BubCam: A Vision System for Automated Quality Inspection at Manufacturing Lines

Jiale Chen
HP-NTU Digital Manufacturing Corporate Lab
Nanyang Technological University, Singapore

Duc Van Le
HP-NTU Digital Manufacturing Corporate Lab
Nanyang Technological University, Singapore

Rui Tan
School of Computer Science and Engineering
Nanyang Technological University, Singapore

Daren Ho
HP Inc., Singapore

ABSTRACT

Visual sensing has been widely adopted for quality inspection in production processes. This paper presents the design and implementation of a smart collaborative camera system, called *BubCam*, for automated quality inspection of manufactured ink bags in Hewlett-Packard (HP) Inc.'s factories. Specifically, BubCam estimates the volume of air bubbles in an ink bag, which may affect the printing quality. The design of BubCam faces challenges due to the dynamic ambient light reflection, motion blur effect, and data labeling difficulty. As a starting point, we design a single-camera system which leverages various deep learning based image segmentation and depth fusion techniques. New data labeling and training approaches are proposed to utilize prior knowledge of the production system for training the segmentation model with a small dataset. Then, we design a multi-camera system which additionally deploys multiple wireless cameras to achieve better accuracy via multi-view sensing. To save power of the wireless cameras, we formulate a configuration adaptation problem and develop a deep reinforcement learning (DRL)-based solution to adjust each wireless camera's operation mode and frame rate in response to the changes of presence of air bubbles and light reflection. Extensive evaluation on a lab testbed and real factory trial shows that BubCam outperforms six baseline solutions including the current manual inspection and existing bubble detection and camera configuration adaptation approaches. In particular, BubCam achieves 1.34x accuracy improvement and 260x latency reduction, compared with the manual inspection approach.

CCS CONCEPTS

• **Computer systems organization → Sensor networks**; • **Computing methodologies → Image segmentation**.

KEYWORDS

Product Quality Inspection, Visual Sensing, Deep Learning, Reinforcement Learning

ACM Reference Format:
Jiale Chen, Duc Van Le, Rui Tan, and Daren Ho. 2023. BubCam: A Vision System for Automated Quality Inspection at Manufacturing Lines. In *ACM/IEEE 14th International Conference on Cyber-Physical Systems (with CPS-IoT Week 2023) (ICCPS '23), May 9–12, 2023, San Antonio, TX, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3576841.3585926

1 INTRODUCTION

With the recent advancements of Internet of Things (IoT) and deep learning (DL) techniques in dealing with complex data patterns, visual inspection has been increasingly employed to enable the automated quality control at manufacturing lines [2, 9, 12]. The purpose is to automatically detect any defects in the manufactured products with low latency before the final distribution to the customers [16]. The designs and implementations of such automated inspection systems often require many efforts in dealing with the stringent requirements, practical constraints, and complex environmental conditions of the industrial processes.

In this paper, the target application is the quality inspection of the ink bag products manufactured in Hewlett-Packard (HP) Inc.'s ink factories for large industry printers. A key task is to measure the total volume of air bubbles inside an inspected ink bag. During the production process, the ink-filling machines may inject air bubbles into the ink bags. The ink bag with air that exceeds a certain volume can significantly reduce the printing quality of the printer using the ink bag. Specifically, during the printing process, the ink may absorb air, leading to unstable velocity of the ink droplets [5]. Thus, the quality inspection is needed to ensure that the ink bags with unacceptably large volumes of air are not distributed to the customers.

The current protocol of HP's factories adopts a manual quality inspection procedure which begins with removing the outer plastic layer of the inspected bag and then squeezes the air bubbles to form an air cone in a corner of the bag. Finally, the size of the air cone is manually measured to estimate the volume of air. This manual measurement suffers from low accuracy and high latency. From the historical records, it can take up to 10 minutes to inspect an ink bag. Moreover, it is a costly destructive test due to the removal of the outer plastic layer of the ink bag.

To increase the inspection accuracy and throughput, and reduce the costs, in this paper, we design and implement a smart camera system, called *BubCam*, which uses a fog computing (FC)-assisted wall-powered depth camera and wireless cameras for automated quality inspection of the ink bags. The primary goal of BubCam is to

ICCPS '23, May 9–12, 2023, San Antonio, TX, USA

ACM ISBN 979-8-4007-0036-1/23/05.
https://doi.org/10.1145/3576841.3585926

accurately estimate the volumes of the air inside the manufactured ink bags without manual intervention. However, in this application, the sizes of the air bubbles are at millimeter-level. Our experiments in §7 show that the existing bubble detection approaches using the conventional computer vision (CV)-based [17] and the recent convolutional neural network (CNN)-based [7] object detection algorithms have inferior performance in estimating the volumes of small bubbles. To achieve high accuracy, BubCam employs a DL-based image segmentation model which extracts multiple regions of bubbles in the images for the air volume estimation.

Due to the dynamic environment condition and complex settings of the production lines, the design of BubCam also faces the following three additional challenges. *First*, in the production lines, the ambient light often reflects on the surface of the ink bags. As a result, the air bubbles may be blocked and invisible in the captured RGB images. Moreover, the location of the reflected areas may change over time due to the dynamic lighting condition in the factories. *Second*, the quality of the images can suffer from the motion blur effect caused by the movement of the ink bag on the conveyor belt. *Third*, labeling images of the manufactured ink bags is labor-intensive and time-consuming.

To address the first two challenges, we develop a late fusion scheme which combines the segmentation results of multiple consecutive RGB images to generate the final result of the air volume estimation. Fusion can help improve the estimation accuracy because different frames may have different quality and reflected areas under variations of the light reflection and motion blur conditions. To achieve higher accuracy under the dynamic lighting conditions, the depth sensing information is also fused with the RGB images when the reflection areas are detected in the RGB images. To address the problem of labeling difficulty, we propose a knowledge-based labeling approach which utilizes the prior knowledge of the production to facilitate the labeling process.

Multiple views of a scene are generally robust against the light reflection variation problem. Thus, we further develop a multi-camera BubCam system which uses multiple wireless cameras to assist the main wall-powered camera. The wireless cameras are deployed to capture the inspected ink bag from different view angles. These images are then fused with the images captured by the main camera. Specifically, due to the low cost and easy installation of the wireless cameras, the multi-camera BubCam can be deployed in an *ad hoc* fashion to achieve better accuracy, especially under the dynamic light reflection condition.

