IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2024, Yokohama, Japan. Cite as T-ASE paper.

Deep Dynamic Layout Optimisation of Photogrammetry Camera Position based on Digital Twin

Likun Wang, Zi Wang, Peter Kendall, Kevin Gumma, Alison Turner, and Svetan Ratchev

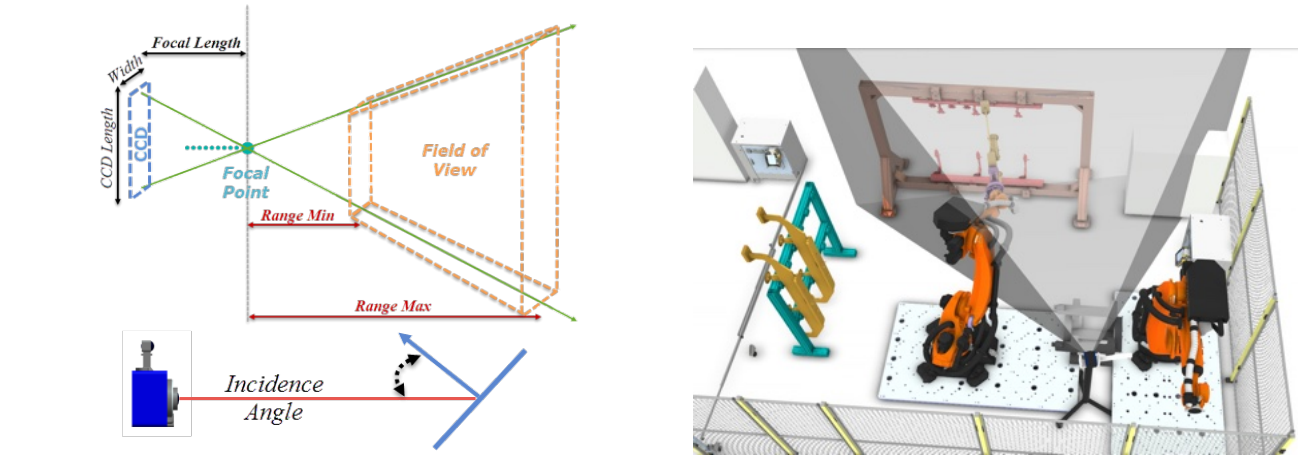
***Abstract*— The photogrammetry system has been widely used in industrial manufacturing applications, such as high-precision assembly, reverse engineering and additive manufacturing. In order to meet the demand of the product variety and short product lifecycle, the factory facilities including photogrammetry devices, should be relocated in response to rapid change in mechanical structure and hardware integration. Nevertheless, the camera position of the photogrammetry system is difficult to select to guarantee an optimal field of view (FoV) coverage of retro-reflective targets during the whole production horizon. Especially in a reconfigurable manufacturing work cell, scaling and calibration of a photogrammetry system requires professional skills and these would cost tremendous labour for rapid configuration each time. In this paper, we propose a novel deep optimisation framework for the photogrammetry camera position for the dynamic layout design based on digital twin. The optimisation framework follows an effective coarse-to-fine procedure to evaluate the FoV visibility over the target frame. In addition, the deep Q-learning algorithm is utilised to find the maximum FoV coverage and avoid collision. Three experiments are implemented to verify the application feasibility of the proposed deep camera position optimisation framework. *Note to Practitioners* - Large-volume in-process metrology is an essential element in flexible manufacturing systems. Quality of large-volume measurement relies heavily on target visibility within its field of view. In a compact industrial robotic cell, this is extremely challenging as the robot would take the primary position and causing view blockage throughout its operation. This makes the simultaneous monitoring of robot head and the work piece key feature extremely difficult. Manual trial-and-error positioning approach is lengthy and requires high level of expertise, due to both safety and spatial concerns. We approached this problem by simulating the camera's view in a digital twin environment and maximising the target visibility throughout the full operation cycle. The generic framework can provide guidance in metrology setup within automated manufacturing environment, accelerate the system commissioning time, remove dependency of skill level and expand the capability for flexible/reconfigurable manufacturing systems. Although V-STARS camera are used in this application, the framework can be applied for other types of vision systems that requires field of view.**

I. Introduction

Manufacturing industry is facing growing demands for customised, high-value and low-volume products. Many new research themes emerged aiming to improve manufacturing flexibility, such as modular production system, human-machine collaboration [1] and digital twin. The conception of reconfigurable manufacturing systems were proposed in [2] to tackle this challenge with a systematic approach. It relies on the rapid transformation of factory facilities to adjust its capacity and functionality and improve production efficiency. Consequently, the reconfiguration of a manufacturing system always involves the physical rearrangement and relocation of autonomous facilities [3], including measurement system, such as photogrammetry devices [4], [5]. Traditionally, product tolerance and key datum can be fixed and maintained with high-specification machining and fixtures. Nevertheless, in flexible and reconfigurable manufacturing systems, bespoke jig and fixed tooling are minimised [6]. Therefore, key characteristics are achieved with tool calibration and external measurement [7], [8].

Over the last decade, photogrammetry has been widely used as a three dimensional measurement tool in developing manufacturing applications, such as additive manufacturing [8], [9], aerospace jig assembly [7], reverse engineering [10] and manipulator calibration [11]. As a key component in modern manufacturing, it allows product key features to be controlled at its root which leads to reduction in rectification and rework [12]. Compared with other portable metrology devices such as laser tracker [13] and laser scanner [14], photogrammetry cameras can simultaneously monitor a large volume of targets and are also suitable for unstable applications, such as measurement on vibrating platform [15], hand-held operation [16], and real-time 3D measurement [17]. A standard optical coordinate measurement process consists of firmware setting, camera position selection, image collection and processing, 3D reconstruction and data analysis [18]. Despite the fact that optical coordinate measurement has gained large market share in industrial application, there is no automated or established method for measuring and inspection planning. Camera position selecting is one of the most important issues that constrain the utilisation of optical coordinate measurement by experienced operators [19]. As camera positions are related to not only image acquisition and data post-processing but also target visibility and measurement accuracy, it makes the optical camera positioning crucial in industrial manufacturing.

However, the quality of photogrammetry measurements is directly related to its field of view (FoV). Especially in a robotic cell, the manipulator motions can cause view obstruction, which leads to invisibility of target features and inaccurate measurement results. Again, manual configuration is not desirable and more importantly, it is difficult to consider all the possible tracking position manually. When integrated in a reconfigurable manufacturing system, the cameras are also required to relocate. Not only the external transformation matrix should be re-characterised but also the target markers can be relocated. This would cost tremendous unavoidable labour in finding the best camera position when the facility layout is changed every time. Consequently, a photogrammetry system optimisation as a part of facility



(a) Graphic explanation of the V-STARS FoV [20]. (b) Virtual V-STARS in digital twin.

Fig. 1: Graphic explanation of the FoV of the V-STARS photogrammetry system and its virtual model in digital twin. The digital twin is established in Visual Components. The cameras of the V-STARS system could provide three dimensional measurement as given in Fig. 1a. However, during the assembly processes, the manipulator might blind the FoV of the V-STARS system as shown in Fig. 1b, which could lead to inaccurate or overfitting measurement results.

