptive region-of-interest (ROI) algorithm to define the local scanning paths. Besides, a Particle Swarm Optimization (PSO)-based method is used for global inspection path generation to minimize the inspection time. The developed method is validated by simulation-based and experimental studies on various free-form workpieces, and its performance is compared with that of two state-of-the-art solutions. The reported results demonstrate the feasibility and effectiveness of our proposed method.**

***Index Terms*—Coverage path planning (CPP), Line-scan sensor, Surface inspection, Robotic inspection, 3C components.**

I. Introduction

DEFECT inspection is essential to quality control, process monitoring, and non-destructive testing (NDT) in the manufacturing industry [1], [2]. Specifically, manufacturing processes in the 3C industry are highly sophisticated and demand detailed and accurate defect inspection. Traditional defect inspection approaches typically rely on visual inspection of an intermediate/finished product by a quality control or quality check inspector. This sole dependence on human workers is a problem for regions and countries with a shortage of manpower [3]–[5]. Furthermore, human-based inspection is inherently subjective and, hence, prone to errors. To address these issues, various researchers have reported the automatic surface inspection system for free-form components [6], [7].

Recently, automatic detection systems equipped with an industrial-grade line scanner, depth camera, and robotic manipulator have been developed to offer effective and rapid non-contact measurement [8]–[10]. During the defect inspection task, the robotic inspection system exhaustively scans the surface of the target workpiece from different viewpoints. Planning an inspection path can be considered as the coverage path planning (CPP) problem [11]. An optimal CPP strategy for automated inspection includes: (1) selecting viewpoints for exhaustive surface measurement, (2) optimizing the sequence of these viewpoints for efficiency and kinematic feasibility, and (3) devising paths to effectively navigate between viewpoints. Critical considerations also encompass complete surface coverage and minimized inspection cycle-time [12]. The existing CPP methods can be categorized into two coarse categories: two-dimensional and three-dimensional methods.

Various researchers reported two-dimensional (2D) CPP for mobile robots in floor cleaning, bridge crack monitoring, and weed mowing tasks [13], [14]. Hung M. La et al. [15] proposed an autonomous robotic system for efficient bridge deck inspection using boustrophedon decomposition to address the CPP problem. Lim et al. [16] developed a genetic algorithm-based system for bridge crack inspection, optimizing the CPP to reduce turns and detection time. Danial Pour Arab et al. [17] introduced a CPP algorithm for agricultural fields, employing tree exploration to identify solutions and similarity comparisons to minimize overlaps, path length, and travel time. As for the three-dimensional (3D) CPP problem, most relative methods have to plan the paths from partial or occluded 3D maps. A CPP method for 3D reconstruction based on building information modeling utilized a robotic arm and lifting mechanism for wall painting, featuring a two-stage framework with a global planner for waypoint sequencing and a local planner for mobile base positioning, improving efficiency [18]. Zbiss. K et al. [19] introduced a path-planning method for collaborative robotic car painting, leveraging computational

This work is supported in part by the Research Grants Council of Hong Kong under Grant 15212721, and in part by Shanghai Microintelligence Technology Co. Ltd. Corresponding author: David Navarro-Alarcon

H. Chen, S. Huo, H.-Y. Lee, J. Li, P. Zheng and D. Navarro-Alarcon are with the Faculty of Engineering, The Hong Kong Polytechnic University (PolyU), KLN, Hong Kong. (e-mail: hongpeng0925.chen@connect.polyu.hk, kyle-sz.huo@connect.polyu.hk, hyin-lee@polyu.edu.hk, 21058587g@connect.polyu.hk, pai.zheng@polyu.edu.hk, dnavar@polyu.edu.hk)

A. Duan is with the Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, United Arab Emirates. (e-mail: anqing.duan@mbzuai.ac.ae)

M. Muddassir is with the Faculty of Construction and Environment, The Hong Kong Polytechnic University (PolyU), KLN, Hong Kong. (e-mail: mmudda@polyu.edu.hk)

Y. Liu is with Department of Ocean Science and Technology at the University of Macau's Faculty of Science and Technology, Macau. (e-mail:mc45259@um.edu.mo)

geometry, convex optimization, and algorithms like Morse cellular decomposition and boustrophedon for collision-free trajectories. For UAV-based bridge inspection, a CPP method integrated genetic and A* algorithms to identify barrier-free, near-optimal paths [20].

Despite numerous recent developments, CPP for free-form surface inspection remains an open research issue. There are very few CPP solutions for line scanning robotic systems [21]. Compared to area scan sensors, a line scan sensor is more suitable for defect inspection in industrial/manufacturing applications due to higher spatial resolution and lower production costs [22], [23]. Unlike conventional area cameras or optical sensors that operate at discrete positions, a line scanner employs a single beam of scanning light to detect 3D objects, requiring continuous movement along a coverage path via a robotic manipulator. This characteristic renders many traditional CPP methods ineffective. For instance, Li et al. [24] proposed a robust CPP method for aerospace structures utilizing feature graph construction based on the Voronoi diagram. However, this approach often neglects geometric variations at surface boundaries, resulting in coverage gaps and reduced inspection precision. Similarly, Liu et al. [9] developed an enhanced rapidly exploring random tree (RRT*) method that incorporates inspection errors and optimizes the number of viewpoints for improved precision. However, the generated viewpoints are not well-suited for continuous scanning, potentially leading to inconsistent inspection quality across varying geometries. Moreover, many existing CPP methods fail to account for geometric differences among individual workpieces, adversely affecting detection performance, particularly at surface boundaries, where inspection clarity is critical. Glorieux et al. [12] introduced a waypoint sampling strategy to minimize inspection time for sheet metal parts, but the method demonstrated limited coverage efficiency for complex 3D free-form surfaces. Huo et al. [8] implemented a nearest neighbor search algorithm to generate near-optimal scanning paths for convex free-form surfaces. Although designed for automatic line-scanning systems, this method struggles to balance high detection coverage with inspection efficiency. Therefore, developing a novel CPP method for the automatic line scanning system becomes imperative and advantageous.

This paper aims to overcome the limitations of existing CPP methods for surface defect inspection. The challenge of providing minimal viewpoints to ensure full coverage and precise defect inspection has been addressed in this work. The method leverages a 6-DOF robotic manipulator equipped with a line scanner for comprehensive surface coverage and a depth sensor for accurate workpiece localization. It integrates two key components: local path definition to ensure precise defect detection and global optimization to minimize scanning time, thereby achieving a balance between efficiency and accuracy. Tailored to the unique requirements of surface inspection, the proposed method generates a complete traversal path for all viewpoints, enabling the robotic system to efficiently and accurately identify surface defects. Furthermore, the approach incorporates optimization-based techniques, segmentation methods, and complex geometric representations to address the challenges of inspecting intricate free-form surfaces. These contributions collectively provide a feasible and effective solution for advancing quality monitoring in industrial manufacturing, particularly within the 3C industry.

The main contribution of this paper includes:

(1) A new region segmentation method and an adaptive region-of-interest (ROI) algorithm to define the local scanning paths for free-form surfaces with complex and high imaging quality assessment requirements.
(2) A Particle Swarm Optimization (PSO)-based global inspection path generation method to minimize the inspection time.
(3) Detailed simulations, experiments, and comparisons to validate the proposed method.

The rest of this article is organized as follows. Section II describes the path planning problem for 3C component surface detection. Section III presented the proposed CPP approach in detail. Section IV shows the specific simulations, experiments, and comparisons on 3C components to validate the method's feasibility. Finally, Section V concludes this article and discusses the limitations and future direction.

II. Coverage Path Planning for Inspection

The CPP problem can be divided into two subproblems: 1) the local path definition is to generate view regions and partial scanning paths to meet the precise scanning and full coverage for 3C free-form workpieces. 2) global path planning aims to find an optimal or near-optimal sequence of all local paths [25].