However, the wireless cameras are often powered by batteries with finite capacities. Thus, it is desirable to adapt their configuration to minimize the camera's energy consumption while maintaining the system performance in response to the variations of the presence of air bubbles and the lighting condition. For instance, the wireless cameras should be only activated when the bubble areas are blocked by light reflection in the images of the main camera. Furthermore, the capturing frame rate of the wireless cameras should increase when the air bubbles appear in their field of view. Otherwise, the frame rate should be kept minimum to save power. To this end, we propose a novel configuration adaptation approach to maintain desired accuracy with minimum energy consumption. Specifically, we formulate an adaptation problem as a Markov decision process (MDP) which configures the activation mode and image frame rate of the wireless cameras. Then, we apply deep reinforcement learning (DRL) to learn the optimal configuration policy.

We perform extensive evaluation via testbed experiments in both controlled lab and factory trials. Specifically, we collect an image dataset to drive the design of the proposed DL-based image processing pipeline and DRL-based configuration adaptation approach. We compare BubCam with six baseline approaches which include the factories' manual inspection, three bubble detection and two configuration adaptation baseline approaches. The evaluation results show that BubCam achieves accuracy improvement of about 1.34x and latency reduction of up to 260x, compared with the factories' manual inspection.

The contributions of this work can be summarized as follows:

- We design and implement BubCam which applies the DL-based image segmentation and fusion techniques to accurately estimate the volume of the air bubbles in the ink bags on the HP's production lines. BubCam addresses challenges caused by combined impacts of the motion blur and strong reflection conditions in the industrial environment.
- We formulate the camera configuration adaptation problem and propose a DRL-based approach to learn the efficient configuration policy in the industrial settings. The DRL-based approach helps reduce the wireless camera's energy consumption while maintaining the system performance.
- We set up real testbeds in both a controlled system and the factories' manufacturing lines to evaluate BubCam. Effectiveness of BubCam is compared with six baseline approaches. The design of BubCam can be useful to the developments of other vision systems for the automated quality inspection of the relevant products in the industrial processes.

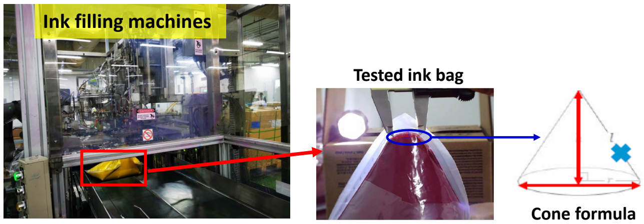
Paper organization: §2 reviews related work. §3 describes background and approach overview. §4 overviews the design of BubCam. §5 and §6 present the detailed design of the single-camera and multi-camera BubCam systems, respectively. §7 presents the evaluation results. §8 concludes the paper.

2 RELATED WORK

In this section, we review the related works on industrial visual inspection, bubble detection, and multi-camera systems.

■ **Industrial visual inspection:** Visual inspection is widely adopted to inspect the quality of manufactured products in factories. For instance, the study in [4] developed a visual sensing system which aims to rectify millimeter-level edge deviation during production at the factory. The developed system can help increase the material utilization rate and save manpower. Moreover, in [12], the authors designed the fog computing (FC)-assisted camera systems to detect defects of the tile products in real time. Similarly, BubCam is a visual sensing system which aims to inspect the quality of the air bubbles in the ink bags on the production lines.

■ **Bubble detection:** Bubble detection is important in many industrial applications. The existing studies [6, 7, 10, 17] mainly applied object detection algorithms for bubble detection, which can be divided into the following two main categories. The first category consists of the conventional CV-based object detection approaches [10, 17] which generally use the edge detection algorithms



a) An ink bag manufacturing line. b) Manual quality inspection.

Figure 1: An illustration of factories' manual inspection.

to extract the air bubbles from the images. Then, the extracted areas are fit to the geometry shapes (e.g., circles or ellipse) whose size is considered as the bubble volume. For instance, the study in [17] adopted a hough circle detection algorithm to extract the bubble areas from the image. Then, a concentric circular arrangement algorithm was used to determine the circles that best fit to the extracted bubble areas. The second category consists of the DL-based object detection approaches [6, 7]. For instance, the study in [7] developed a region-based CNN (RCNN) model to detect the bounding boxes of the bubbles from the image. Then, a shape regression CNN was adopted to transform the extracted boxes to circles or ellipse whose total number of pixels is considered as the bubble volume. In [6], the authors proposed a generative adversarial network (GAN) model to generate the augmented images for efficiently training the RCNN model with a limited labeled images dataset. However, as shown by our experiments in §7, these existing CV-based and DL-based object detection approaches have inferior performance in estimating the volume of the small bubbles in the inspected ink bags. Thus, BubCam employs a DL-based segmentation model for the bubble volume estimation.

■ **Multi-camera systems:** Existing studies [3, 13, 14, 20] proposed the use of multiple cameras for product quality inspection in the production lines. For instance, the study in [3] developed a multi-camera system to inspect defects on the stereo skeleton of the car. Specifically, multiple cameras are deployed to cover different parts of the car for detecting the tiny defects in real time. In [14], the authors designed a multi-camera system to achieve a highly accurate 3D profile measurement. Moreover, the study in [20] proposed a multi-camera system to inspect the quality of the product parts moving on the conveyor belt in the factories. The proposed multi-camera BubCam uses the multiple wireless cameras to improve the bubble volume estimation accuracy under the dynamic ambient light reflection conditions. Similar to [15, 20], we adapt the configuration for the camera's parameters to achieve the desired performance with minimum energy consumption under the time-varying environment conditions.

3 BACKGROUND, MOTIVATION & APPROACH

In this section, we present the background of the ink bag quality inspection at HP's factories. Then, we describe the design approach and challenges of BubCam.