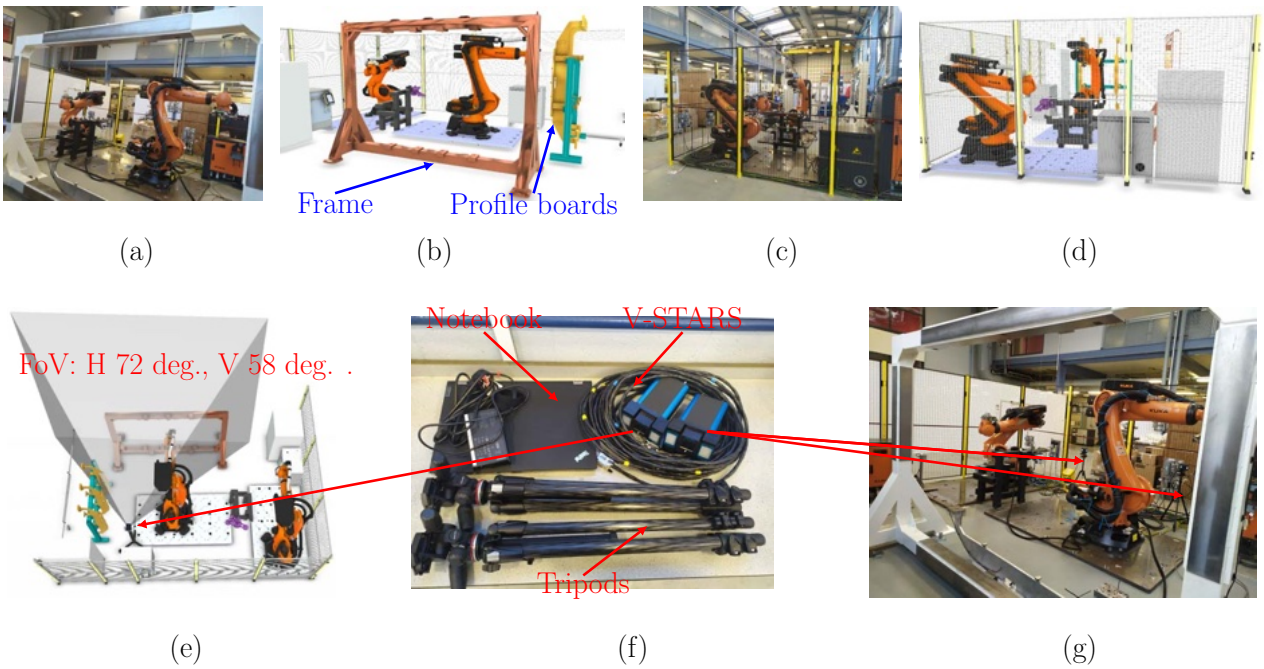


Fig. 2: The digital twin of the work cell and V-STARS photogrammetry system. (a) Front view of the assembly work cell. (b) Front view of the digital win model. (c) Side view of the assembly work cell. (d) Side view of the digital win model. (e) V-STARS camera FoV representation (72 deg. horizontally and 58 deg. vertically) in the digital twin. (f) V-STARS Photogrammetry system. (g) V-STARS system located in the work cell. The physical work cell are shown in Fig. 2 (a) and Fig. 2 (b). Their corresponding side view in VC are presented in Fig. 2 (b) and Fig. 2 (d), separately. The V-STARS system includes two tripods, two V-STARS cameras and a laptop PC for measurement data analysis as shown in Fig. 2 (f). The parameters of the FoV are 72 deg. horizontally and 58 deg. vertically as indicated in Fig. 2 (e). The V-STARS system is located in the work cell as presented in Fig. 2 (g).

layout planning is needed to facilitate the rapid changeover.

However, there are very limited studies considering camera FoV in a production environment. Currently studies [21]–[23] mainly focuses on calibration when facing limited object visibility with fixed camera FoV under challenging circumstances, such as monitoring and surveillance. These applications do not require relocation or rapid decision about camera positioning. A few publications address the camera positioning derived from product CAD models [24]. Nevertheless, these publications only focus on the product

and is only applicable in the final product inspection phase. Theses studies also ignore the fixtures involved and the overall assembly processes. To the best of our knowledge, no research has been done for camera position optimisation of photogrammetry systems in a reconfigurable manufacturing environment.

In this paper, a novel camera position optimisation framework is proposed in a digital twin environment as presented in Fig. 1a and Fig. 1b. The framework considers the camera FoV and collision for the entire assembly process, enabling in-process metrology simulation, removing high manual effort and improving system commissioning efficiency. By using coarse-to-fine registration, the FoV coverage of the target frame is estimated. Considering a dynamically changing assembly process, a deep reinforcement learning algorithm is applied, instead of any mode. The photogrammetry system is considered as an agent, which searches for its optimal position in the digital twin environment regarding FoV estimation and collision detection within the facility. In order to improve readability, the methodology is outlined as follows:

1) The digital twin of the generic assembly work cell along with the photogrammetry system is established in a manufacturing simulation software package, Visual Components (VC) and the camera FoVs throughout the overall production are simulated and captured;
2) A generic FoV evaluation approach based on target geometry information and coarse-to-fine point cloud registration is proposed;
3) Position optimisation algorithm based on deep Q reinforcement learning is introduced aiming to maximise camera FoV and avoid collision with the robot arm;
4) Three experiments are performed to validate the digital twin environment and camera parameters, and then demonstrate the feasibility of the proposed optimisation framework.

The remainder of the paper is organised in four parts. After the introduction, a literature review is conducted in Section II. The FoV visibility evaluation approach is given in Section III. Then, the deep learning framework for layout optimisation is presented in Section IV. Finally experiments are described and analysed in Section V, after which conclusions are drawn in the last section.

II. Related Works

Given that measurement accuracy is directly related to object visibility within the camera FoV, the photogrammetry camera position is crucial. In [24], the optimisation of camera positions was studied based on a hidden point removal method, which reconstructs the surface of several products and quantitatively estimates the object visibility. Although the camera position optimisation was derived from CAD models and the result seems promising, it only considers the measurement of the product at a single moment and nothing suggested that the impact of the camera position was considered for the production lifecycle.

In [21], a global calibration method of multi-camera was introduced with assumption of non-overlapping FoV. This method is based on a dual-camera photogrammetry system and a reconfigurable target. To enlarge the FoV coverage of an object, a network of multiple camera viewpoint was proposed in [22], where one camera with relative movements and multiple cameras are compared and discussed. However, these two publications focused on camera extrinsic parameter calibration to increase the FoV visibility not the optimisation of camera positions.

In [23], two algorithms, namely, alternate global greedy algorithm and global greedy algorithm, for optimal camera configuration and multi-camera locations. As the use case application is for surveillance networks, it doesn't need to take into account the view blockage of other dynamic facilities in typical industrial manufacturing applications. In [25], multi-criteria decision analysis was proposed to select the optimal position and the appropriate FoV for a photosensor. Nevertheless, the factors taken into account were lighting levels, corresponding to energy saving and lighting adequacy.

Additional applications of photogrammetry technology in manufacturing layout design mainly focus on the construction or reconstruction of the existing layout. Regarding inspection application in industrial production systems, instead of using a laser scanner, in [26], photogrammetry techniques were applied to update their digital twin model of the work cell. Similarly, in [27], the manufacturing layout redesign is studied with three-dimensional scanning. By using a simplified version of systematic layout planning, Lowder found that the overall time of design can be reduced. Although these works applied digital twin for modelling overall facility layout, digital twin of the photogrammetry system itself hasn't been investigated.

Furthermore, photogrammetry system was applied in reconstruction applications such as UAV (Unmanned Aerial Vehicle) structure [28], railway [29], and dam inspection [30]. Besides, photogrammetry was used to monitor production processes in manufacturing processes. Given that geometrically complex components are required to be modelled in additive manufacturing, photogrammetry was deployed in [8] to reverse engineer and redesign components. Even though digital information is established via means of a photogrammetry system in these publications, the camera positioning is not optimised.

According to the literature review, no research has been found for position optimisation of photogrammetry systems in reconfigurable manufacturing systems based on their digital twin and hence, this paper is novel in the following aspects. Firstly, it is the first systematic analysis in camera positioning in a production environment, quantifying the target visibility through a FoV evaluation, which would help to remove manual skill requirement, and reduces safety risks. Secondly, it is the first holistic optimisation approach including all possible movement and interaction in the entire production time span within a digital twin, enabling in-process metrology simulation early in system design, as well as in parallel with physical installation.

III. FoV Visibility Evaluation

This section addresses the FoV visibility evaluation using point cloud registration techniques. The digital twin of the assembly cell and V-STARS system are established in Visual Components and described in Section III-A, followed by the point cloud identification based on Gaussian mixture models outlined in Section III-B. Finally, point cloud alignment is implemented in a coarse-to-fine manner as detailed in Section III-C.

A. Digital Twin Modelling and Reconfigurable Manufacturing Work Cell

A digital twin presents all the digital information of a physical factory and accompanies the factory through its whole production lifecycle, and the digital modelling can be realised in commercial simulation packages, such as Process Simulate [31], Gazebo [32] and Visual Components [33] etc. In this paper, the digital twin platform is built in the Visual Components environment.

The example taken here is a generic assembly work cell, as shown in Fig. 2, in a reconfigurable manufacturing system which supports multiple products within a product family. The rectangular frame can be configured with different tooling to suit the assembly of different products. As long as the product family shares commonalities in size and build philosophy, a reconfigurable assembly system can be used. A similar tooling system was presented in [34], [35]. As presented in Fig. 2, the digital twin built in Visual Components replicates the physical layout of the assembly work cell. There are two Kuka KR270 ultra robots located in the cell as shown in the physical layout in Fig. 2 (a) and Fig. 2 (c), and the virtual environment in Fig. 2 (b) and Fig. 2 (d). The task for the manipulator is to configure a jig frame, with an attachable board to support the product's profile, through a pick-and-place process.

The digital twin first starts as loosely aligned simulation that validate the system design. Through positional data input, it is then aligned with the physical counterpart more accurately. The loosely defined digital twin would already have meaningful reference coordinate systems, whether that is the datum system of the frame, or the base frame/world frame of a robot. The digital twin is used for two purposes, one is to simulate the camera FoV, and the other is to put the captured data into a meaningful reference system. In the experiment validation in Section V-A, the layout positional data are acquired via the robot programs, hence maintaining a good level of alignment between the physical and digital systems.

During assembly, the position of the frame, the profile board and the manipulator are monitored by a photogrammetry system, which consists of two V-STARS cameras as displayed in Fig. 2 (e), Fig. 2 (f) and Fig. 2 (g). The camera is simulated as 3D scanners in the digital twin model as given in Fig. 2 (e) to represent the physical cell setup as illustrated in Fig. 2 (g). In the digital twin, cameras output the ASCII-data including point cloud positions and their corresponding colours.

Regarding reconfiguration, the components with a common interface are able to be loaded and unloaded robotically. The V-STARS photogrammetry system is utilised to accurately position the profile boards to the target frame. Nevertheless, it is necessary to investigate the FoV coverage over the target frame to guarantee the assembly quality. In this paper, we focus on the pick-and-place of profile boards in a single work cell and demonstrate optimisation framework for the FoV visibility during the dynamic configuration process.

B. Point Cloud Identification

Through the production lifecycle, the visual servo control is usually considered as a key measurement assistance to guarantee the assembly quality. However, the region of interest can be blocked by other autonomous devices, such as manipulators and AGVs. Based on the digital twin, production processes can be replicated and simulated in a virtual environment, including the V-STARS photogrammetry camera.