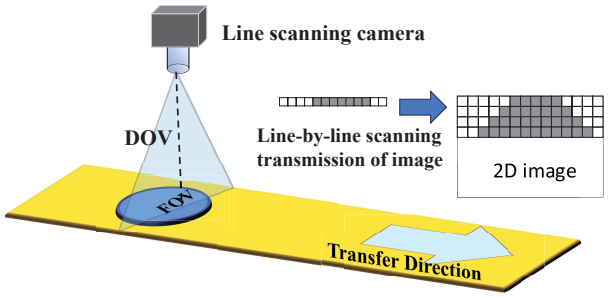

Fig. 1. Conceptual representation of line scanning sensor.

The first sub-problem involves determining the position and orientation of viewpoint pairs at both ends of each local path. Since the line-scan camera captures one pixel line at a time, relative motion perpendicular to the pixel line is essential for 2D image acquisition during defect inspection (see Fig. 1). In this robotic scanning system, the camera moves along the stationary object with its depth of view (DOV) aligned perpendicular to the scanned region to ensure image quality. Thus, the scanned area must remain as flat as possible, even for workpieces with complex geometries (see Fig. 2). Each local path consists of two viewpoints, and the robotic end-effector scans between them, maintaining consistent orientation to preserve image quality. Critical factors such as field of view (FOV) and DOV also influence this sub-problem [9].

The global path planning problem involves finding the sequence and path connecting the selected viewpoints to minimize travel costs. This generated coverage path must reach

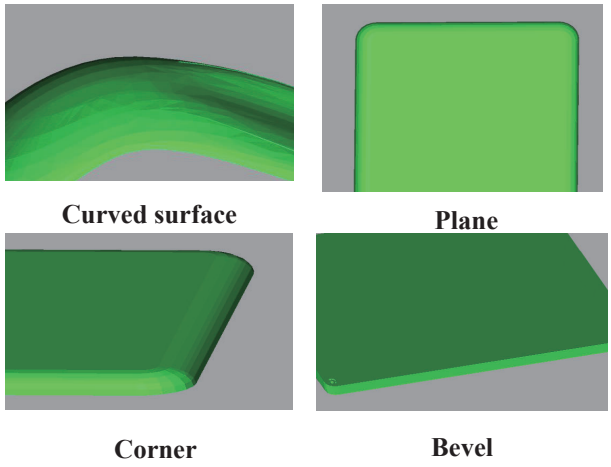

Fig. 2. Geometric features of 3C workpieces.

all local paths with the shortest connection path. In other words, the objective is to find the minimum kinematic feasible path for the robot manipulator to target the scanning sensor at each viewpoint precisely through all local paths without colliding with any obstacles in the workspace.

The proposed method aims to generate an efficient and feasible coverage path that minimizes inspection time while ensuring comprehensive traversal of all local paths. It accounts for diverse measurement directions to maintain high detection precision. To achieve accurate defect imaging, scanning parameters are automatically optimized, reducing reliance on manual adjustments and operator expertise.

III. Methodology

A CPP generation and optimization approach is presented based on the robotics line scanning system (see Fig. 3). This includes i) a new hybrid region segmentation method based on the random sample consensus (RANSAC) and K-means clustering method, ii) an adaptive ROI method to define the local scanning paths, and iii) one PSO-based global optimization approach for the minimum inspection time. The detailed flowchart is shown in Fig.4 about the working process of the CPP method.

The 3D model is converted into a point cloud to extract geometric features. Unlike mesh (STL) or other representations, the point cloud offers a detailed depiction of objects and simplifies the analysis of complex structures, including position, normal, and curvature. [26]. The sampling procedure is based on selecting a series of points randomly and uniformly from the model to form a point cloud that can be used to segment and process all surfaces of the workpiece. The acquired point cloud O consists of points $p_i = [x_i, y_i, z_i], i = 1, 2, ..., m$ (m is the total sampling number of O), which preserves the geometric information of all faces.

A. Hybrid region segmentation based on RANSAC and K-means clustering

The image acquisition characteristics of line-scan cameras require flat scanning areas for optimal image quality. To achieve this, we propose a novel hybrid region segmentation approach for 3C component surfaces, combining RANSAC for plane identification with an enhanced K-means clustering algorithm. This method effectively segments complex surfaces into flatter regions, ensuring high-quality image acquisition.

Here, we use RANSAC to partition O first. Given different hypothesis geometrical models, RANSAC can identify planes, spheres, cylinders, and cones [27]. Since the flat regions are required for precise line scanning, RANSAC utilized the equation of a plane as a feature model in the proposed system. It selects N sample points of O and estimates the plane model parameters by those sample points. The position of a point is selected as an inlier if the distance between the point and plane is less than the fixed thresholds and the shape that contains the greatest number of outlier points could be split and extracted after multiple iterations. The plane model can be represented as

$$aX + bY + cZ + d = 0 \tag{1}$$

where $[a, b, c, d]^T$ is the plane model parameter, and $[X, Y, Z]^T$ denotes any point in the 3D coordinates.

This method can extract a nearly planar point cloud region C_0 when the best plane model has been identified. RANSAC does not require complex optimization or high memory resources so that we can obtain C_0 rapidly. However, this approach cannot segment the remaining point cloud O^r with the size η^r because O^r consists of bevels, curved surfaces, and other complex geometrical information.

Traditional K-means clustering methods regarded region segmentation as a clustering analysis problem of geometric features of the surface. They applied the position and surface normals of the point cloud for segmentation, which are not appropriate for workpieces with significant variations in curvature or many bevels and corners [28], [29]. Therefore, some different factors should be considered when describing the features of the object. The enhanced K-means clustering is proposed in this paper to process O^r. In the standard K-means method, the number of clusters N dramatically affects the performance of this method, and many trials are required to find a near-optimal N in some classical methods [30]. In this developed method, we apply not only the corresponding surface normals $n_i^r = [n_{ix}^r, n_{iy}^r, n_{iz}^r]$ of the points in O^r but also the Gaussian curvature K_i^r and Mean curvature H_i^r of each point p_i^r in O^r as the inputs of the enhance K-means clustering. Besides, a feasible weighting factor ω among n_i^r, K_i^r, and H_i^r is determined through many manual experiments. K_i^r is the product of the principal curvatures of p_i^r, and it neutralizes the maximum and minimum curvatures. A positive Gaussian curvature value means the surface is locally either a summit or a valley, while a negative value illustrates the surface locally consists of saddle points. And zero Gaussian curvature indicates the surface is flat in at least one direction, like a plane or cylinder [31]. In mathematics, the mean curvature of a surface presents the curvature of an inset surface in Euclidean space or other ambient spaces. The curvature of the point can be represented by $c_i^r = [K_i^r, H_i^r]$. With adding these two parameters in this enhanced K-means method, the clustering quality can be improved than before, so the geometric feature

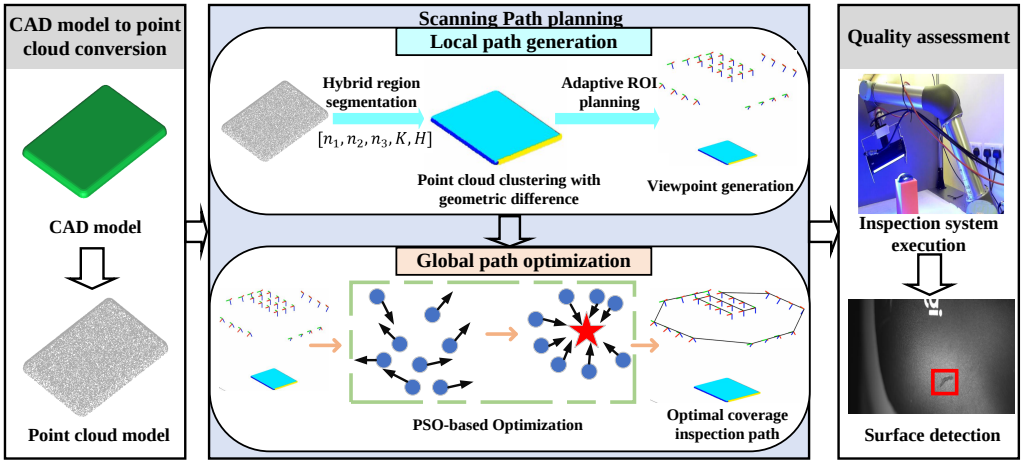

Fig. 3. Framework of the proposed method, including CAD-to-point cloud conversion, surface detection, path planning, and PSO-based optimization for inspection execution.