3.1 Background and Motivation

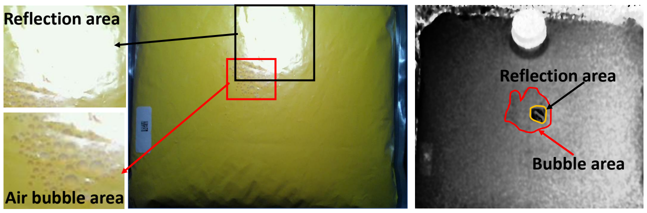
As discussed in §1, the current protocol of the HP's factories adopts a manual quality inspection approach to estimate the volume of the air bubbles inside the manufactured ink bags. Fig. 1 illustrates how the ink bags are inspected in the HP factories. Specifically, the empty ink bags with two plastic layers are filled by the ink-filling machines. Then, the filled ink bags are moved through the conveyor belt to the packaging process. However, the filling machines may inject air into the bags. The air bubbles with a sufficiently large volume can reduce the printing quality of the industrial printers [5]. Thus, the quality inspection is performed to ensure that the ink bags with air exceeding a certain volume are not distributed to the customers and provide feedback to determine the need for corrective actions in the ink-filling process. To achieve the goal, during the production process, the technicians continuously monitor and perform human eye-tracking to examine the ink bags moving on the conveyor belt. The ink bags detected with the air bubbles are taken for a manual measurement procedure which begins with removing the outer plastic layer of the inspected ink bag for better visibility. Then, the air bubbles are manually squeezed to a corner of the bag, which approximately forms an air cone as illustrated in Fig. 1 (b). A backlight is placed behind the ink bag to better visualize the boundary of the formed air cone. Finally, a vernier caliper is used to measure the radius and height of the cone for determining the volume of the air bubbles.

The above manual quality inspection approach has two main drawbacks as follows. First, it is a destructive test because the outer layer of the inspected ink bag is removed for the better visibility of the air bubbles. As such, all inspected ink bags are destructed even they do not have the large air bubbles. Second, it has high latency. From the historical records, the manual inspection procedure including the outer layer removal, bubble squeezing and air cone size measurement can take about 5 minutes to 10 minutes per ink bag. Thus, to avoid the destructive tests and increase the inspection throughput, it is desirable to develop an automated inspection system that can reliably estimate the volumes of the air bubbles in the ink bags with low latencies.

3.2 Design Approach and Challenges

In this work, we design and implement an automated visual quality inspection system, called BubCam, to replace the factories' manual inspection approach. The key design approach of BubCam is to deploy a single wall-powered camera to continuously capture images of the manufactured ink bags moving on the conveyor belt. The camera is deployed at a relatively far distance from the conveyor belt such that a captured image can contain the entire ink bag. Meanwhile, the actual air bubbles often have a size of millimeter-level. As a result, the air bubbles appear with an extremely small size in the captured images. For instance, as shown in an image sample in Fig. 2(a), an air bubble has a size of around 1,200 pixels which only account for 1/768 of the entire 1-megapixel image.

It is nontrivial to accurately estimate the volume of such small bubbles in the captured images. A possible approach to address this problem is to place the camera closer to the ink bags. However, in the production lines, the manufactured ink bags can have three different sizes of 36.5cm × 31cm, 41.5cm × 37cm, and 42cm × 49cm. Multiple cameras are needed to capture the entire ink bags with these large sizes at a short distance. The use of multiple high-quality



(a) RGB image.

(b) Depth image.

Figure 2: Samples of captured RGB and depth images.

wall-powered cameras increases the system cost and complexity. Thus, in BubCam, we use a single wall-powered camera.

Furthermore, due to the buoyancy of the ink and the curved bag surface, the air bubbles often cluster into multiple groups on the bag's internal surface. This observation suggests that the air bubbles in the same group should be detected together. Thus, for BubCam, we develop a DL-based image segmentation approach which segments the air bubbles into multiple image regions for the bubble volume estimation. We also investigate the feasibility of the existing CV-based [17] and CNN-based [7] object detection approaches for our bubble estimation problem. However, from our experiments (cf. §7), these existing approaches have inferior performance, compared with BubCam. The main reason is that these approaches aim to recognize the air bubbles as individual objects. Thus, they fail to accurately detect the small bubbles which are squeezed in multiple groups in the ink bag. Moreover, the DL-based segmentation model generally requires less effort for image labeling than the DL-based object detection approaches. Specifically, determining the ground-truth label of the bubble regions is often easier than determining the ground-truth label of the individual bubbles.

Furthermore, due to the complex production settings and requirements, the design of BubCam also faces the following two additional challenges. First, the bubble volume estimation accuracy suffers from the ambient light reflection problem. Specifically, in the factories, the production systems are often set up in an open space or an enclosed space covered by the transparent frames. The ambient light can reflect on the surface of the ink bags at certain locations which may change over time due to the motion of the ink bags on the conveyor belt and the surrounding industrial objects and humans. As a result, the air bubbles can be blocked by the light reflection on the captured RGB images as illustrated in Fig. 2(a). The blocked bubbles cannot be detected based on the optical visual sensing information only. Second, as the ink bags move on the conveyor belt during the inspection process, the quality of captured images can be low due to the motion blur effect. The low-quality images result in low accuracy in detecting small air bubbles.

To address the above challenges, BubCam fuses multiple consecutive RGB images which may have different quality and reflected areas under the dynamic light reflection and motion blur conditions. Furthermore, we prototype BubCam by a depth camera which can provide both the RGB and depth images. The hardware components of BubCam will be described in §4. Figs. 2(a) and (b) show the RGB and depth images captured by the camera under the light reflection. As shown in Fig. 2, the air bubble areas which are blocked in the RGB frame is still visible in the depth image. Thus, to achieve better accuracy, we fuse the RGB and depth images when a reflected area is detected on the RGB image.

We further develop a multi-camera BubCam system for higher accuracy, especially under the dynamic lighting condition from the ambient environment. The multi-camera BubCam additionally uses multiple wireless cameras to capture the inspected ink bag from different view angles. These images are fused with the images captured by the main camera to generate the final bubble volume estimation result. Specifically, due to the low cost and easy installation of the wireless cameras, the multi-camera BubCam can be deployed in an *ad hoc* fashion for accuracy improvement, especially when the strong light reflection is observed. Moreover, we formulate a camera configuration adaptation problem and propose a DRL-based learning approach to save the camera's battery power.

Lastly, data labeling is a challenging task for developing BubCam. Specifically, labeling process is time-consuming and requires a collaboration with the experienced technicians. Thus, it takes a lot of efforts to create a big labeled image dataset for training and testing the BubCam's DL-based image segmentation model. To address this issue, we design the knowledge-based labeling approach which utilizes the prior knowledge about the motion of the ink bags to facilitate the labeling process.

4 SYSTEM OVERVIEW AND HARDWARE

In this section, we overview the design and hardware of BubCam as illustrated in Fig. 3.