Given the digital twin established in Visual Components, the nominal frame point cloud (based on its CAD model) can be retrieved by cropping the overall work cell mesh model as illustrated in Fig. 3 (C) and Fig. 3 (D). In addition to this, the reference position of the frame can also be obtained. With a defined camera position vector $\boldsymbol{\xi}^c = [\xi_x^c, \xi_y^c]$, where ξ_x^c and ξ_y^c are the x coordinate and y coordinate, respectively, point cloud is scanned as shown in Fig. 3 (a).

In order to extract the frame from the overall point cloud, DBSCAN (Density-based spatial clustering of applications with noise) [36] is employed to cluster the overall point cloud. As shown in Fig. 3 (b), it groups and categorises four clusters, which are part of robot (dark blue), frame (red), end-effector stand (pink) and power cabinet (light blue).

After clustering, the overall point cloud is divided into several clusters with no identification. Therefore, it is important to distinguish each part and find the target frame point cloud. Using EM-GMM algorithm (Expectation Maximisation of Gaussian Mixture Models) [37], the frame point cloud can be automatically recognised. Once the training is finished, it does not need to be trained again. For establishing the GMM models, 500 point cloud datasets were used, which consist of geometry cluster centres and manually marked corresponding labels. Note that each cluster centre is calculated relative to the V-STARS camera position. After training, given input point cloud clusters, the frame can be chosen based on the largest probability output from the trained GMM models.

C. Coarse-to-fine Registration

After identifying, the frame is extracted from the scanned point cloud as indicated in Fig. 3 (c). The next step is to register two point clouds, namely the identified point cloud (blue) and the nominal frame point cloud (yellow). The nominal frame point cloud is obtained by extracting the STL model from the digital twin environment as shown in Fig. 3 (d). Next, a fast global registration algorithm is used to

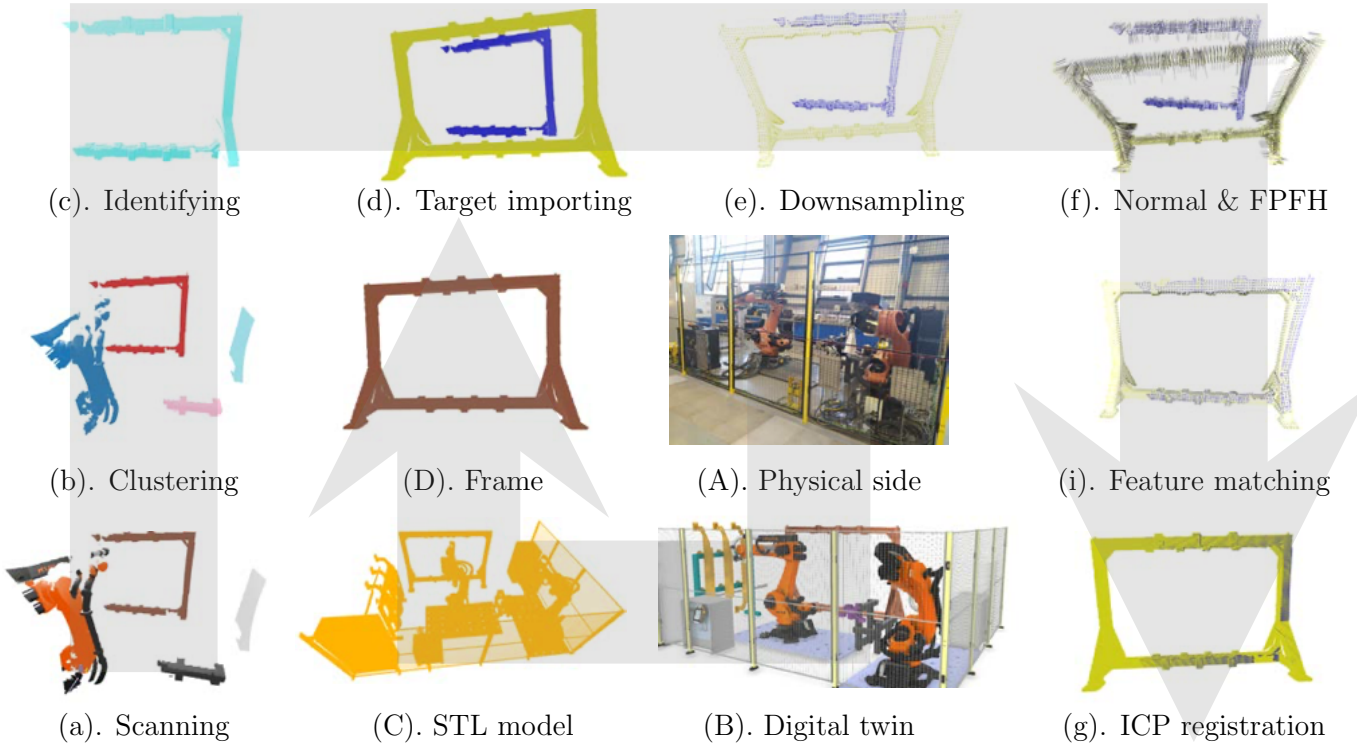


Fig. 3: The whole procedure of the Fov evaluation. The target frame (as given in Fig. 3(D)) is obtained from the stl model as shown in Fig. 3(C). Note that the stl model is derived from the digital twin of the physical layout as represented in Fig. 3(B) and Fig. 3(A), respectively. In addition, the evaluation procedure consists of scanning (Fig. 3(a)), clustering (Fig. 3(b)), identifying (Fig. 3(c)), target importing (Fig. 3(d)), downsampling (Fig. 3(e)), feature extraction (Fig. 3(f)), feature matching (Fig. 3(i)) and iterative closet point registration (Fig. 3(g)).

roughly align the two point clouds and the error objective function is defined as

$$\varepsilon(\mathcal{T}, L) = \sum_{(\mathsf{p},\mathsf{q})\in\mathcal{S}} l_{(\mathsf{p},\mathsf{q})}||\mathsf{p} - \mathcal{T}\mathsf{q}|| + \sum_{(\mathsf{p},\mathsf{q})\in\mathcal{S}} \psi(l_{(\mathsf{p},\mathsf{q})}), \quad (1)$$

where $\mathcal{S} = \{\mathcal{S}|(\mathsf{p},\mathsf{q}) \in \mathcal{S}, \mathsf{p} \in \mathsf{P}, \mathsf{q} \in \mathsf{Q}\}$ is the collection of correspondences. p and q are matching points in point clouds P and Q, respectively. Moreover, in order to minimise the error objective function, $\psi(l_{(\mathsf{p},\mathsf{q})})$ is denoted as a prior and defined as

$$\psi(l_{(\mathsf{p},\mathsf{q})}) = \mu(\sqrt{l_{(\mathsf{p},\mathsf{q})}} - 1)^2. \quad (2)$$

where μ is chosen as the diameter of the largest surface. The term $l_{(\mathsf{p},\mathsf{q})}$ can be calculated from the partial derivative of the error function with respect to each $l_{(\mathsf{p},\mathsf{q})}$. Hence, the term $\psi(l_{(\mathsf{p},\mathsf{q})})$ can be obtained.

The optimisation is performed iteratively between the coefficients $l_{(\mathsf{p},\mathsf{q})}$ and the transformation matrix $\mathcal{T}$

$$\mathcal{T} = \begin{bmatrix} 1 & -\gamma & \beta & a \\ \gamma & 1 & -\alpha & b \\ -\beta & \alpha & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathcal{T}^k \quad (3)$$

with $\mathcal{T}^k$ being the last estimated transformation and the vector $\xi = [a, b, c, \alpha, \beta, \gamma]$ consisting of rotation $[\alpha, \beta, \gamma]$ and translation $[a, b, c]$. The error objective function in Eq. 1 is a least-square objective on ξ and it can be solved by Gaussian-Newton method as detailed in [38]. Then, the transformation $\mathcal{T}$ can be updated according to Equ. 3.

Nevertheless, before registration, the initial correspondence set $\mathcal{S}$ is generated by the Fast Point Feature Histogram (FPFH) [39]. The FPFH is derived from the PFH, which depends on the presence of 3D coordinates and their surface normals. Consequently, for a point p, its neighbours in the sphere with radius r are chosen. For each point pair $(\mathsf{p}_i, \mathsf{p}_j)$ with i, j the index label, the Darboux frame is given as

$$\begin{aligned} u &= n_i \\ v &= (\mathsf{p}_i - \mathsf{p}_j) \times u \\ w &= u \times v, \end{aligned}$$

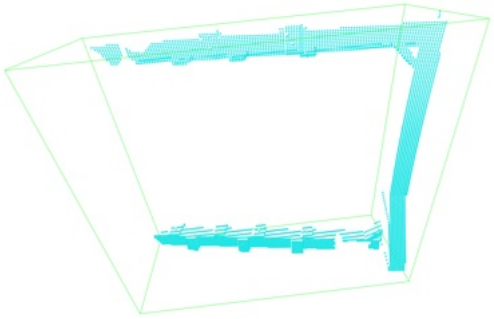
where n_i is the normal from point p_i, which will be used in the following expressions. Additionally, the angular variations of the normals n_i and n_j are

$$\begin{aligned} \beta &= v \cdot n \\ \phi &= (u \cdot (\mathsf{p}_i - \mathsf{p}_j))/||\mathsf{p}_i\mathsf{p}_j|| \\ \vartheta &= \arctan(w \cdot n_i, w \cdot n_j). \end{aligned} \quad (4)$$

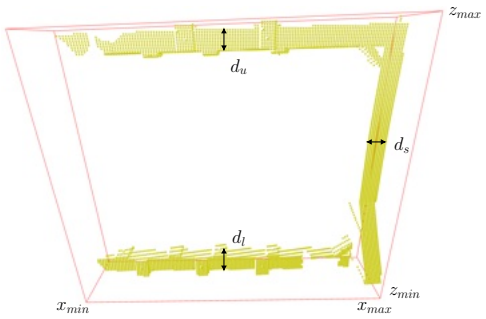
where n_j is the normal from point p_j.