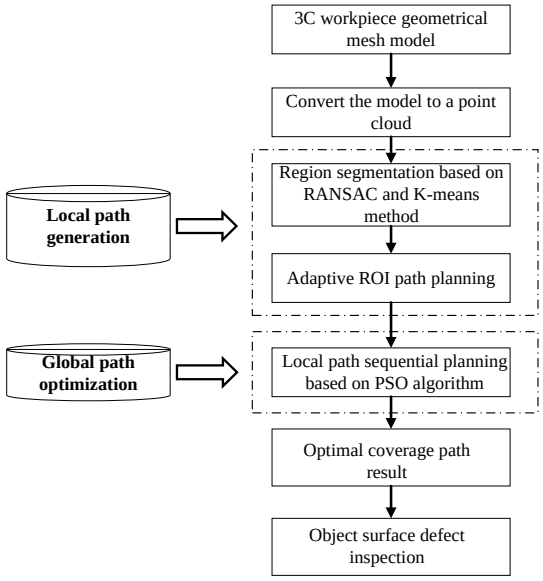

Fig. 4. Flowchart showing the working process of the proposed CPP method.

of the point of O^r is presented as $I_i^r = [n_i^r, c_i^r]$. Besides, we present a method to automatically adjust N since N affects the result of the classification, and the traditional techniques set one fixed N, whose drawback is its poor flexibility. The algorithm depends on a two-looped 1D search, with the inner loop for similarity comparison and the outer loop for iterating N. The iteration can end when the largest intra-class difference is smaller than a threshold T. The entire procedure of this enhanced K-means method is illustrated in Algorithm 1.

For the outer loop, we represent the feature vectors of the N-cluster set as

$$Q_j = [q_n, q_c] \quad q_n = [q_1, q_2, q_3] \quad q_c = [q_4, q_5] \tag{2}$$

Q_j is one 5-dimensional vector ($j = 1, 2..., N$). All of them can be initialized with a random value. Afterward, the procedure goes into the inner loop, composed of two steps: 1) similarity comparison and 2) updating. In the first step, cosine similarity is used in this proposed method for assessing the similarity between I_i^r and Q_j, which is considered as a measure of similarity between two sequences of numbers in data analysis [32]. The similarity α_{ij} is described in detail as follows:

$$\alpha_{ij} = \omega_1 \cos\left(\frac{n_i^r \cdot q_n}{|n_i^r| \cdot |q_n|}\right) + \omega_2 \cos\left(\frac{c_i^r \cdot q_c}{|c_i^r| \cdot |q_c|}\right) \tag{3}$$

$$\omega_1 + \omega_2 = 1, \quad \omega_1, \omega_2 \geq 0 \tag{4}$$

where ω_1 and ω_2 are the weighting factors for α_{ij}, and they are set as 0.6 and 0.4 respectively in this method based on many trials.

Then, this method should find the cluster C_j with the smallest α_{ij} and extract the corresponding p_i^r and I_i^r to it. The next step is to determine whether the classification has met the termination condition. For each cluster C_j, the termination parameter λ_j is calculated from the maximum intra-class difference D_j as:

$$\lambda_j = \begin{cases} 0, D_j > T \\ 1, else \end{cases} \quad ; D_j = \max_i \alpha_{ij} \tag{5}$$

β_t represents the sum of λ_j from every region C_j at this iteration t .If $\beta_t = N$, the current segmentation is satisfactory and the algorithm can finish iteration. Otherwise, the procedure continues. In this stage, the search direction should be considered since the method includes two loops, the inner one that compares similarity and clusters concerning N and the outer one that increases the value of N gradually. The change relies on the performance of β_t. If the performance deteriorates at the iteration step t (i.e. β_t is smaller than β_{t-1}), the inner loop must stop immediately and a new outer loop starts with $N \leftarrow N + 1$ because the current N is not ideal. If the performance is better(i.e. β_t is larger than β_{t-1}), the search within the inner loop continues.

Before switching to the next inner iteration, all feature vector $Q_j = [q_n, q_c]$ are updated to improve the representation level:

$$q_n = \frac{\frac{1}{\eta_j}\sum_{i=1}^{\eta_j} n_{ij}}{\left\|\frac{1}{\eta_j}\sum_{i=1}^{\eta_j} n_{ij}\right\|} \qquad q_c = \frac{\frac{1}{\eta_j}\sum_{i=1}^{\eta_j} c_{ij}}{\left\|\frac{1}{\eta_j}\sum_{i=1}^{\eta_j} c_{ij}\right\|} \tag{6}$$

where n_{ij}, c_{ij} and η_j are i-th normal feature vector in C_j, curvature feature vector in C_j and the size of the C_j separately.

The proposed algorithm only takes the limited features of the region C_j into consideration, which can lead to a high sparsity of the clustered points within the same region. Therefore, Euclidean cluster extraction is implemented as a post-processing step to verify if it is necessary to subdivide the region C_j into two new regions according to the location of the points in it.

The similarity threshold T should be selected before region segmentation. If T is large, the segmentation process needs more computation time to allow finer distinctions between clusters, which may result in better segmentation for complex geometries but at a higher computational cost. On the contrary, a smaller value of T reduces the computational overhead by grouping more features into a single cluster but may compromise segmentation accuracy. Consequently, selecting this component must balance the segmentation accuracy and calculation efficiency. $[0.57, 0.69]$ is an optimal value range for T, found by many hit and trials.

As for weight factors ω_1 and ω_2, these factors are used in the cosine similarity calculation to evaluate the similarity between feature vectors during clustering. ω_1 represents the weight assigned to the surface normals n_r , and ω_2 represents the weight for curvature features (K_r and H_r). A higher ω_1 emphasizes geometric alignment, which is crucial for planar surfaces.If ω_2 is large, the sensitivity to curvature changes is improved, which is essential for effectively handling complex surfaces. The value of ω_1 is generally chosen to be greater than ω_2. This is primarily because surface normals (n_r) play a more significant role in defining geometric features and determining region segmentation. Multiple experiments have demonstrated that appropriately increasing ω_1 (e.g., setting it within the range of $0.55 \leq \omega_1 \leq 0.65$) can significantly enhance segmentation accuracy. Conversely, assigning a lower value to ω_2 (e.g., within the range of $0.35 \leq \omega_2 \leq 0.45$) effectively controls computational complexity while maintaining reasonable sensitivity to curvature variations.

Algorithm 1: The enhanced K-means Region Segmentation

Input: T, O_r
Output: $C_j, j = 1, 2..., N$
while $\beta_t < N$ **do**
  Initialize Q_j randomly, $j = 1, 2..., N$;
  while $\beta_t \geqslant \beta_{t+1}$ **do**
    for $i = 1 : \eta^r$ **do**
      Compute similarity $\alpha_{ij} \leftarrow (3)$;
      $j_{argmin} \leftarrow \arg\min \alpha_{ij}$, take $p_i^r \rightarrow C_j$;
    end
    Calculate: $\beta_t \leftarrow (5)$;
    Update: $Q_j \leftarrow (6)$;
  end
  $N \leftarrow N + 1$;
end

B. Adaptive ROI Based Path Planning

Local paths are generated according to the proposed planning method, which takes the segmented region C_j as input. Due to the synchronization of the scanning inspection of the line camera and the robot's motion, every viewpoint in these local paths should be produced through a feasible method for accurate detection, and all local paths are required to cover the whole region C_j of the workpiece. Hence, this part presents an adaptive ROI method for generating local paths that aim to adapt scan paths and viewpoints to the various shapes of objects. The adaptive ROI selection dynamically adjusts the region of interest (ROI) based on the geometric features of the segmented regions (e.g., surface normals). For each segmented region C_j, the system evaluates the longest dimension of the region and divides it into subregions W_{jf} along this direction. This ensures efficient scanning paths and consistent DOV alignment. This dynamic mechanism ensures full coverage of the surface while minimizing redundancy and scanning errors.