4.1 Overview of Single-Camera System

For the wall-powered camera, we choose L515 [11], an off-the-shelf LiDAR camera which can provide both RGB and depth images with a resolution of 1920 × 1080 pixels and 1024 × 768, respectively, at a frame rate up to 30 frames per second (FPS). Moreover, it has a sensing range of from 0.25 meters to 9 meters. The L515 is connected to a fog node prototyped by a Jetson AGX Xavier unit that is equipped with a 2.03GHz CPU and a mobile 1.2GHz GPU. The images are processed at the fog node for the bubble volume estimation. Specifically, the image processing pipeline of the single-camera BubCam consists of the following three main steps.

■ **Image preprocessing:** As illustrated in Fig. 3, the main camera L515 captures the ink bags moving through its view on the conveyor belt. Per image capturing, the L515 provides both RGB and depth image frames. Upon receiving the captured consecutive RGB frames, the fog node computes the pixel-wise absolute difference between them and then selects the key frames whose pixel difference is higher than a certain threshold. Moreover, a color filtering algorithm is implemented to detect the ink bag and light reflection areas in each key RGB frame. When any reflection area is detected, the corresponding depth frame is also selected. The selected RGB and depth images are fed to the segmentation module.

■ **DL-based image segmentation:** We adopt a deep CNN-based segmentation model, called pyramid scene parsing network (PSPNet) [19] to extract the bubble areas from the images. However, due to the nature of a deep CNN, training the PSPNet often requires a large labeled image dataset to achieve satisfactory inference accuracy. As discussed earlier, determining the ground truth of the bubble areas is nontrivial. Thus, we develop a knowledge-based

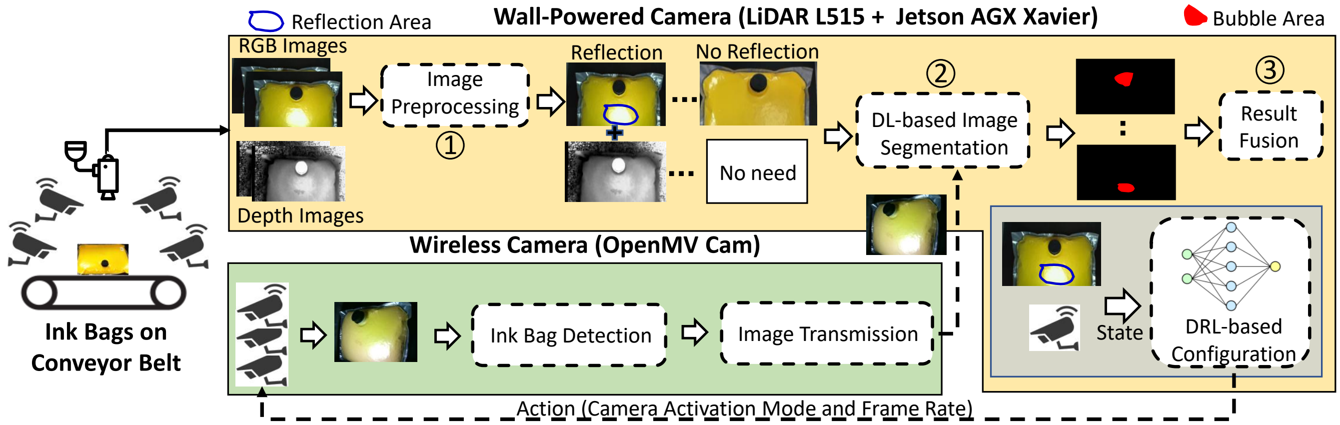


Figure 3: BubCam system overview. A main wall-powered camera is supported by multiple wireless cameras.

image labeling approach which uses the prior knowledge about the motion speed of the ink bags to facilitate the labeling process. We also design a new loss function which utilizes relationship between the bubble volume in the consecutive frames to improve the training accuracy.

■ **Result fusion:** The PSPNet takes an RGB/depth image as input to predict a pixel score map of the image. For the RGB key frame with reflected areas, its score is fused with the score of its corresponding depth frame to generate the final score map of the frame. Then, we consider the total number of pixels with a score greater than a certain threshold as the bubble volume of the frame. Finally, the bubble volumes of all frames of the same ink bag are aggregated to yield the bubble volume of the inspected ink bag.

4.2 Overview of Multi-Camera System

For wireless camera, we choose the lower-power OpenMV H7 Plus that includes an OV5640 module providing an image resolution up to 2592×1944 (5 Megapixels). The OpenMV is equipped with an ARM Cortex M7 480 MHz processor, a 32MB SDRAM and a 1MB SRAM which allow us to implement simple algorithms to preprocess the images locally. Moreover, it is capable of adjusting its frame rate from 0 to 30 FPS.

■ **Local preprocessing:** Due to the movement of the ink bags, the captured images may only contain the conveyor belt. Thus, to reduce the image processing and communication overheard, the wireless camera runs a color filtering algorithm to detect the presence of the ink bag on its images. Then, it only sends the images with the ink bag to the fog node.

■ **DRL-based configuration adaptation:** At the fog node, we implement a DRL-based controller which aims to adapt the configuration for the wireless cameras's parameters in response to variations of the lighting condition and bubble presence. Specifically, the DRL controller periodically observes a system state including the bubble and light reflection presence in the image of the main camera and the residual battery energy levels of the wireless cameras. Then, it selects an action for configuring the activation mode and frame rate of the wireless cameras. The main objective is to maintain the desired accuracy of the bubble volume estimation while maximizing the system lifetime. In §6.2, we formally formulate the configuration adaptation problem and present our DRL solution.

5 DESIGN OF SINGLE-CAMERA BUBCAM

In this section, we describe details of the image preprocessing, segmentation and fusion modules of the single-camera BubCam.

5.1 Image Preprocessing

5.1.1 Key Frame Selection.

The captured consecutive RGB image frames may have similar contents. To reduce the image processing overheard, BubCam selects the key frames only for the image segmentation and fusion. Specifically, among the captured frame, the first frame is always selected as a key frame. Then, the pixel-wise absolute difference between the first and second frames is computed to generate a pixel difference map. The difference map is converted to a gray-scale image. Finally, a median blur algorithm is used to filter out the noises from the gray-scale image.