Therefore, the FPFH of the point p can be defined as

$$FPFH(\mathsf{p}) = SPF(\mathsf{p}) + \frac{1}{M}\sum_{m=1}^{M}\frac{1}{d_m} SPF(\mathsf{p}_m), \quad (5)$$



(a)



(b)

Fig. 4: FoV evaluation of the frame. (a) FoV evaluation of the frame without registration. (b) FoV evaluation of the frame with registration. Fig. 4 presents the bounding volume information of the extracted frame point cloud. As shown in Fig. 4 (a), without registration, the point cloud cannot be properly aligned along coordinate axes. Hence, it is difficult to evaluate the visibility. In contrast, after registration as given in Fig. 4 (b), the visibility of the extracted frame point cloud can be quantitatively estimated using bounding volume information.

where SPF (Simplified Point Feature) is derived from the angular variations in Eq. 4. In the above equation, M neighbours of point p are taken into consideration. The distance between a neighbour point p_m and the query point p is defined as d_m in a given metric space.

In summary, for coarse alignment, two point clouds are downsampled as indicated in Fig. 3 (e). Then, the normals at each point are extracted and these features are described as FPFH using KD-tree search [40] as shown in Fig. 3 (f). Thirdly, as given in Fig. 3 (i), the registration of two points are implemented with RANSAC (Random Sample Consensus) algorithm [41], which is typically used for iteratively fitting two point clouds.

Although the fast global registration could provide a rapid alignment and significantly reduce the computational time, the alignment still needs to be further refined. Thus, the alignment obtained from the fast global registration are optimised using point-to-plane iterative closest point (ICP) registration algorithm. The error objective function is defined as

$$\varepsilon(\mathcal{T}) = \sum_{(\mathsf{p},\mathsf{q})\in\mathcal{S}} ((\mathsf{p} - \mathcal{T}\mathsf{q})\boldsymbol{n}_\mathsf{p})^2, \tag{6}$$

with $\boldsymbol{n}_\mathsf{p}$ being the normal from point p, obtained from the normal estimation. The point cloud alignment are further refined as illustrated Fig. 3 (g) with point-to-plane ICP registration algorithm.

Generally, the ICP algorithm recursively implemented over two steps. Firstly, it locates the corresponding set $\mathcal{S}$ generated from FPFH features in both source point cloud P and target point cloud Q based on current transformation $\mathcal{T}$. Secondly, the transformation $\mathcal{T}$ is updated by minimising the objective functions as given in Equ. 1 or Equ. 6 defined over the corresponding set $\mathcal{S}$.

Furthermore, after ICP registration as presented in Fig. 3, FoV visibility evaluation p_{FoV} for the photogrammetry system can be calculated via the bounding volume information derived from the registration result in Section III. Based on the geometry information as given in Fig. 4 the FoV evaluation is defined as

$$p_{FoV} = (d_u+d_l)(x_{max}-x_{min}-d_s)+(z_{max}-z_{min})d_s \tag{7}$$

where d_u and d_l are the width of the upper beam and lower beam, respectively. In addition, d_s is the width of the side beam. x_{min}, x_{max}, z_{min}, and z_{max} are the coordinates derived from the bounding volume information. Most assembly processes consist of fixtures with simple geometry due to the ease of fabrication, therefore the proposed bounding volume calculation can be easily modified, and still be effective in evaluating FoV based on different geometry information.

In fact, without registration, the retrieved point cloud model cannot be aligned to a meaningful coordinate system as indicated in Fig. 4a. Therefore, after the registration, the initial point cloud with transformation can be successfully aligned to a defined coordinate system as given in Fig.4b.

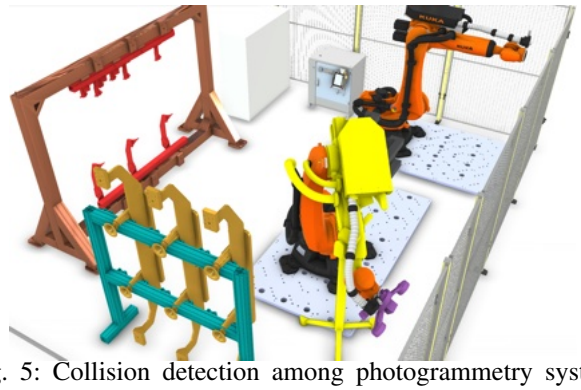


Fig. 5: Collision detection among photogrammetry system and other components.

IV. Deep Camera Position Optimisation

The deep camera position optimisation of the V-STARS photogrammetry system is detailed in this section. Firstly, the learning objectives of the photogrammetry system are presented in Section IV-A, followed by the deep Q-learning framework for the photogrammetry camera position optimisation in Section IV-B.

A. Learning Objectives

The learning objectives are closely related to the reward in reinforcement learning. One basic learning objective is that during the whole production process, the V-STARS photogrammetry system will not clash with other components in the work cell. Given that the functionality of the photogrammetry system in this application is to monitor, guide and assist assembly, its FoV should see all the retro-reflective targets with minimal object overlapping and blindness, in order to accurately identify and track movement. FoV evaluation follows the workflow presented in Section III, and in this subsection the objective regarding collision detection in VC is given.

Collision should be monitored for the whole production horizon. When a collision happens, the layout design should be considered unacceptable. Collision can be detected in Visual Components as given in Fig. 5, where the parts in clash are highlighted in yellow.

```
from vcScript import *

simu = getSimulation()
app = getApplication()
comp = getComponent()


# obtain all components
scanner = app.findComponent('scanner')
robot1 = app.findComponent('kuka1')
robot2 = app.findComponent('kuka2')
toolStand = app.findComponent('stand')
fence = app.findComponent('fence')
tool = app.findComponent('dbTool')

# collision detection signal
odSignal = comp.findBehaviour('odSignal')

def OnSignal( signal ):
  pass

def OnRun():
  # collision detector
  detector = simu.newCollisionDetector()
  # group 1: scanner
  detector.NodeListA = [(scanner, VC_NODELIST_INCLUDE, VC_NODELIST_TREE )]
  # group 2: robot, end-effector, tool stand, fence
  detector.NodeListB = [(robot1, VC_NODELIST_INCLUDE, VC_NODELIST_TREE ),
                        (robot2, VC_NODELIST_EXCLUDE, VC_NODELIST_TREE ),
                        (toolStand, VC_NODELIST_EXCLUDE, VC_NODELIST_TREE ),
                        (fence, VC_NODELIST_EXCLUDE, VC_NODELIST_TREE ),
                        (tool, VC_NODELIST_EXCLUDE, VC_NODELIST_TREE )]

  while True:
    delay(0.001)
    # detect collision
    hit = detector.testOneCollision(0.1)
    if hit:
      # send signals
      odSignal.signal(True)
      print 'collision detected'
    else:
      odSignal.signal(False)
```

Fig. 6: Example code of collision detection in Visual Components.

Collision detection is implemented in a separate thread which can be seen as a background program. As given in Fig. 6, the collision is checked for the interaction of two organised groups. The first group is only defined for the scanner (V-STARS) component. The second group consists of two robots, tool stand, fence and end-effector. Once a collision is detected, a confirmed signal will be sent to the main thread and then the corresponding penalty will be generated.

Thus, the penalty function is defined as

$$p_{col}(\boldsymbol{\xi}_n) = \begin{cases} 0 & \text{if } V_d = \emptyset \\ v_0 & \text{otherwise,} \end{cases} \tag{8}$$

where $\boldsymbol{\xi}_n$ is the position of the photogrammetry system, v_0 is the positive constant penalty, and V_d is the detected collection of collision. If the V_d collision detection set is empty, there will be no penalty, otherwise the penalty must be considered in the camera position optimisation.

B. Deep Reinforcement Learning Optimisation

The location of the photogrammetry system is optimised by using deep Q-learning [42]. In the learning network, the V-STARS system is considered as an agent which observes the current position state $s_t = [x_t, y_t]$, chooses an action $a_x^t, a_y^t \in \mathcal{A}$ derived from a potential stochastic policy π, and then transit to a novel position state s_{t+1} with a reward $R(s_t, a_t)$, with $a_t = [a_x^t, a_y^t]$. The actions are defined as discrete incremental movement, such as $\mathcal{A} = \{a_x^t \in [-\Delta x, 0, \Delta x], a_y^t \in [-\Delta y, 0, \Delta y]\}$, where Δx and Δy are positive constants. The reward consists of two parts

$$R(s_t, a_t) = \omega_{FoV} p_{FoV} - \omega_{col} p_{col}, \tag{9}$$

where p_{FoV} is obtained from the FoV evaluation at different positions and p_{col} is the reward based on collision detection. Two weighted parameters $\omega_{FoV}, \omega_{col}$ are added to the above reward equation in order to flexibly adjust the learning objectives, as well as to distinguish reward/penalty contributed by each term.