Given that the scanning sensor acquires images in horizontal lines, the area it covers during linear movement can be visualized as a three-dimensional rectangular prism. This space encompasses the DOV represented by V_D, the FOV depicted by V_F, and the vector of linear motion V_L (see Fig. 5). Besides, the key of this approach is to determine the position $\mu = [x, y, z]$ and pose $i = [\vec{d}, \vec{l}]$ of the viewpoints (v^p, v^{p*}) at both ends of a local path $G_t, t = 1, 2, ..., U$. The pose i is described by the direction $\vec{d}$ of V_D and the direction $\vec{l}$ of V_L.

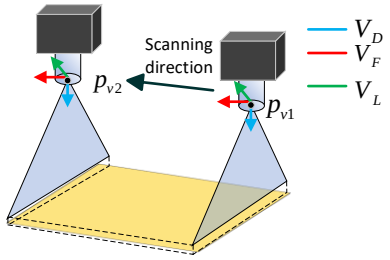

Fig. 5. Cuboid coverage model illustrating the line scanner's DOV, FOV, and scanning motion vector during linear motion.

To make the geometric scanning model effective and keep the detection accuracy of this system, our algorithm further segments every C_j into 3 sub-regions $W_{jf}, f = 1, 2, 3$. Due to the irregular shape of each C_j, we stipulate that the C_j is divided into 3 sub-region W_{jf} evenly following the direction $\vec{k}$ of the longest length of each C_j and the scanning motion is also along $\vec{k}$ for every area ($\vec{l} = \vec{k}$). In addition, we define that $\vec{d}$ is the reverse direction of the surface normal $\vec{w_{jf}}$ of this W_{jf} ($\vec{d} = -\vec{w_{jf}}$).

Thus, the corresponding μ_1, μ_2 are located on :

$$\mu = \tau - \vec{w_{jf}} \cdot |V_D| \tag{7}$$

The center of the sub-region W_{jf} is regarded as $c_{jf} = [c_x, c_y, c_z]$, and the intersections τ_1, τ_2 of the W_{jf} 's edge and the line $\vec{k} \cdot c_{jf}$ are deemed as the inspection points of viewpoints v^p, v^{p*} at both ends of a local path G_t on this sub-region surface. $|V_D|$ is the magnitude of V_D.

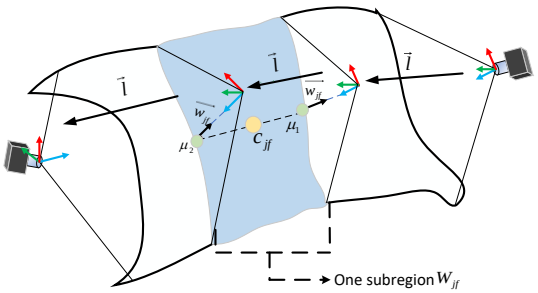

Fig. 6. Adaptive ROI-based region segmentation and linear path planning. The segmented region (C_j) is divided into subregions (W_{jf}), with the scanning motion aligned along the region's longest axis ($\vec{l} = \vec{k}$). The DOV is set opposite to the surface normals ($\vec{w}$), ensuring consistent alignment and full coverage for effective defect detection.

C. PSO-based global path optimization

Based on the local path definition in the previous step, the next challenge is to sequence these paths optimally to scan the entire surface of the free-form workpiece efficiently. The goal is to minimize the total movement time of the robot, assuming the sensor moves at a constant speed during the inspection. Practical demands dictate that the robotic arm must pass through all predetermined viewpoints to accomplish the scanning inspection. This sequencing conundrum is akin to the Traveling Salesman Problem (TSP), where we seek the most time-efficient route. The necessity of solving this problem lies in the efficiency gains for the inspection process, ensuring the robotic manipulator operates in the most time-effective manner without compromising the inspection quality [33]. The TSP is one integrated optimization problem and nondeterministic polynomial time (NP)-hard. The problem of global path planning can be formulated

$$\min \left\{ \sum_{t=1}^{U} \sum_{s=1}^{U-1} T_t^{scanning} + T_s^{across} \right\} \tag{8}$$

where $T_t^{scanning}$ is the cost time of passing every local path G_t, T_s^{across} means the cost time from G_t to G_{t+1} and U represents the total number of local paths. The cost time in the context of the robot manipulator's end-effector is determined by the straight-line distance between two viewpoints, considering the constant speed of movement. In contrast to the general TSP, our scenario requires sequential traversal of adjacent viewpoints within the same local path to ensure optimal inspection performance. This constraint is imposed due to the limitations of region segmentation and the necessity for adaptive ROI local path definition. The limitation can be summarized as

$$T_t^{scanning}(G_t) = \begin{cases} T(v_t^p \rightarrow v_t^{p*}) \\ T(v_t^{p*} \rightarrow v_t^p) \end{cases} \tag{9}$$

$$T_s^a cross(G_t, G_{t+1}) = \begin{cases} T(v_t^p \rightarrow v_{t+1}^p) \\ T(v_t^p \rightarrow v_{t+1}^{p*}) \\ T(v_t^{p*} \rightarrow v_{t+1}^{p*}) \\ T(v_t^{p*} \rightarrow v_{t+1}^p) \end{cases} \tag{10}$$

The prior studies on this problem include branch and bound linear programming, and dynamic programming methods [34], [35]. However, as the quantity of targets expands, the computation of a feasible path becomes exponentially more difficult, and obtaining the global optimal solution becomes more challenging. In the proposed method, the PSO-based method is employed to tackle the TSP due to its inherent adaptability in finding solutions to this issue. To prevent the PSO algorithm from converging to local optima, the inertia weight can be dynamically adjusted, and the initial particle positions can be diversified to balance the global exploration and local exploitation capabilities. After selecting the shortest path, the optimal general path sequence can be acquired in this step. In PSO [36], a swarm of particles are used to describe the possible solutions. Every particle ξ is related to two vectors in D-dimension space, i.e., the velocity vector $\boldsymbol{V_\xi} = [v_\xi^1, v_\xi^2, ..., v_\xi^D]$ and the position vector $\boldsymbol{X_\xi} = [x_\xi^1, x_\xi^2, ..., x_\xi^D]$. Both of them are initialized by random vectors. During the PSO process, the velocity and position of particle ξ on dimension d are updated as [37]:

$$\begin{aligned} V_\xi^d = \omega V_\xi^d &+ c_1 \operatorname{rand}_1^d \left(pBest_\xi - X_\xi^d\right) \\ &+ c_2 \operatorname{rand}_2^d \left(gBest - X_\xi^d\right) \end{aligned} \tag{11}$$

$$X_\xi^d = X_\xi^d + V_\xi^d \tag{12}$$

where ω represents the inertia weight (it is set as 1), and c_1 and c_2 are random numbers within [0,1]. $pBest_\xi$ is the position with the best fitness value for the ξth particle and $gBest$ is the best position in the global. The main steps of PSO are:

(1) Initialize all particles, including their velocity and position.
(2) Establish the fitness function and calculate the fitness value of each particle,
(3) Update the $pBest_\xi$ and $gBest$.
(4) Update the velocity and position of each particle according to (10) and (11).
(5) Increase the number of iterations, Go to step 3 and repeat until the termination condition.