A pixel is considered as a changed pixel from the first frame to the second frame if its value in the gray-scale image is greater than a threshold value of 80. The ratio of the number of changed pixels to the total number of frame's pixels is used to represent the difference degree between the contents of the two frames. Then, the second frame is selected as a key frame if its difference degree is higher than 20%. Given a new frame, the above processing pipeline is repeated to determine if it is selected as a key frame based on its difference degree, compared with the latest selected key frame.

5.1.2 Ink Bag and Reflection Detection.

BubCam aims to fuse the key frames of the same ink bag to generate the final volume estimation of the bag. Thus, the third preprocessing step is to determine groups of the key frames which contain the same ink bag. Specifically, in the production lines, the ink bags move on the conveyor belt one by one with a certain distance. As a result, the groups of the same bag images are interleaved with the images with the conveyor belt only. Thus, we use presence of the conveyor belt images to determine the images of the same group.

As the color of the ink bags is different from that of the conveyor belt, we implement a color filtering algorithm to detect the presence of the ink bag in the images. Specifically, the algorithm converts

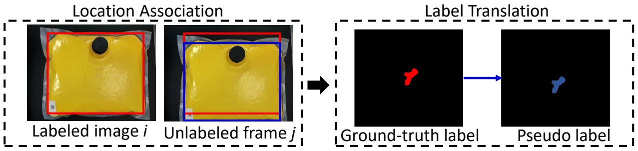


Figure 4: Knowledge-based labeling approach of BubCam.

the RGB image to the HSV (hue, saturation, value) color map. A pixel is considered belonging to the ink bag area if its H, S, V values are within specific ranges which are determined based on the ink bag color. For instance, for the yellow ink bag, we use the H, S, V value ranges of $[20, 40]$, $[40, 255]$, and $[40, 255]$, respectively. The image is considered as a conveyor belt image if no bubble pixel is detected.

We also use the above algorithm to detect the presence of reflection areas on the ink bag to determine the need of depth fusion. Specifically, the color of the reflection areas is white. Thus, we use the H, S, V value ranges of $[222, 256]$, $[0, 256]$, and $[0, 256]$, respectively, to detect pixels of the reflection areas.

5.2 Image Segmentation

5.2.1 Segmentation Model. BubCam adopts the PSPNet [19], a state-of-the-art deep segmentation model which uses a ResNet50 [8] as the backbone CNN model to extract the feature map of the input image. Then, the feature map is forwarded into a pyramid pooling module in which the features are fused to generate a feature representation. Finally, the representation is fed into a convolutional layer to yield a predicted score map.

5.2.2 Knowledge-Based Labeling. To label the images for training the PSPNet, we developed a manual labeling approach which requires collaboration with technicians in the production lines. This manual labeling approach is labor intensive and time-consuming. More details of this approach will be described in §7.1.1. Thus, we design a knowledge-based labeling approach which utilizes the prior knowledge about the motion of the ink bags to label the images based on the ground-truth labels obtained by the manual labeling. Fig. 4 illustrates our knowledge-based labeling approach.

Specifically, in the production lines, the motion speed of the ink bags is known due to the constant rotation speed of the conveyor belt. We consider the consecutive images of the same ink bag that have the same ink bag area size but different locations of the ink bag in the frames. First, we adopt the manual labeling to generate the ground-truth label (i.e., the pixel score map) of a frame (e.g., frame i in Fig. 4) among these frames. Our proposed labeling approach generates a pseudo label of the unlabeled frame j based on the ground-label of frame i. First, we use the color filtering algorithm as described in §5.1.2 to detect bubble pixels in frames i and j. Then, a connected component analysis algorithm [1] is used to extract the ink bag area as a rectangle bounding box from the frames. Due to the constant motion speed of the ink bag, the relative distance between locations of two extracted boxes in frames i and j can be determined. The pseudo label of frame j is generated based on the calculated distance and the ground-truth label of frame i.

5.2.3 Model Training. We train the PSPNet with a loss function, given by: $L = L_{ce} + L_s$, where L_{ce} is the pixel-wise cross-entropy loss function of the PSPNet, and L_s is a new loss function that utilizes the prior system knowledge for improving the accuracy. Specifically, the consecutive images of the same ink bag should have consistent volume of the air bubbles. Thus, the L_s is designed to minimize the difference between the predicted score maps of these frames. Let β denote the number of image training groups, each of which consists of the consecutive images of the same ink bag with the same bubble volume in the training dataset. If we define $\Omega_i = 1, \ldots, |\Omega_i|$ as the set of images in group i, then $L_s = \sum_{i=1}^{\beta} \sum_{j=1}^{|\Omega_i|} \sum_{k=1}^{|\Omega_i|} |s_j - s_k|$, where s_j and s_k are the predicted total score of images j and k in group i, respectively.

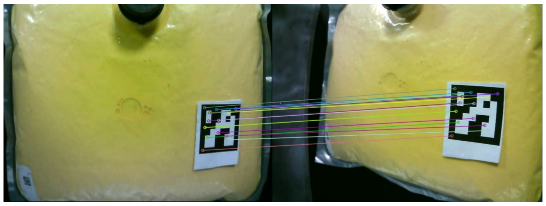


Figure 5: Samples of images with a QR code.

5.3 Result Fusion

The PSPNet predicts a pixel score map for each input key RGB/depth frame. First, the score map of the key RGB frame with the light reflection is fused with the score map of its corresponding depth frame. The L515 of BubCam provides the RGB and depth images with a resolution of 1920×1080 pixels and 1024×768 pixels, respectively. Fusing the entire low-quality depth frame with the high-quality RGB frame can reduce the image quality. Moreover, the purpose of the depth fusion is to achieve better accuracy in detecting the bubbles blocked by the reflection in the RGB image. Thus, we only fuse the score of RGB pixels in the reflected areas with those in the depth image. Specifically, the average fusion is adopted to generate the final score map of the reflected areas in the RGB image. Then, the score map of the fused frame is used to estimate the bubble volume by the total number of pixels with scores higher than a threshold value which is determined based on the ink bag color. For instance, we use a threshold of 0.5 for the yellow ink bags. Finally, the bubble volume of an ink bag is obtained by averaging the volume estimation results of all its key frames.