These sequential decision problems are commonly investigated as a finite Markov decision process governed by a tuple of parameters $\langle \mathcal{S}, \mathcal{A}, R, T, \gamma \rangle$. Thus, the Q value function regarding a policy π is given as $Q^\pi(s, a) = \mathcal{E}[R_t | s_t = s, a_t = a]$. The optimal action-value function $Q^*(s, a) = maxQ^\pi(s, a)$ should follow the Bellman optimality equation [43]

$$Q^*(s, a) = \mathcal{E}_{s'}[R(s, a) + \gamma max_{a'} Q^*(s', a') | s, a]. \tag{10}$$

In deep Q-learning, a neural network is used to learn the policy instead of a Q-value table [42]. The following objective function defined by reinforcement learning should be minimised

$$\mathcal{L}_i(\theta_i) = \mathcal{E}_{s,a,r,s'}[\varrho_i - Q(s, a; \theta_i)^2], \tag{11}$$

with θ_i defining the neural network consisted of two hidden layers. Since the point cloud data only consists of two dimensions, two hidden layer is sufficient in capturing data information. Compared with classic Q-learning, the parameter θ_i^- in the target $\varrho_i = r + \gamma max_{a'} Q(s', a'; \theta_i^-)$ is maintained constant for several iteration while learning the online network $Q(s, a; \theta_i)$ by gradient descent optimisation.

V. Evaluation

Evaluation is divided into three parts. The first experiment given in Section V-A is to validate FoV capture between digital twin model and physical environment. In Section V-B, the second experiment is to test the proposed V-STARS FoV visibility evaluation during the overall assembly implementation. Based on the second experiment, in Section V-C, the third experiment aims to find optimal position for photogrammetry camera in the robotic assembly cell.

TABLE I: FoV evaluation result corresponding to Fig. 9

Num.	min x	min y	min z	max x	max y	max z	FoV Eval.
a	3.00088	-0.408994	0.088762	3.73141	2.41133	2.36868	792.0559
b	2.99278	-0.408994	0.088762	3.73141	1.98764	2.36868	707.3186
c	3.00346	-0.408994	0.088762	3.73141	2.70085	2.36868	849.9605
d	2.41384	-0.408994	0.088762	3.73141	1.73632	2.36868	656.7157
e	2.99278	-0.408994	0.088762	3.73141	2.65957	2.36868	841.7047
f	2.99278	-0.408994	0.088762	3.73141	2.00658	2.36868	710.7683

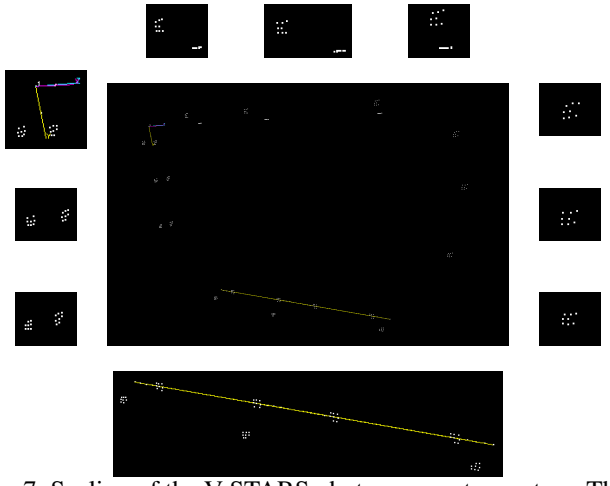
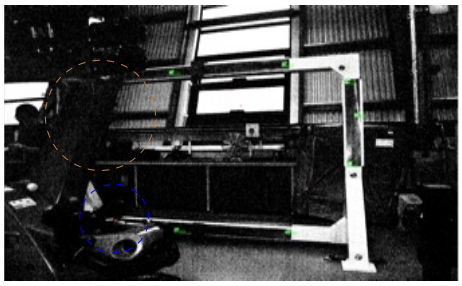
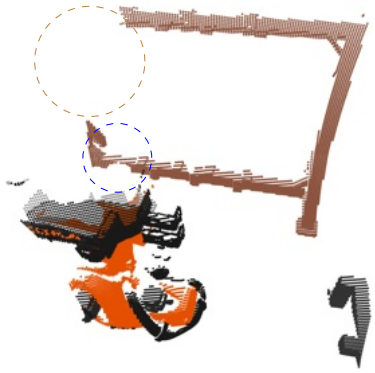


Fig. 7: Scaling of the V-STARS photogrammetry system. The V-STARS must be calibrated before measuring. Hence, the scale bar is used for calibration as indicated in red dashed block.



(a)



(b)

Fig. 8: FoV comparison between physical side and its digital twin. (a) FoV obtained from the V-STARS system located in th work cell. (b) FoV derived from the virtual V-STARS system in digital twin.

A. *FoV Comparison*

The FoV of the V-STARS DynaMo D12 camera is $72deg. \times 58deg.$ by default and can provide highly accurate 3D measurements (10 $\mu m/m$ within 30 m). To obtain the spatial information during assembly, two D12 cameras are employed in the photogrammetry system, where two-dimensional images are transformed into 3D measurement using re-section and triangulation.

In Fig. 7, the identified retro-reflective targets on the frame are indicated in green. A scale bar is also presented as a scaling reference as shown in Fig. 7 in the red rectangle. In addition, the frame is captured by the D12 camera in real world as shown in Fig. 8 (a). Meanwhile, at the same position in the digital twin, the frame point cloud is scanned as presented in Fig. 8 (b). Both are obtained with identical configuration, where the FoV parameters and the tilt angle are the same.

As given in Fig. 8, the point cloud in Fig. 8 (b) directly indicates the blinded areas of the frame as shown in brown and blue dashed circle, compared with the image captured in Fig. 8 (a). The V-STARS photogrammetry system aims to create a stereoscopic vision of the target frame in a three-dimensional space, while in our work, the point cloud obtained from the virtual V-STARS model with the same FoV is used for evaluating the target visibility. Hence, by comparing FoV visibility between the virtual environment and the physical side, the digital twin model of the V-STARS system can effectively represent its functionalities in the real world.

B. *Lifecycle Single Episode Learning*

Given that the digital twin accompanies the physical facilities during the entire production horizon, the camera position of the photogrammetry system layout should be considered for a lifecycle optimisation. Hence, the single episode lifecycle learning is detailed in this subsection.

The task of this work cell is to pick up three profile boards and assemble them to the frame. The measuring algorithm for V-STARS photogrammetry system is based on bundle adjustment, which is refining 3D geometry coordinates, relative motion parameters, and camera optical characteristics simultaneously given a set of images containing co-visible features. More specifically, bundle adjustment aims to minimise the reprojection errors between the observed image location and predicted image point using nonlinear least-square algorithm. Nevertheless, since minimising reprojection error is a maximum likelihood estimation, the nonlinear least-square algorithm could be overfitting, if the FoV coverage