D. Suitability of the proposed method

Due to the uniqueness of the line-scan camera, it is necessary to propose a new CPP method for a robotics line scanner. In our method, a new hybrid region segmentation method can provide precise local scanning paths. It is hard to obtain accurate region segmentation results using only the RANSAC method or k-means clustering. The RANSAC method can detect a region with planar geometry. It can also remove some points with

minimum curvature from the entire point cloud, enhancing the computation speed of the whole procedure [38]. Furthermore, it can effectively remove outliers, thereby improving the accuracy of the subsequent K-means clustering process. The conventional K-means clustering methods used the location and surface normals of the point cloud for region segmentation, which are not valuable for objects with large variations in curvature or some corners. In the enhanced K-means method, we use the surface normals and Gaussian curvature and Mean curvature, both of which can evaluate the geometric features comprehensively. The PSO-based method is used to solve the global path sequence optimization. However, the constraints and requirements are different from the previous work due to the limitations of region segmentation and the necessity for local path definition.

As for state-of-the-art region segmentation methods, such as deep learning (DL), the creation and annotation of high-quality training data can be prohibitively expensive, making it challenging for small and medium-sized enterprises to implement such solutions. In surface defect detection for 3C products, sufficient labeled datasets are often limited. Besides, High-performance DL-based methods typically rely on specialized hardware, which may not be available in industrial environments where standard equipment is the norm. Furthermore, in real-time detection scenarios, the inference latency of ML models may fail to meet stringent real-time requirements [39]. While pre-trained models may perform well in specific environments, they often require retraining or fine-tuning when applied to different surface materials or complex geometries. Additionally, maintaining complex DL-based systems in industrial settings can impose additional burdens, such as model updates and data management, further complicating their deployment and scalability [40]. In contrast, our proposed method circumvents these limitations by leveraging computationally efficient traditional techniques such as RANSAC and enhanced K-means clustering for region segmentation and PSO for path optimization. These methods do not require labeled datasets, extensive training, or high-end hardware, making them more accessible to resource-constrained environments. Furthermore, the direct processing of geometric features ensures robust adaptability to varying surface geometries without the need for retraining, while the computational efficiency of the method supports real-time operation.

IV. Case Study

To illustrate the performance of the proposed method, we provide three case studies for simulation tests (Case 1: a camera lens, Case 2: a computer fan, Case 5: convex free-form surface) and two case studies for experimental evaluation (Case 3: a tablet back cover, Case 4: upper part of computer mouse) on 3C component surface inspection. A state-of-the-art CPP method is also used for comparison with the developed method in Section IV-C.

A. Case study setup

Fig. 7 shows the experimental setup for evaluating the proposed method. A custom-made end-effector housed the defect inspection system consisting of a line scanning sensor (Hikvision MV-CL041-70GM camera) and a uniform line illumination source (TSD-LSH230200-B from TSD company). The Intel RealSense L515 LiDAR camera was mounted on the top of the workspace to capture the real-time stream of point clouds. The pose of the workpiece was estimated using the point clouds from LiDAR. An analog control box with a high-power strobe ensures an adjustable and stable voltage for the light source. The system consisted of a UR5 manipulator from Universal Robots to manipulate the end-effector in order to scan the workpiece automatically. The entire automated line scanning framework is based on ROSon Linux PC, which can simultaneously monitor the sensors (line scanner, depth sensor) and control the actuator (manipulator).

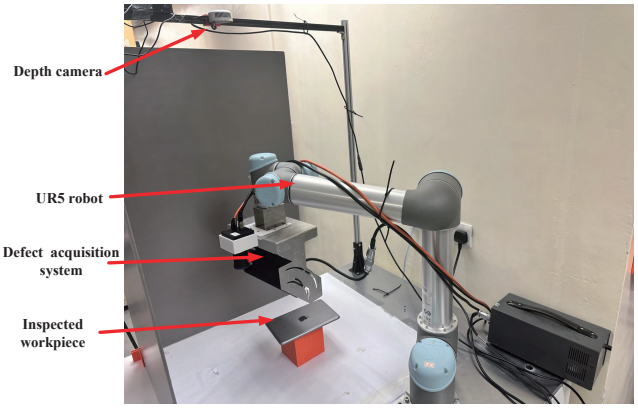

Fig. 7. Experimental setup of the automated line scanning system.

TABLE I
The Parameters Used in the Case Study

Parameter	Value
V_D(mm)	300
V_F(mm)	70
Sampling frequency(Hz)	10000
Image resolution	3000*680

The line velocity and acceleration of the manipulator's end-effector were empirically set to 0.05 $\mathrm{m/s}$ and 0.5 $\mathrm{m/s^2}$, respectively. During trajectory execution, the robot manipulator followed a constant line speed to maintain consistency of image acquisition (the acquisition line rate of the scanner is 3000 $\mathrm{line/s}$). Table I summarizes the other parameters for the line scanning system used for the experiment.

B. Path generation and defect inspection

Fig. 8 presents five 3C component models. Each 3D mesh model (or CAD model) was converted into a point cloud to identify the geometrical features through uniform and random sampling [41], as shown in Fig. 8. Some geometrical features, such as surface normals, Gaussian curvature, and mean curvature, are computed by a point cloud processing software named CloudCompare [42]. Then, the point cloud was inputted into the proposed method for estimating the scanning path.

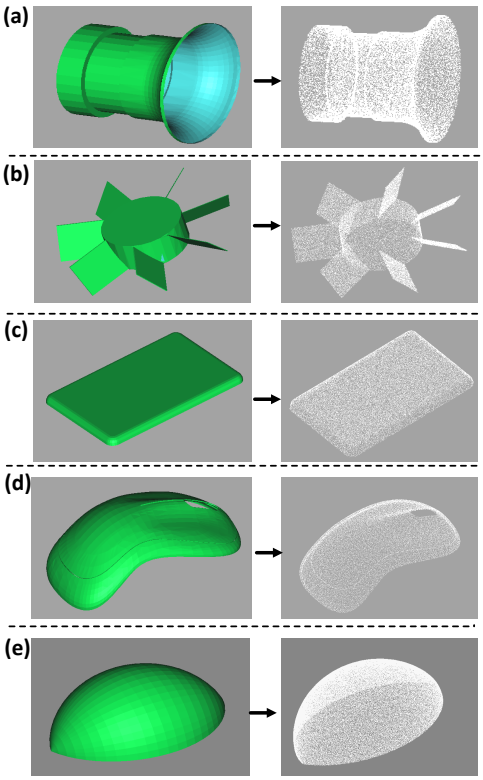

Fig. 8. The model and point cloud of workpieces, (a) Case 1: the camera lens; (b) Case 2: the computer fan; (c) Case 3: the tablet back cover; (d) Case 4: upper part of computer mouse; (e) Case 5: convex free-form surface.

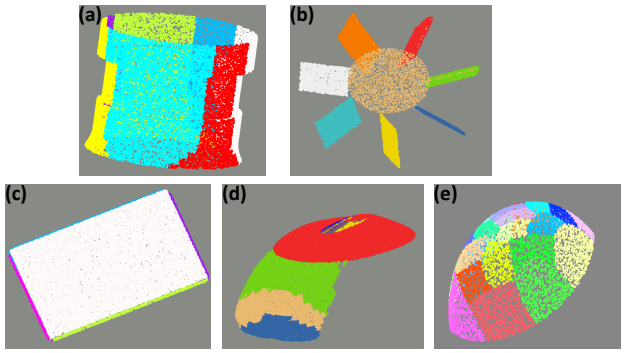

Fig. 9. The segmentation result by the proposed method, (a) Case 1: the camera lens; (b) Case 2: the computer fan; (c) Case 3: the tablet back cover; (d) Case 4: upper part of computer mouse; (e) Case 5: convex free-form surface.

The results from the hybrid segmentation method are shown in Fig. 9, where the different colors indicate various segmented regions (or clusters). Here, the methods used RANSAC to cluster the plane region. In Case 3 and Case 4, a significant portion of the planar/near-planar region has been grouped in one cluster, as shown in Fig. 9(c). Initial clustering using RANSAC significantly reduces the processing time. After the hybrid unsupervised region segmentation, the surfaces with similar geometric features were clustered together. Fig. 9 shows the five geometrically diverse workpieces, and each is divided into different regions based on the features. Some segmentation errors will remain due to the uncertain nature of computing features, but if they do not affect the scanning path generation.