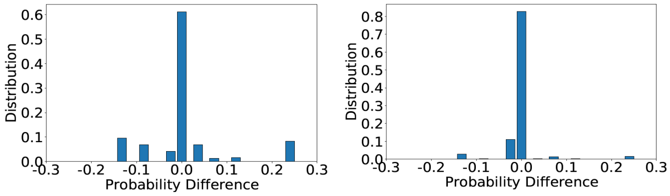
6 DESIGN OF MULTI-CAMERA BUBCAM

In this section, we present the design of the multi-camera BubCam.

6.1 Image Preprocessing

The images captured by the wireless cameras may contain the conveyor belt only. Thus, to save the camera's battery power, the wireless camera should only send the ink bag images to the fog node. To achieve this goal, we use the color filtering algorithm in §5.1.2 to detect the presence of the ink bag in the captured images.

The main and wireless cameras have different view angles. Thus, we adopt a homography projection algorithm to associate the images of the wireless cameras with the images of the main camera. Let H_i denote the 3×3 homography matrix of the wireless camera i. The $P_i = [x_i, y_i, 1]$ denotes the coordinate vector of a pixel p_i in the image of the wireless camera i. Then, the pixel q_m with a



(a) Reflection appearance. **(b) Bubble appearance.**

Figure 6: Compliance with Markov assumption.

coordinate vector, denoted by $Q_m = [x_m, y_m, 1]$, in the image of the main camera is associated with pixel p_i as follows: $Q_m = H_i \times P_i$. The matrix H_i for the wireless camera i is predetermined during the deployment phase. Specifically, we paste a QR code on a surface area of an ink bag. Fig. 5 shows two samples of the ink bag images with the QR code captured by the main and wireless cameras i from different view angles. Then, the quick response (QR) code areas in these two images are extracted and associated to determine the H_i.

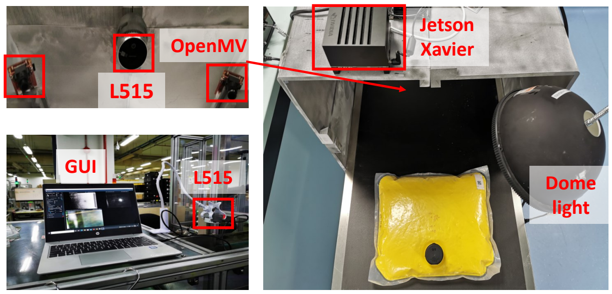
6.2 DRL-based Configuration Adaptation

To increase the battery lifetime, a wireless camera should be activated only when the air bubbles are blocked by the light reflection in the images captured by the main camera. Moreover, the wireless camera should increase its frame rate when the air bubbles appear in its captured images. Otherwise, it can keep a minimum frame rate. To this end, we develop a DRL-based solution to adapt the configuration for the activation mode and frame rate of the wireless cameras in responses to the two exogenous stochastic processes, i.e., the time-varying presence of the bubbles and reflection.

6.2.1 Assessment of Markov Property. Markov assumption (MA) is a basic system property where reinforcement learning is applicable. Thus, we conduct experiments to assess if the above two stochastic processes satisfy the MA, i.e., $\mathbb{P}[X_t|X_{t-1}] = \mathbb{P}[X_t|X_{t-1}, \dots, X_{t-M}]$ with $M \geq 0$, where the X_t represents the reflection/bubble presence at time t. Specifically, we consider the probability difference, denoted by $\Delta P = \mathbb{P}[X_k|X_{k-1}] - \mathbb{P}[X_k|X_{k-1}, \dots, X_{k-M}]$, as an MA compliance metric. A lower absolute value of ΔP indicates better compliance. Fig. 6 shows the distribution of ΔP with $M = 2$ for the reflection and bubble presence in a dataset consisting of 1,000 consecutive images that we captured in the factory. From Fig. 6, we can see that these two stochastic processes have good compliance with the MA because their values of ΔP concentrate at zero. Thus, we formulate the configuration adaptation as an MDP problem and propose a DRL-based solution to learn the optimal adaptation policy.

6.2.2 MDP Formulation. Time is divided into identical intervals of τ seconds, which is referred to as adaptation period. We divide the last image captured by the main camera into a number of equal subareas, denoted by Γ. At the beginning of every adaptation period, called a *time step*, the presence of the air bubbles and light reflection on these subareas is observed to configure the activation mode and frame rate for the wireless cameras. Let N denote the number of wireless cameras in the system.

System state: The system state, denoted by s, is a 3-tuple: $s = (B, L, E)$, where the $B = [b_1, \dots, b_\Gamma | b_i \in \{0, 1\}]$ and $L = [l_1, \dots, l_\Gamma | l_i \in \{0, 1\}]$ represent the presence of the air bubbles and light reflection in the Γ image sub-areas, respectively, end the $E = [e_i, \dots, e_N]$ is a residual energy level vector of N cameras.



(a) Factory deployment. **(b) Lab deployment.**

Figure 7: System deployment at lab and factory

Configuration action: The configuration action, denoted by a, is a vector $a = [\alpha_1, \dots, \alpha_N, f]$, where $\alpha_i \in \{0, 1\}$ is the activation mode of the camera i and the $f \in [f_{min}, f_{max}]$ is the frame rate of all wireless cameras. If the α_i is equal to 1, the camera i is activated. Otherwise, it is set to the sleep mode. Moreover, the f_{min} and f_{max} denote the minimum and maximum frame rates, respectively.

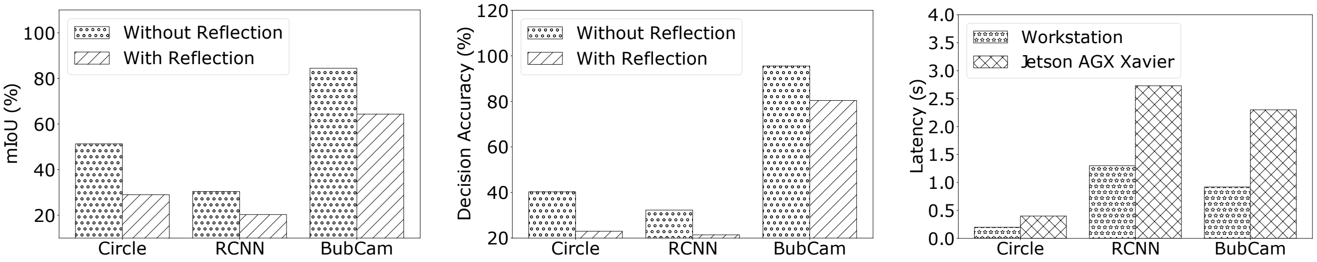
Reward function: Let e_{min} denote the minimum of the remaining energy levels of N wireless cameras at the end of the k^{th} adaptation period and ϕ_{k} denotes the accuracy of the bubble volume estimation of the images captured during the k^{th} period. Then, the immediate reward, denoted by $r(s, a)$, is defined as

$$r(s, a) = \lambda_1 \mathcal{N}(e_{\text{min}}) + \lambda_2 \mathcal{N}(\min\{\phi_k - \phi_{\text{req}}, 0\}), \tag{1}$$

where λ_1 and λ_2 are weights, ϕ_{req} is the required accuracy, and $\mathcal{N} = \max(x, 0)/x_{max}$ represents a normalization process.