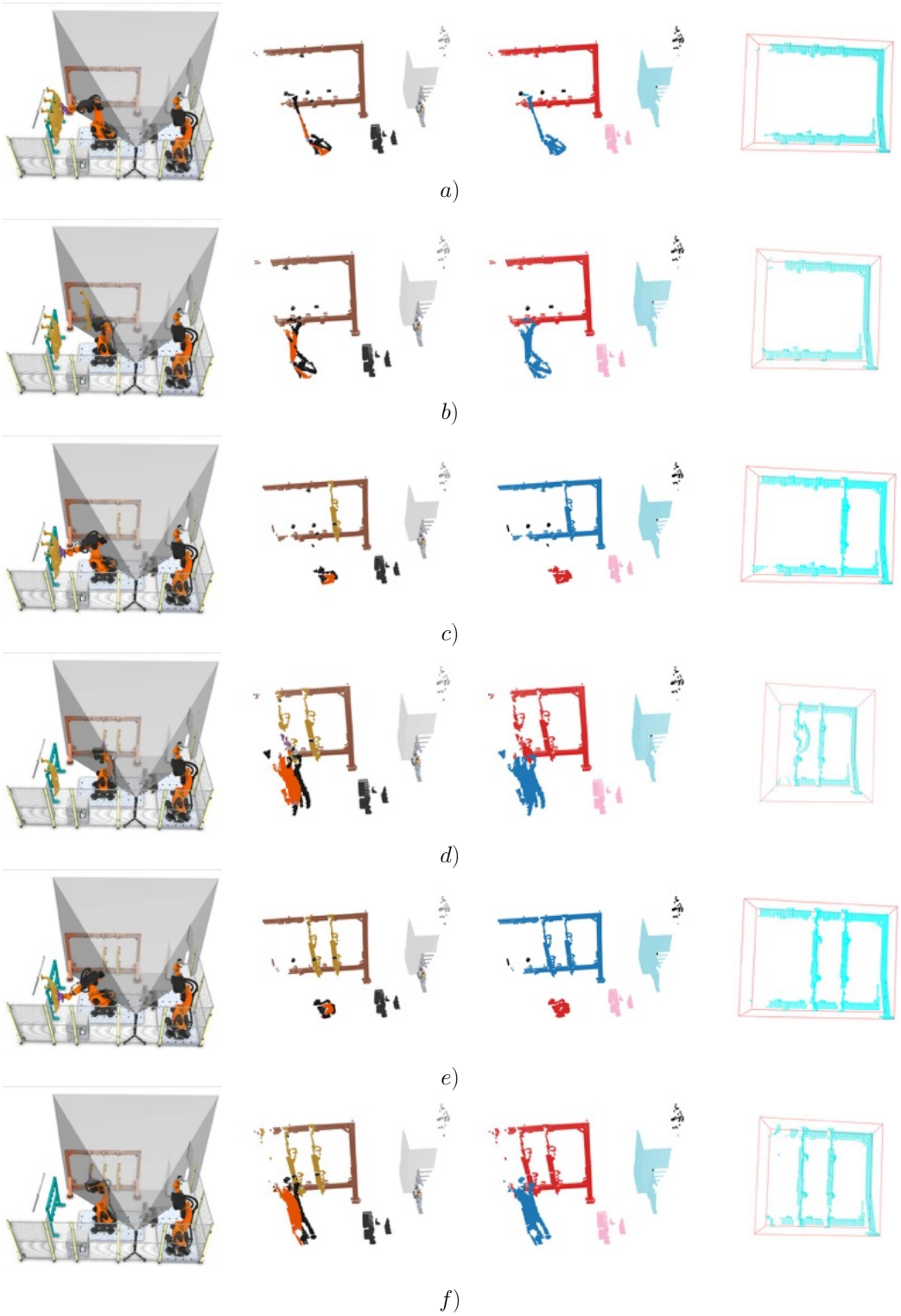
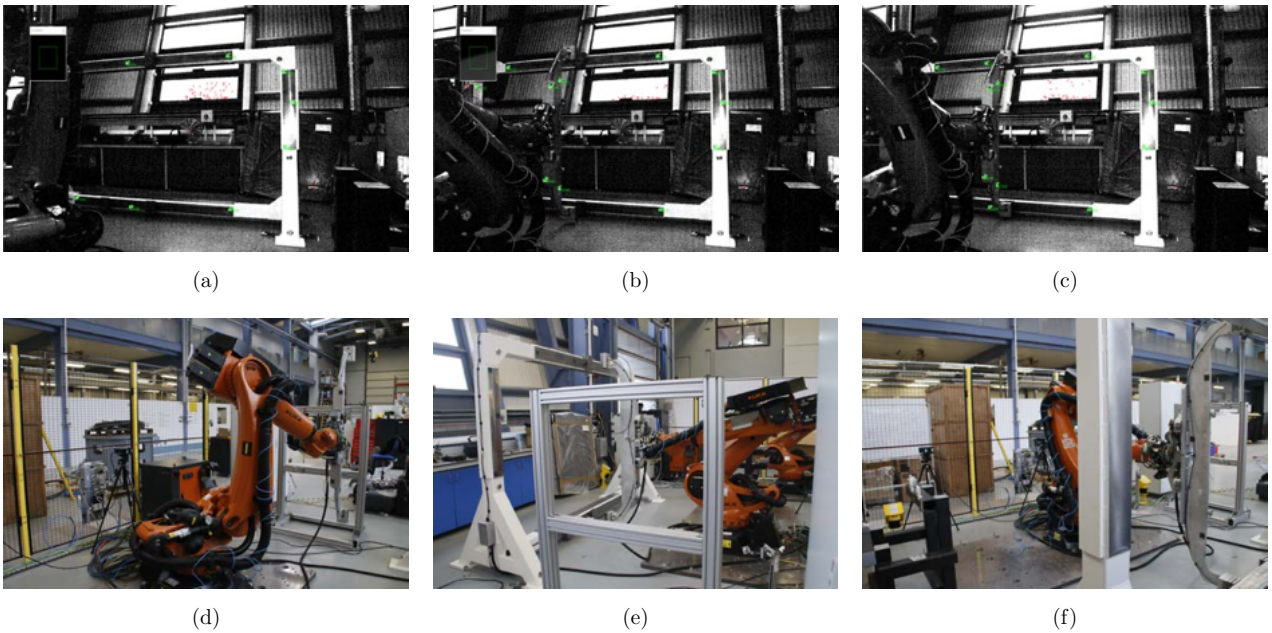


Fig. 9: Operation cycle experiment of a single episode learning. One operation cycle consists of six FoV evaluations regarding different robot poses in a single episode learning.

is too small. Therefore, as presented in Fig. 7, the larger the FoV covers the frame, the more retro-reflective markers are captured by the V-STARS system, which would lead to more robust and accurate measurements.

Real-time data can be captured by VSTARS M-Mode measurement. However, within the assembly processes, and in the layout planning phase, real-time measurement is often not required. Instead, six key measurement steps within the robotic operation is considered as shown in Fig. 10. In addition, exact robot path can be replicated and predicted



(a) (b) (c)

(d) (e) (f)

Fig. 10: Real-world experiment with optimised V-STARS photogrammetry system. The picking and placing profile board is implemented as shown in Fig. 10. (a) Targets captured (in green) while profile board picking. (b) (c) Targets captured (in green) while profile board placing. (d) Profile board picking regard Fig. 10 (a). (e) (f) Profile board placing corresponding to Fig. 10 (b) and Fig. 10 (c), respectively.

in the digital twin with VRC (Virtual Robot Controller) or add-on RCS (Robot Controller Simulation) module. So that all possible robot poses within the production cycle are simulated. Evidently, the manipulator blocks frame visibility while assembling profile boards. In each row of Fig. 10, the learning procedure includes V-STARS relocation, point cloud collection, frame identification and coarse-to-fine registration as detailed in Section IV. If one wishes to facilitate a near real-time application, multiple snapshots at the rate of controller or camera cycle-time can be considered within the proposed FoV evaluation method with computational power balancing between downsampling and optimisation exploration steps.

As shown in Fig. 9, due to the profile board pick-and-place operation, the robot poses lead to various FoV coverages of the photogrammetry system as given in the second column in Fig. 9. Correspondingly, the clusters are identified in the third column. The evaluation result of each target frame is given in the forth column. For example, while picking the third profile board from the stand as given in Fig. 9 (e), the target frame visibility is much larger than the one in Fig. 9 (b), where the manipulator assembles the first profile board to the frame.

The single episode learning result is presented in Table I as indicated by the number in the first column. During the production cycle, different robot assembly positions result in different FoV coverages of the V-STARS system. The largest target frame visibility is shown in row (c) while the manipulator is picking up the second profile board. In contrast, the minimum is indicated in the row (d) when the robot is assembling the second profile board. Therefore, in order to find an optimal position of the V-STARS system in the whole production horizon, the entire process and possible manipulator poses are required to be investigated.

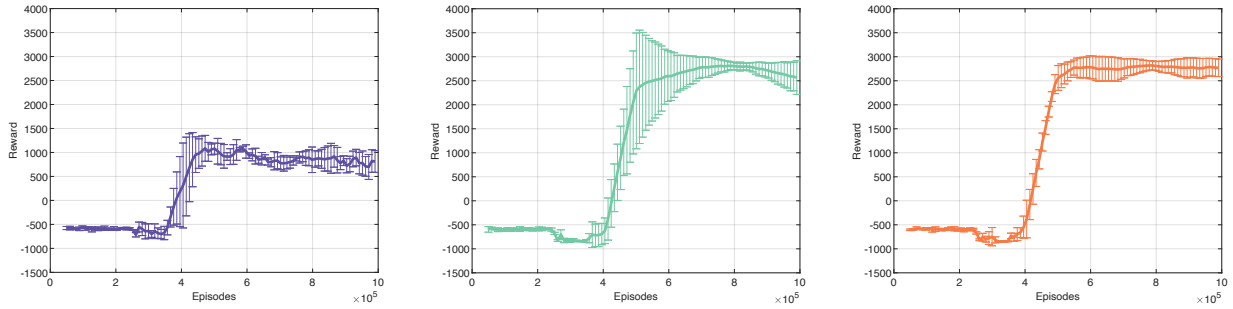
C. Deep Camera Position Optimisation of the V-STARS Photogrammetry System

In this subsection, the position of the V-STARS system is optimised using deep Q-learning. The camera tripod is constrained in an area as shown in the two-dimensional projection of the whole assembly work cell, given in Fig. 13. Besides the reward scheme presented in Section IV, the V-STARS cameras in the digital twin are given additional punishment, if their locations are out of the predefined area.