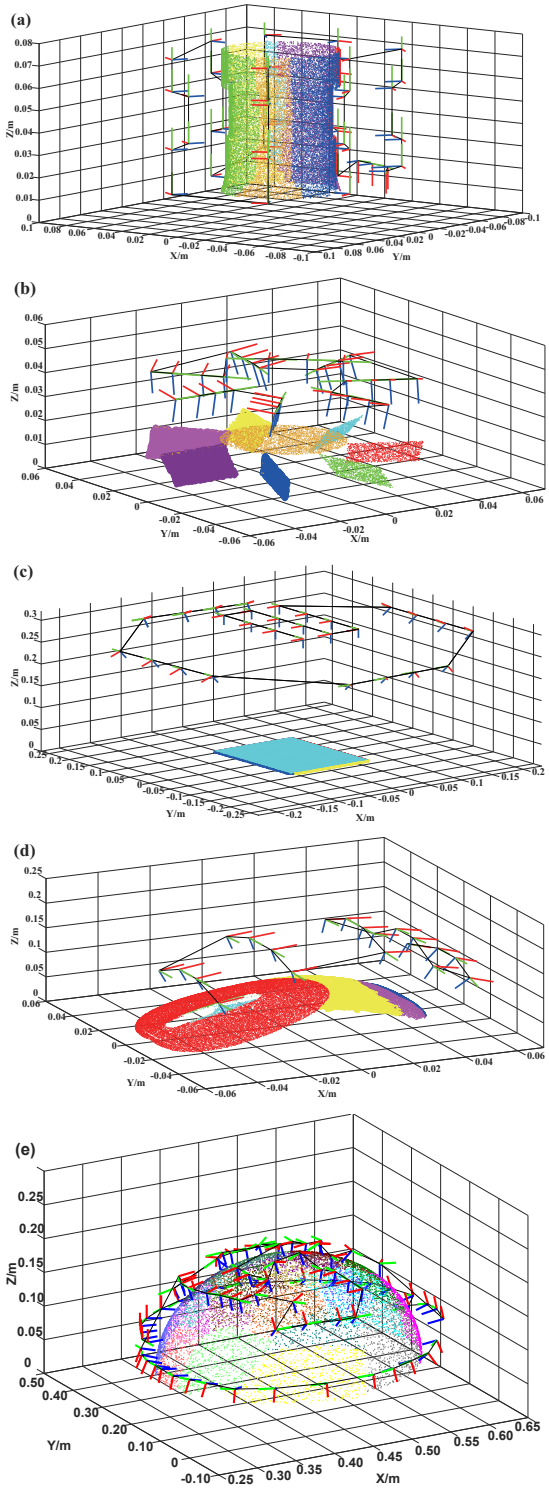

Fig. 10. Coverage path planning through the proposed method, (a) Case 1: the camera lens; (b) Case 2: the computer fan; (c) Case 3: the tablet back cover; (d) Case 4: upper part of computer mouse; (e) Case 5: convex free-form surface.

With adaptive ROI-based path planning and PSO-based global path generation, a complete and near-optimal inspection path can be produced, which is visualized in Fig. 10. The number of viewpoints is 48, 48, 42, 30, 96 in Case 1-5 respectively, displayed by the frames. They show the pose of the

robot's end-effector during the inspection task. The global path planning is demoted with a black line and every segmentated region has a corresponding local path. The different viewpoints are connected by straight lines in the optimal sequence. The robotics motion should follow this detection path to achieve full object coverage.

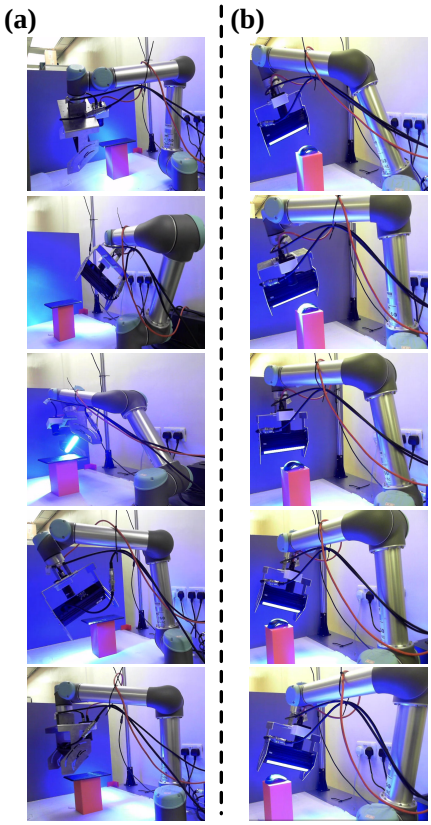

Fig. 11. Experimental scenarios of 3C component surface inspection by the proposed method, (a) Case 3: the tablet back cover; (b) Case 4: upper part of computer mouse.

We input the inspection paths to the automatic line scanning system to scan the tablet back cover and upper part of a computer mouse in order to mimic the real defect inspection, as illustrated in Fig. 11. Fig. 12 illustrates the surface defects of these two objects. The segmentation process in our method results in uniformly geometric regions, allowing for the strategic selection of inspection viewpoints using an ROI-based technique congruent with line-scan camera parameters. This precision enables the system to detect surface defects with high clarity, surpassing human visual inspection capabilities, particularly in typically overlooked areas such as corners and curves. The innovative aspect of our method lies in its integrated approach, combining advanced region segmentation with intelligent local and global path planning algorithms. This comprehensive approach facilitates meticulous inspection of surface imperfections, thereby enhancing processing and quality control within the 3C industry.

C. Comparative analysis and verification

To further validate the proposed CPP method, two related cutting-edge scanning CPP methods, the clustering-based

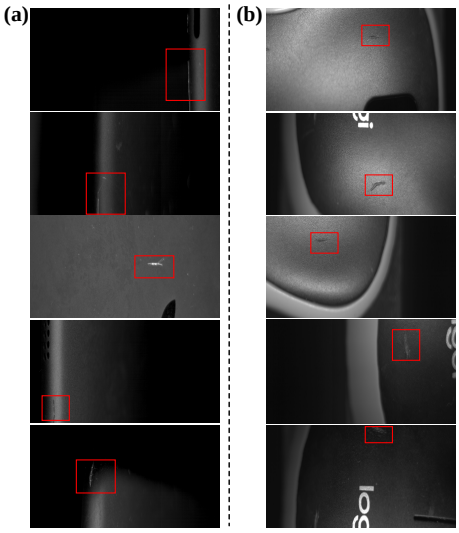

Fig. 12. Representative defect images obtained by the proposed method, (a) Case 3: the tablet back cover; (b) Case 4: upper part of computer mouse.

method TCluster [43], the convex specular surface inspection method [8], and CPP with measurement uncertainty control [9] are applied as benchmark approaches for comparative analysis.

In the local path definition, the region segmentation time was used as a measure of efficiency for region segmentation methods. Additionally, when defect results or coverage rates were similar, preference was given to the CPP method that generated fewer viewpoints as it was considered a more viable path planning approach [44]. The generation results are indicated in Table II and Table III respectively. For region segmentation time, the proposed method used less time to finish this procedure. Due to the usage of RANSAC and more geometric features, the proposed method can obtain the subregions with planar/near-planar geometry efficiently. As for the viewpoints, our developed approach produces fewer viewpoints since more accurate region segmentation results and concise ROI generation. Conversely, the convex specular surface inspection method and TCluster method employed a more complex iteration process for viewpoint determination, as they struggled to segment objects precisely with intricate geometries. The proposed hybrid region segmentation method eliminates the need for complex data training prior to execution. By directly leveraging geometric features rather than relying on regression-based feature analysis, it achieves accurate segmentation with high computational efficiency. This approach is particularly well-suited for real-time applications or resource-constrained industrial settings, offering a practical and cost-effective solution.