The objective of the above MDP problem is to find an optimal adaptation policy that determines action a based on state s to maximize the expected reward over a long run, i.e., $\mathbb{E}[r(s, a)]$. As shown in Eq. (1), the reward $r(s, a)$ is defined based on the weighted sum of the minimum remaining energy and the degree of violating the accuracy requirement. Thus, the optimal policy is to satisfy the accuracy requirement while maximizing the system lifetime which is defined as the operational time of the system until the first wireless camera runs out of battery.

6.2.3 DRL-based Solution. We adopt the learning framework of a DRL algorithm, called the proximal policy optimization (PPO) to learn the optimal configuration adaptation policy. Typically, the DRL agent interacts with the system to learn the optimal policy. However, for the formulated problem, training the DRL agent at run time faces the following two challenges. First, training needs long times to converge, which may lead to large energy consumption of the wireless cameras. Second, it is cumbersome to measure camera's power and the bubble volume accuracy, without the ground-truth labels during the online learning phase. To address these two challenges, we adopt an offline training approach, in which we collect an image dataset and measure the camera's power traces during the collection. The collected images are processed to determine the presence of the light reflection and the air bubbles. Moreover, we use the measured power trace to model the camera's power consumed to capture an image. Then, we use the image dataset and the built power model to drive the offline training of the DRL agent.



(a) mIoU **(b) Decision accuracy** **(c) Latency.**

Figure 8: Comparison with existing bubble detection approaches.

Table 1: Comparison with manual inspection approach.

Approach	Decision accuracy	Latency
Manual inspection	55.4%	5-10 mins
BubCam	74.5%	2.3 seconds

Finally, the trained DRL agent is used to adapt the configuration of the wireless cameras for the online bubble volume estimation.

7 TRIALS AND EVALUATION

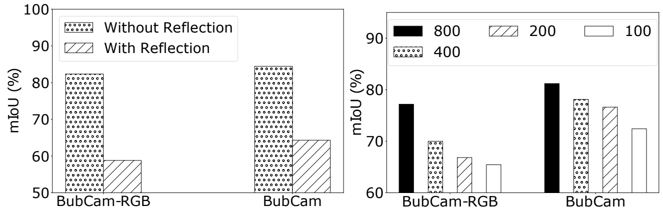
In this section, we present the evaluation results of our trials in factory and lab environments.

7.1 Factory and Lab Trials

7.1.1 Factory Trial. We deploy a multi-camera BubCam's testbed in the HP's production lines to capture images of the manufactured ink bags moving through a conveyor belt as illustrated in Fig. 7(a). We develop a software program based on Labelme [18] to manually label the captured images in collaboration with the HP's product engineers. Specifically, we first use the developed program to manually create a polygon for cropping out the bubble areas. These bubble areas are processed to create the label of the captured images. Then, we work with the engineers to confirm the labeling results. However, this manual labeling process is tedious and extremely time-consuming. Thus, we can create a small dataset of images with confirmed ground-truth labels. Furthermore, we use our proposed knowledge-based labeling approach (cf. §5.2.2) to create more training images based on these confirmed labels.

7.1.2 Lab Trial. We also build a conveyor belt sized $1.5m \times 0.25m \times 0.7m$ as illustrated in Fig. 7(b) to simulate the production line in a lab environment. Specifically, we manually inject different volumes of air bubbles into the ink bags. Then, we attach the cameras and fog node to a 3D-printed frame holder to capture images of the ink bags moving on the conveyor belt. The deployed conveyor belt is capable of adjusting its rotation speed, which allows us to evaluate the impacts of different motion speeds on the image quality and processing accuracy. Moreover, we control the ambient lighting condition in lab to create various light reflection conditions. This simulated system is used to drive the design and conduct controlled experiments to evaluate BubCam.

7.2 Evaluation of Single-Camera BubCam



(a) Comparison with BubCam-RGB. **(b) Impacts of pseudo labels.**

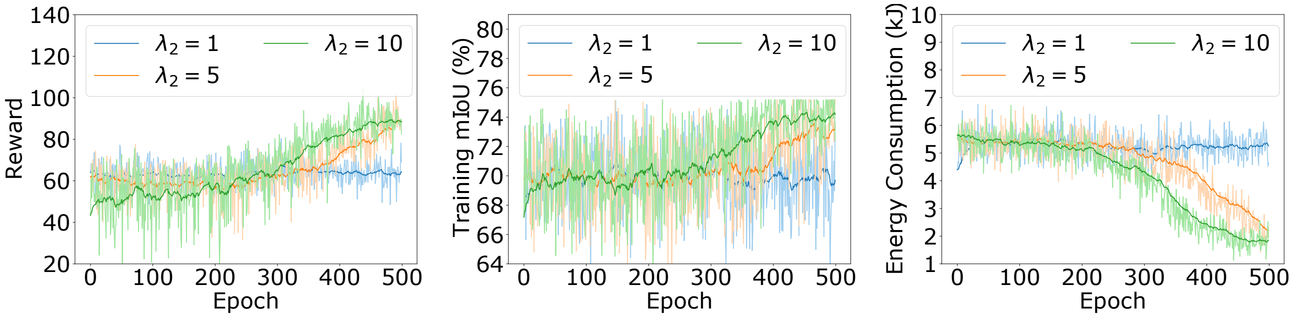
Figure 9: Comparison of BubCam's variants.