Regarding the technique aspect, the Visual Components environment class for V-STARS system layout optimisation created in this work is inherited from the environment class in the Python package Stable-Baselines3 [44], which contains a number of highly integrated implementations of reinforcement learning algorithms in PyTorch. Moreover, the reward of each episode is calculated from the point cloud collection after carrying out the step function [44] with an action (V-STARS relocation) as shown in Section V-B. Finally, to accelerate the computational process, NVIDIA GeForce GPU (GTX 1080) is used to accelerate the deep Q-learning optimisation. In terms of the hyperparameters, there are factors such as, learning rate (0.0001), discount factor(0.99), and final value of random action probability (0.01). The learning rate determines how much the neural network learns in each iteration. The discount factor calculates the future discounted reward. These two parameters define the camera position optimisation model. The parameter selection is derived from

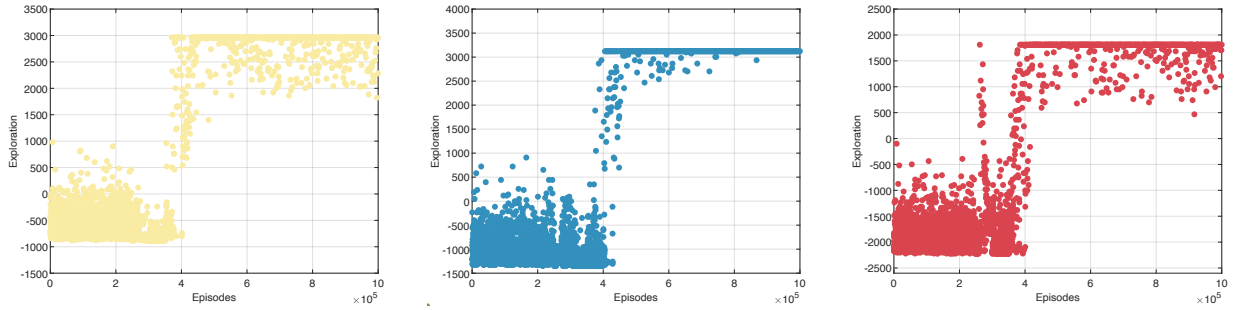
TABLE II: FoV evaluation comparison between the optimal position and the other positions corresponding to Fig. 13.

Num.	a	b	c	d	e	f	Optimal
x	7170.875	7770.875	8370.875	7170.875	7770.875	8370.875	7951.232
y	1903.185	1903.185	1903.185	1503.185	1503.185	1503.185	1619.778
FoV Eval	467.14	1265.26	998.20	635.74	1260.87	1173.20	1308.22



(a) Final value, 0.5. (b) Final value, 0.05. (c) Final value, 0.01.

Fig. 11: Optimising performance comparison regarding the different final values of random action probability.



(a) Initial state P_1 (6669.232, 2070.778) (b) Initial state P_2 (8672.273, 1364.518) (c) Initial state P_3 (8672.443, 2070.860)

Fig. 12: Camera position optimisation for V-STARS photogrammetry system regarding the different initial states.

the original DQN code in [45]. However, the final value (epsilon) will be discussed and balanced for efficiency and implementation practicality in the following experiment.

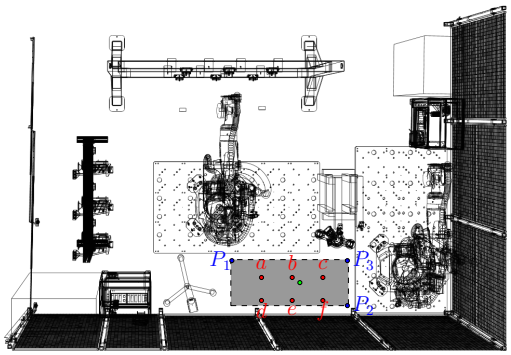


Fig. 13: Comparison between the optimal position and the other positions. The optimal position learned from the deep Q-learning is shown in green. In contrast, six more positions are presented in red dots.

As the epsilon-greedy strategy is used in deep Q-learning, choosing a proper final value of random action probability is crucial in deep Q-learning. In Fig. 11, the whole learning episode is set to 10×10^5 and three different final values (epsilon) for the camera position optimisation are verified. For each final value, we test five times and presented their means and standard deviations in Fig. 11a, Fig. 11b, and Fig. 11c, respectively.

In Fig. 11a, although the learning process converges at around 4.5×10^5, the maximised reward is about 1000, which is much lower compared with the other two final values. In Fig. 11b, we choose a smaller final value (0.05) of random action probability. Despite the fact that the maximised reward is increased, it starts to decay again after convergence. As given in Fig. 11c, the learning process converges at about 6×10^5-th episode, which is slower than the convergence given in Fig. 11a. However, the best reward is much larger and it doesn't decay after convergence. Therefore, by using the trained deep Q network model in Fig. 11c, the optimal position is obtained from the learning history and it is presented in Fig. 13 as shown in green dot, at the position of $(7951.232mm, 1619.778mm)$. Compared to the other V-STARS positions in red dots as shown in Table II, the green dot position has the most FoV evaluation rewards.

In addition, the optimal solution in the physical layout

is tested. Before the implementation, the target environment (also known as the driver file) is already established, where several images from different locations and camera orientations were used. The V-STARS system is set to the single camera mode (S-Mode), which allows a single camera to be triggered at discrete events to obtain target information. During the assembly processes, the V-STARS system can operate in S-mode, as well as the multi-camera mode (M-mode), which allows cameras to take pictures continuously and monitor featured objects in real-time. Each camera in M-mode requires maximum target frame visibility within its FoV, and the proposed camera positioning optimization framework is still applicable. In Fig. 10, the overall assembly process of a profile board is presented. The images captured for picking and placing are given in Fig. 10(a)-(c). Correspondingly, the side view of the assembly process is shown in Fig. 10 (d)-(f). Note that, in Fig. 10 (b) and Fig. 10 (e), the manipulator holding the profile board is approaching the assembly position.

TABLE III: Optimised camera positions corresponding to the different initial states

Initial state	P_1	P_2	P_3
Opt. Position x	7951.232	7996.443	7987.273
Opt. Position y	1620.778	1627.860	1641.518
FoV Eval.	1308.22	1310.77	1297.63

In order to test the impact of initial states on the performance of the proposed camera position optimisation framework, an additional experiment is implemented as shown in Fig. 12. Three initial states are chosen from the box area as shown in Fig. 13 (blue dots). The final value of random action probability is defined as 0.01 according to the previous experimental result. As presented in Fig. 12b, it shows that the camera position learning process converged faster compared with the convergence performance in Fig. 12a and Fig. 12b. However, in Fig. 12c, the optimal solution appears earlier than the others. Another observation is that three camera optimisation processes do not converge to the exact same location, and the maximum deviation between solutions is 45.928mm according to Table III. The deviation is within the exploration step set for the optimisation process, which is at 50mm per step. While reducing the exploration step can bring a smaller solution deviation, it can cause the exploration to be stuck in the collision zone, and fail to reach to a solution. More importantly, it becomes less practical to implement manually. From a practical point of view, the camera placement would be not tracked and monitored. Therefore a larger placement tolerance should be allowed. The maximum solution deviation is close to the actual implementation tolerance one would expect with manual camera placement. If comparing the FoV evaluations of three learning processes in Table III (third row), there is less than 1% difference observed. Hence, one can conclude that the initial states have a neglectable impact on the optimisation process.

VI. Conclusion

In this paper we propose a novel deep dynamic camera position optimisation framework for photogrammetry system of a reconfigurable manufacturing assembly cell. The optimisation framework is based on the digital twin of a robotic assembly work cell. In addition, the camera FoV for the frame visibility as well as collision detection are taken into consideration for the camera position optimisation during the dynamic configuration process. The deep Q-learning algorithm is used to automatically optimise the camera position given the penalties of FoV visibility evaluation and collision detection. In the experiment, feasibility of the virtual camera model in stead of its physical side for FoV evaluation is verified, following by the lifecycle single episode learning. The camera position is optimised and demonstrated in real-world implementation, in which the maximum assembly deviation magnitude of 0.073mm can be achieved.

Without the digital model, for V-STARS photogrammetry system configuration, at least twelve images should be manually taken and chosen for establishing the measuring environment. In addition, during the implementation, operators should guarantee that the FoV is not blocked throughout the entire manufacturing process. For safety, operators should also make sure that there is no collision between photogrammetry devices and other components in the cells. To provide decent FoV coverage, the above-mentioned configuration procedure should be repeated several times. By taking advantages of digital twin techniques, the photogrammetry camera position optimisation can be implemented in a virtual environment which replicates the physical facilities during the whole product lifecycle. This could significantly simplify the configuration processes and avoid manual work for scaling, recalibration and camera position allocation.

The approach is generic and suitable for a range of applications. Firstly, it can be extended to other vision system applications that requires quality FoVs. Secondly, most assembly processes consist of fixtures with simple geometry due to the ease of fabrication, therefore the proposed bounding volume calculation can be easily modified, and still be effective. Thirdly, uncoupled position optimisation ensuring maximum visibility of each camera would be applicable if using more than one camera. In addition, near-real-time application is possible with appropriate downsampling and iterative steps. Lastly, upon changes within a reconfigurable manufacturing environment, the existing assets rearrange and relocate instead of a complete rebuild, therefore, the only update required is the positional layout, which includes the camera position optimisation process.

Evaluating actual camera visibility in the physical world is another challenge, since the camera recognise retro-reflective markers, the marker arrangement on the target object would become the main uncertainty in FoV evaluation. The proposed framework aims to maximise the object visibility as a whole in its FoV. However, to make the algorithm more lean, research in mapping of good/bad area for marker allocation on the target object, and how that would tie into

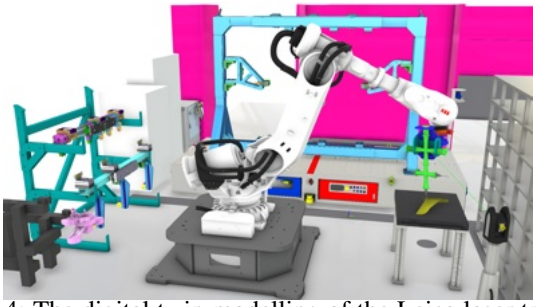


Fig. 14: The digital twin modelling of the Leica laser tracker in VC. If the laser is blocked, the line will be shown in red, otherwise it will be in green.

the proposed optimisation framework are being recognised and under development. Besides photogrammetry systems, there are other metrology devices such as laser radars (point cloud, similar to this work) and laser trackers (as presented in Fig. 14). Currently, we are focusing on addressing the position optimisation of all three typical metrology devices together based on our novel designed Omnifactory (please see our official website https://www.omnifactory.com/)

ACKNOWLEDGMENTS

The authors would like to acknowledge the funding support from Innovate UK project ELCAT (ref 113235), Made Smarter Innovation - Research Centre for Connected Factories (ref EP/V062123/1) and GKN Aerospace, and for the technical support from Nicolas Tanala.