For the global path planning, the comparison results are shown in Fig. 13. The inspection path length and total detection time served as indicators of overall path efficiency in CPP methods. Our proposed method demonstrates superior performance in terms of both inspection path length and time compared to benchmark approaches. While these benchmarks relied on relative optimization techniques, they were unable to produce a globally feasible inspection path for CPP. In contrast, our PSO-based approach adeptly tackles the TSP by setting

realistic optimization targets and selecting feasible viewpoints that align with the inspection requirements. Although the performance of these two benchmark approaches is good, our method significantly enhances efficiency by completing inspections in less time and with shorter paths.

In this comparison, a defect detection method was applied for surface quality inspection [8], ensuring consistency by using the same image enhancement technique as the benchmark for noise reduction. The surface defect detection rate was employed as a direct metric to evaluate the effectiveness of defect acquisition, reflecting the accuracy of region segmentation and the quality of path planning. Furthermore, our method achieves a higher detection rate compared to the benchmark, primarily due to the superior performance of viewpoint selection, which ensures better coverage and precision. Noise in surface inspection, commonly caused by factors such as illumination inconsistencies, surface roughness, and scanning misalignment, poses significant challenges. Our approach leverages intrinsic geometric features, including surface normals and curvatures, to effectively mitigate the impact of these variations. This enhances the robustness of the proposed method to noise, ensuring consistent and reliable performance under diverse operational conditions. This comprehensive evaluation positions our CPP method as a more effective alternative to conventional line scanning inspection techniques. The results underscore the value and practicality of our method in enhancing CPP for surface defect inspections.

TABLE II
COMPARISON OF REGION SEGMENTATION EFFICIENCY IN CASE 1-5

Region segmentation time (s)	Case 1	Case 2	Case 3	Case 4	Case 5
The proposed method	8.06	2.19	10.31	3.34	12.33
Convex specular surface inspection method	23.34	4.28	31.73	11.46	126.01
TCluster method	30.87	5.38	39.51	13.28	67.78

TABLE III
COMPARISON OF NUMBER OF VIEWPOINTS IN CASE 1-5

Number of viewpoints	Case 1	Case 2	Case 3	Case 4	Case 5
The proposed method	48	48	42	30	96
Convex specular surface inspection method	96	66	78	54	196
TCluster method	90	80	75	63	112
CPP with measurement uncertainty control	94	73	85	61	172

V. CONCLUSION

This paper proposes a systematic framework for an inspection CPP method for 3C component surfaces. According to this framework, a high-resolution line scanning sensor, mounted on a multi-DOF robotic manipulator, can execute surface scanning and detection precisely and flexibly, effectively addressing the line scanner viewpoint planning challenges for complex surface detection with accurate quality assessment requirements in the 3C industry. The developed methodology includes (1) a new hybrid region segmentation method based on the RANSAC and K-means clustering method; (2) an adaptive ROI method to define the local measurement paths; and (3) a PSO-based global optimization approach for the minimum inspection time. Five case studies verify the effectiveness and efficiency of this method. The results show it outperforms the state-of-the-art line scanning CPP method according to comparison. Overall, the proposed method can achieve precise and efficient surface inspection for 3C free-from components. It can be applied in the 3C industry and be extended to inspect other structures such as auto spare parts and industry-standard components.

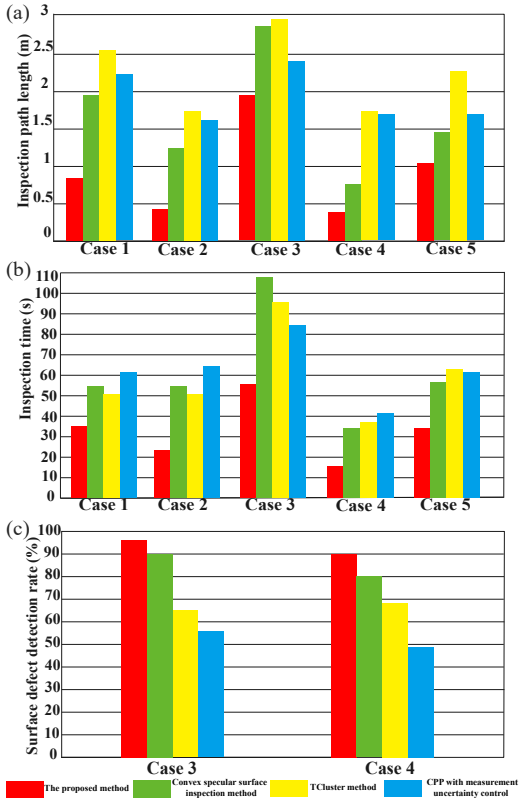

Fig. 13. Results for global path planning and defect inspection of the proposed method, Convex specular surface inspection method, TCluster method, and CPP with measurement uncertainty control in Case 1-5. Comparison of (a) Inspection path length, (b) Inspection time, and (c) Surface defect detection rate.

However, it should be noted that the proposed method may encounter challenges when applied to workpieces with complex structures, making it less suitable for parts with intricate shapes. The proposed segmentation approach might struggle with surfaces containing sharp corners or microtextures. Line scanners have inherent field-of-view constraints. For highly intricate geometries, areas with deep recesses or overhangs might remain uninspected unless additional scanning mechanisms or multi-view strategies are employed. As the geometric complexity increases, the computational demand for region segmentation and global path optimization grows significantly. Future research should focus on optimizing the design of the system end-effector to enhance the flexibility of the inspection framework. For instance, adaptive sensor configurations, including multi-angle line scanners or hybrid

scanning systems, could address field-of-view limitations. Additionally, exploring mathematical methods for optimal path planning and investigating the potential of information theory and deep learning techniques, such as convolutional neural networks, could further improve the effectiveness of the segmentation method.

REFERENCES

[1] H. Chen, H. Xu, and Z. Yang, "A novel hybrid method to detect arrival times of elastic waves with low snr based on jensen-shannon divergence and cumulative sum algorithm," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, 2022.

[2] H. Chen and Z. Yang, "Arrival picking of acoustic emission signals using a hybrid algorithm based on aic and histogram distance," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, 2021.

[3] Y. Liu, W. Zhao, T. Lutz, and X. Yue, "Task allocation and coordinated motion planning for autonomous multi-robot optical inspection systems," *Journal of Intelligent Manufacturing*, vol. 33, no. 8, pp. 2457–2470, DEC 2022.

[4] W. Ming, F. Shen, X. Li, Z. Zhang, J. Du, Z. Chen, and Y. Cao, "A comprehensive review of defect detection in 3c glass components," *Measurement*, vol. 158, JUL 1 2020.

[5] J. Fu, I. Burzo, E. Iovene, J. Zhao, G. Ferrigno, and E. De Momi, "Optimization-based variable impedance control of robotic manipulator for medical contact tasks," *IEEE Transactions on Instrumentation and Measurement*, 2024.

[6] Z. Li, K. Tang, P. Hu, and L. Huang, "Five-axis trochoidal sweep scanning path planning for free-form surface inspection," *IEEE Transactions on Automation Science and Engineering*, vol. 20, no. 2, pp. 1139–1155, APR 2023.

[7] T. Yang, J. V. Miro, M. Nguyen, Y. Wang, and R. Xiong, "Template-free nonrevisiting uniform coverage path planning on curved surfaces," *IEEE/ASME Transactions on Mechatronics*, pp. 1–9, 2023.

[8] S. Huo, B. Zhang, M. Muddassir, D. T. W. Chik, and D. Navarro-Alarcon, "A sensor-based robotic line scan system with adaptive roi for inspection of defects over convex free-form specular surfaces," *IEEE Sensors Journal*, vol. 22, no. 3, pp. 2782–2792, FEB 1 2022.

[9] Y. Liu, W. Zhao, H. Liu, Y. Wang, and X. Yue, "Coverage path planning for robotic quality inspection with control on measurement uncertainty," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 5, pp. 3482–3493, OCT 2022.