REFERENCES

[1] J. Schmitt, A. Hillenbrand, P. Kranz, and T. Kaupp, "Assisted human-robot-interaction for industrial assembly: Application of spatial augmented reality (sar) for collaborative assembly tasks," in *Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, 2021, pp. 52–56.

[2] Y. Koren, U. Heisel, F. Jovane, T. Moriwaki, G. Pritschow, G. Ulsoy, and H. Van Brussel, "Reconfigurable manufacturing systems," *CIRP annals*, vol. 48, no. 2, pp. 527–540, 1999.

[3] H. Guo, M. Chen, K. Mohamed, T. Qu, S. Wang, and J. Li, "A digital twin-based flexible cellular manufacturing for optimization of air conditioner line," *Journal of Manufacturing Systems*, vol. 58, pp. 65–78, 2021.

[4] P. Pérez-Gosende, J. Mula, and M. Díaz-Madroñero, "Facility layout planning. an extended literature review," *International Journal of Production Research*, vol. 59, no. 12, pp. 3777–3816, 2021.

[5] M. Bortolini, F. G. Galizia, and C. Mora, "Reconfigurable manufacturing systems: Literature review and research trend," *Journal of manufacturing systems*, vol. 49, pp. 93–106, 2018.

[6] A. Montalto, S. Graziosi, M. Bordegoni, L. Di Landro, and M. J. L. Van Tooren, "An approach to design reconfigurable manufacturing tools to manage product variability: the mass customisation of eyewear," *Journal of Intelligent Manufacturing*, vol. 31, no. 1, pp. 87–102, 2020.

[7] A. Drouot, R. Zhao, L. Irving, D. Sanderson, and S. Ratchev, "Measurement assisted assembly for high accuracy aerospace manufacturing," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 393–398, 2018.

[8] K. Rakesh, B. Satyanarayana, and M. sai Bollu, "Manufacturing of complex components using photogrammetry in association with additive manufacturing," *Materials Today: Proceedings*, 2021.

[9] S. Grazioso, T. Caporaso, M. Selvaggio, D. Panariello, R. Ruggiero, and G. Di Gironimo, "Using photogrammetric 3d body reconstruction for the design of patient–tailored assistive devices," in *2019 II Workshop on Metrology for Industry 4.0 and IoT (MetroInd4. 0&IoT)*. IEEE, 2019, pp. 240–242.

[10] Z. Geng and B. Bidanda, "Tolerance estimation and metrology for reverse engineering based remanufacturing systems," *International Journal of Production Research*, pp. 1–14, 2021.

[11] A. Filion, A. Joubair, A. S. Tahan, and I. A. Bonev, "Robot calibration using a portable photogrammetry system," *Robotics and Computer-Integrated Manufacturing*, vol. 49, pp. 77–87, 2018.

[12] P. Maropoulos, J. Muelaner, M. Summers, and O. Martin, "A new paradigm in large-scale assembly—research priorities in measurement assisted assembly," *The International Journal of Advanced Manufacturing Technology*, vol. 70, no. 1-4, pp. 621–633, 2014.

[13] L. Gordon, "Portable laser tracker measures large volumes accurately," *Machine Design*, vol. 83, no. 14, pp. 55–55, 2011.

[14] J. Brown, "V-stars/sa cceptance t est r esults," in *Seattle: Boeing Large Scale Optical Metrology Seminar*, 1998.

[15] E. Nocerino, F. Menna, and G. Verhoeven, "Good vibrations? how image stabilisation influences photogrammetry," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 46, pp. 395–400, 2022.

[16] W. Karel, "Integrated range camera calibration using image sequences from hand-held operation," *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci*, vol. 37, pp. 945–951, 2008.

[17] Y. Zhong, Z. Wang, A. V. Yalamanchili, A. Yadav, B. R. Srivatsa, S. Saripalli, and S. T. Bukkapatnam, "Image-based flight control of unmanned aerial vehicles (uavs) for material handling in custom manufacturing," *Journal of Manufacturing Systems*, vol. 56, pp. 615–621, 2020.

[18] W. Płowucha, W. Jakubiec, and M. Wojtyła, "Possibilities of cmm software to support proper geometrical product verification," *Procedia CIRP*, vol. 43, pp. 303–308, 2016.

[19] K. A. Tarabanis, P. K. Allen, and R. Y. Tsai, "A survey of sensor planning in computer vision," *IEEE transactions on Robotics and Automation*, vol. 11, no. 1, pp. 86–104, 1995.

[20] N. R. KINEMATICS, "Spatial analyzer user manual," *Williamsburg, VA, USA*, 2013.

[21] R. Xia, M. Hu, J. Zhao, S. Chen, and Y. Chen, "Global calibration of multi-cameras with non-overlapping fields of view based on photogrammetry and reconfigurable target," *Measurement Science and Technology*, vol. 29, no. 6, p. 065005, 2018.

[22] S. Gai, F. Da, and M. Tang, "A flexible multi-view calibration and 3d measurement method based on digital fringe projection," *Measurement Science and Technology*, vol. 30, no. 2, p. 025203, 2019.

[23] M. S. Suresh, A. Narayanan, and V. Menon, "Maximizing camera coverage in multicamera surveillance networks," *IEEE Sensors Journal*, vol. 20, no. 17, pp. 10 170–10 178, 2020.

[24] H. Zhang, J. Eastwood, M. Isa, D. Sims-Waterhouse, R. Leach, and S. Piano, "Optimisation of camera positions for optical coordinate measurement based on visible point analysis," *Precision Engineering*, vol. 67, pp. 178–188, 2021.

[25] L. Doulos, A. Tsangrassoulis, and F. Topalis, "Multi-criteria decision analysis to select the optimum position and proper field of view of a photosensor," *Energy conversion and management*, vol. 86, pp. 1069–1077, 2014.

[26] J. Li, J. Berglund, F. Auris, A. Hanna, J. Vallhagen, and K. Åkesson, "Evaluation of photogrammetry for use in industrial production systems," in *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2018, pp. 414–420.

[27] T. Lowder, "Use of 3d scanning for manufacturing layout redesigns," Master's thesis, NTNU, 2020.

[28] F. Agueera-Vega, F. Carvajal-Ramirez, P. Martinez-Carricondo, J. S.-H. López, F. J. Mesas-Carrascosa, A. Garcia-Ferrer, and F. J. Pérez-Porras, "Reconstruction of extreme topography from uav structure from motion photogrammetry," *Measurement*, vol. 121, pp. 127–138, 2018.

[29] A. Paixão, R. Resende, and E. Fortunato, "Photogrammetry for digital reconstruction of railway ballast particles–a cost-efficient method," *Construction and Building Materials*, vol. 191, pp. 963–976, 2018.

[30] S. Zhao, F. Kang, J. Li, and C. Ma, "Structural health monitoring and inspection of dams based on uav photogrammetry with image 3d reconstruction," *Automation in Construction*, vol. 130, p. 103832, 2021.

[31] (2021) Siemens tecnomatix. [Online]. Available: https://www.plm.automation.siemens.com/global/en/products/tecnomatix/

[32] (2021) Gazebo. [Online]. Available: http://gazebosim.org/

[33] (2021) Visual components - 3d manufacturing simulation software. [Online]. Available: https://www.visualcomponents.com/

[34] Z. Wang, P. Kendall, K. Gumma, A. Smith, A. Turner, and S. Ratchev, "Development of an affordable and auto-reconfigurable solution for small box assembly," *IFAC-PapersOnLine*, vol. 55, no. 10, pp. 2463–2468, 2022.

[35] Z. Wang, P. Kendall, K. Gumma, A. Turner, and S. Ratchev, "An adaptive, repeatable and rapid auto-reconfiguration process in a smart manufacturing system for small box assembly," in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2022, pp. 1472–1478.

[36] D. Birant and A. Kut, "St-dbscan: An algorithm for clustering spatial–temporal data," *Data & knowledge engineering*, vol. 60, no. 1, pp. 208–221, 2007.

[37] G. Xuan, W. Zhang, and P. Chai, "Em algorithms of gaussian mixture model and hidden markov model," in *Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205)*, vol. 1. IEEE, 2001, pp. 145–148.

[38] Q.-Y. Zhou, J. Park, and V. Koltun, "Fast global registration," in *European conference on computer vision*. Springer, 2016, pp. 766–782.

[39] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *2009 IEEE international conference on robotics and automation*. IEEE, 2009, pp. 3212–3217.

[40] M. Greenspan and M. Yurick, "Approximate kd tree search for efficient icp," in *Fourth International Conference on 3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings.* IEEE, 2003, pp. 442–448.

[41] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[42] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[43] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[44] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, 2021.

[45] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.