[10] H. Chen, S. Li, J. Fan, A. Duan, C. Yang, D. Navarro-Alarcon, and P. Zheng, "Human-in-the-loop robot learning for smart manufacturing: A human-centric perspective," *IEEE Transactions on Automation Science and Engineering*, 2025.

[11] J. Molina, J. Ernesto Solanes, L. Arnal, and J. Tornero, "On the detection of defects on specular car body surfaces," *Robotics and Computer-Integrated Manufacturing*, vol. 48, pp. 263–278, DEC 2017.

[12] E. Glorieux, P. Franciosa, and D. Ceglarek, "Coverage path planning with targetted viewpoint sampling for robotic free-form surface inspection," *Robotics and Computer-Integrated Manufacturing*, vol. 61, FEB 2020.

[13] R. Almadhoun, T. Taha, L. Seneviratne, J. Dias, and G. Cai, "A survey on inspecting structures using robotic systems," *International Journal of Advanced Robotic Systems*, vol. 13, NOV 18 2016.

[14] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258–1276, DEC 2013.

[15] H. M. La, R. S. Lim, B. B. Basily, N. Gucunski, J. Yi, A. Maher, F. A. Romero, and H. Parvardeh, "Mechatronic systems design for an autonomous robotic system for high-efficiency bridge deck inspection and evaluation," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 6, pp. 1655–1664, DEC 2013.

[16] R. S. Lim, H. M. La, and W. Sheng, "A robotic crack inspection and mapping system for bridge deck maintenance," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 2, pp. 367–378, APR 2014.

[17] D. P. Arab, M. Spisser, and C. Essert, "Complete coverage path planning for wheeled agricultural robots," *Journal of Field Robotics*, 2023 MAY 9 2023.

[18] Y. Zhou, P. Li, Z. Ye, L. Yue, L. Gui, X. Jiang, X. Li, and Y.-h. Liu, "Building information modeling-based 3d reconstruction and coverage planning enabled automatic painting of interior walls using a novel painting robot in construction," *Journal of Field Robotics*, vol. 39, no. 8, pp. 1178–1204, DEC 2022.

[19] K. Zbiss, A. Kacem, M. Santillo, and A. Mohammadi, "Automatic collision-free trajectory generation for collaborative robotic car-painting," *IEEE Access*, vol. 10, pp. 9950–9959, 2022.

[20] N. Bolourian and A. Hammad, "Lidar-equipped uav path planning considering potential locations of defects for bridge inspection," *Automation in Construction*, vol. 117, SEP 2020.

[21] N. Kapetanovic, N. Miskovic, A. Tahirovic, M. Bibuli, and M. Caccia, "A side-scan sonar data-driven coverage planning and tracking framework," *Annual Reviews in Control*, vol. 46, pp. 268–280, 2018.

[22] C. Steger and M. Ulrich, "A camera model for line-scan cameras with telecentric lenses," *International Journal of Computer Vision*, vol. 129, no. 1, JAN 2021.

[23] G. Wang, L. Zhao, and H. Zheng, "A new calibration method of stereo line-scan cameras for 2d coordinate measurement," *Frontiers in Physics*, vol. 10, JUN 21 2022.

[24] B. Li, P. Feng, L. Zeng, C. Xu, and J. Zhang, "Path planning method for on-machine inspection of aerospace structures based on adjacent feature graph," *Robotics and Computer-Integrated Manufacturing*, vol. 54, pp. 17–34, DEC 2018.

[25] S. Gerbino, D. M. Del Giudice, G. Staiano, A. Lanzotti, and M. Martorelli, "On the influence of scanning factors on the laser scanner-based 3d inspection process," *The International Journal of Advanced Manufacturing Technology*, vol. 84, no. 9-12, pp. 1787–1799, JUN 2016.

[26] E. Strakova, D. Lukas, Z. Bobovsky, T. Kot, M. Mihola, and P. Novak, "Matching point clouds with stl models by using the principle component analysis and a decomposition into geometric primitives," *Applied Sciences-Basel*, vol. 11, no. 5, MAR 2021.

[27] B. Xu, W. Jiang, J. Shan, J. Zhang, and L. Li, "Investigation on the weighted ransac approaches for building roof plane segmentation from lidar point clouds," *Remote Sensing*, vol. 8, no. 1, JAN 2016.

[28] D. Li, Y. Cao, X.-s. Tang, S. Yan, and X. Cai, "Leaf segmentation on dense plant point clouds with facet region growing," *Sensors*, vol. 18, no. 11, NOV 2018.

[29] M. Liu, Y. Shao, R. Li, Y. Wang, X. Sun, J. Wang, and Y. You, "Method for extraction of airborne lidar point cloud buildings based on segmentation," *Plos One*, vol. 15, no. 5, MAY 29 2020.

[30] L.-H. Juang and M.-N. Wu, "Psoriasis image identification using k-means clustering with morphological processing," *Measurement*, vol. 44, no. 5, pp. 895–905, JUN 2011.

[31] P. Li, R. Wang, Y. Wang, and G. Gao, "Automated method of extracting urban roads based on region growing from mobile laser scanning data," *Sensors*, vol. 19, no. 23, DEC 2019.

[32] M. Kirisci, "New cosine similarity and distance measures for fermatean fuzzy sets and topsis approach," *Knowledge and Information Systems*, vol. 65, no. 2, pp. 855–868, FEB 2023.

[33] R. M. Claro, M. I. Pereira, F. S. Neves, and A. M. Pinto, "Energy efficient path planning for 3d aerial inspections," *IEEE Access*, vol. 11, pp. 32 152–32 166, 2023.

[34] Z. Shang, J. Bradley, and Z. Shen, "A co-optimal coverage path planning method for aerial scanning of complex structures," *Expert Systems with Applications*, vol. 158, NOV 15 2020.

[35] Y. Xu, Y. Han, Z. Sun, W. Gu, Y. Jin, X. Xue, and Y. Lan, "Path planning optimization with multiple pesticide and power loading bases using several unmanned aerial systems on segmented agricultural fields," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 3, pp. 1882–1894, MAR 2023.

[36] A. A. Karim, N. A. M. Isa, and W. H. Lim, "Hovering swarm particle swarm optimization," *IEEE Access*, vol. 9, pp. 115 719–115 749, 2021.

[37] Z.-H. Zhan, J. Zhang, Y. Li, and H. S.-H. Chung, "Adaptive particle swarm optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 6, pp. 1362–1381, DEC 2009.

[38] Z. Su, Z. Gao, G. Zhou, S. Li, L. Song, X. Lu, and N. Kang, "Building plane segmentation based on point clouds," *Remote Sensing*, vol. 14, no. 1, JAN 2022.

[39] M. Bertolini, D. Mezzogori, M. Neroni, and F. Zammori, "Machine learning for industrial applications: A comprehensive literature review," *Expert Systems with Applications*, vol. 175, p. 114820, 2021.

[40] C. Gupta and A. Farahat, "Deep learning for industrial ai: Challenges, new methods and best practices," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 3571–3572.

[41] E. Arias-Castro, D. L. Donoho, and X. Huo, "Adaptive multiscale detection of filamentary structures in a background of uniform random points," *Annals of Statistics*, vol. 34, no. 1, pp. 326–349, FEB 2006.

[42] Y. Tang, Y. Wang, H. Tan, W. Peng, H. Xie, and X. Liu, "A digital twin-based intelligent robotic measurement system for freeform surface

parts," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–13, 2023.

[43] M. Na, H. Jo, and J.-B. Song, "Cad-based view planning with globally consistent registration for robotic inspection," *International Journal of Precision Engineering and Manufacturing*, vol. 22, no. 8, pp. 1391–1399, 2021.

[44] Y. Liu, W. Zhao, R. Sun, and X. Yue, "Optimal path planning for automated dimensional inspection of free-form surfaces," *Journal of Manufacturing Systems*, vol. 56, pp. 84–92, JUL 2020.