7.2.1 Evaluation Settings. We use TensorFlow 2.1 and OpenCV 4.5.3 libraries to build the image processing pipeline and DRL model in Python 3.8. We employ the following three performance metrics. (1) *Mean intersection over union (mIoU)* is the average of ratios of the overlap area to the union area between the ground-truth and predicted bubble areas in all testing images. A higher value of mIoU indicates better segmentation accuracy. *(2) Decision accuracy* is used to evaluate the accuracy in estimating the bubble volume. Let m_{gt} and m_{pred} denote the ground-truth and predicted numbers of the bubble pixels. Then, the volume estimation result of the ink bag is considered accurate if $|m_{\text{pred}-m_{gt}}|/m_{gt} \leq 0.2$. This accuracy metric is used for communication in the factories. *(3) Latency* is the total latency for estimating the bubble volume of an image.

We compare BubCam with the following four baseline approaches. *(1) Manual* is the factories' manual inspection approach (cf. §3.1). *(2) Circle* is a bubble detection approach proposed in [17]. It uses the CV algorithms to detect individual air bubbles as circles. *(3) RCNN* proposed in [7] employs an RCNN model to extract bounding boxes of all individual bubbles in the image. Then, a shape regression CNN is used to transform the extracted boxes into circles whose total number of pixels is considered as the bubble volume. *(4) BubCam-RGB* is a variant of our BubCam which only processes RGB images to estimate the bubble volume without depth fusion.

7.2.2 Evaluation Results. Now, we present evaluation results of the single-camera BubCam and the four baseline approaches.

■ **Comparison with bubble detection approaches:** We evaluate the BubCam and two baseline approaches including the Circle and RCNN based on a dataset of 1,000 RGB images which are collected in our lab's conveyor belt system. In particular, each RGB image has one corresponding depth image. The dataset is divided to the training and testing sets by a ratio of 8:2. Fig. 8 shows the mIoU, decision accuracy and latency of BubCam, Circle and RCNN approaches on a total of 200 testing RGB images which consists of 100 RGB images with/without the light reflection. From Fig. 8, we



(a) Reward. **(b) mIoU.** **(c) Energy consumption.**

Figure 10: DRL Training results.

can see that the presence of the light reflection reduces the image processing performance of three approaches. Moreover, BubCam always achieves the highest mIoU and decision accuracy among the three approaches. For instance, BubCam has mIoU of 64.3% and 84.4%, while RCNN has mIoU of 20.3% and 30.4% in tthe presence and absence of the light reflection, respectively. These results imply that BubCam achieves mIoU improvement of about 2x and 1.56x, compared with RCNN. Furthermore, Fig. 8(c) presents the latency of three approaches in the fog node (i.e., Jetson AGX Xavier) and a workstation with a 3.3GHz CPU, a RTX8000 GPU and a 48GB RAM. From Fig. 8(c), the Circle has the lowest latency since it employs the simple CV algorithms to detect the air bubbles. Differently, BubCam and RCNN have higher latencies due to the use of deep CNN models. However, BubCam can achieve lower latencies than RCNN. Specifically, BubCam can achieve 2.3 seconds latency on the Jetson AGX Xavier. These results demonstrate the superior performance of BubCam, compared with the existing CV-based and CNN-based bubble detection approaches.

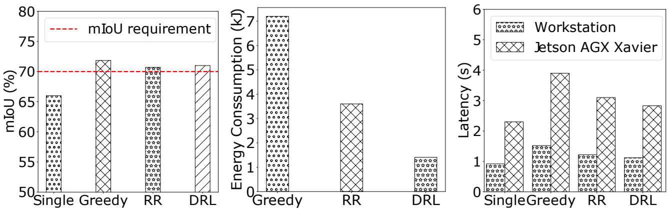
■ **Comparison with manual inspection:** Now, we compare BubCam with the manual inspection approach in the factories based on 400 RGB and 400 depth images collected in the production lines. Table 1 shows the decision accuracy and latency of the BubCam and manual inspection approaches on 200 testing RGB images. Due to the manual measurement procedure, the factory's current inspection approach achieves a low accuracy of 55.4% only. Moreover, it takes from 5 minutes to 10 minutes to manually inspect an ink bag in the factory. BubCam can achieve accuracy improvement of 1.34x and latency reduction of up to 260x, compared with the manual inspection.

■ **Comparison between BubCam's variants:** Figs. 9(a) and (b) show the impacts of the depth fusion and the number of pseudo labels on the mIoU of BubCam. Specifically, from Fig. 9(a), with the presence of the light reflection, BubCam can achieve better accuracy than its BubCam-RGB (i.e., BubCam without depth fusion) in estimating the air bubble volume under the presence of the light reflection. Specifically, BubCam and BubCam-RGB have mIoU of 64.3% and 58.8%, respectively, in the presence of light reflection. These results demonstrate the effectiveness of the depth fusion in BubCam. Moreover, Fig. 9(b) shows mIoU of the BubCam and BubCam-RGB on 200 testing images under various number of training images with the pseudo labels. First, we use 400 images with ground-truth labels to create a small training dataset. Then, we additionally include a number of the images with pseudo labels varying from 800, 400, 200, to 100 into the dataset for training the BubCam and BubCam-RGB. From Fig. 9(b), we can see that more training images with pseudo labels can help improve accuracy of the BubCam and BubCam-RGB.

7.3 Evaluation of Multi-Camera BubCam

7.3.1 Evaluation Settings. In this section, we evaluate the performance of the multi-camera BubCam with two wireless cameras (i.e., $N = 2$). As mentioned in §4, the OpenMV Cam Plus is used to prototype the wireless cameras whose activation mode and frame rate are controlled by the DRL-based adaptation agent implemented in the fog node. Moreover, the OpenMV libriaries and Micro Python 1.5.3 are used to implement the image preprocessing pipeline and camera's parameter configuration in the wireless cameras. In the fog node, we use Tensorforce 0.6.3 to implement PPO-based DRL model of two neural networks, called the actor and value networks. Each network consists of an input layer, two hidden layers and an output layer. Each hidden layer has 64 Tanh units. The Adam optimizer with a learning rate of 10^{-3} is used for training. Moreover, the adaptation period τ is set to 2 seconds. At the beginning of every period, the DRL agent observes a system state s including the presence of bubbles and reflection in nine sub-areas (i.e., $\Gamma = 9$) of the main camera's last image and the remaining energy of the wireless cameras. Then, it selects an action a to configure the activation mode and frame rate of the two wireless cameras $f \in [f_{\min}, f_{\max}]$. The $f_{\min}$ and $f_{\max}$ are set to 0 and 30 FPS, respectively. The frame rate of the main camera L515 is fixed at 30 FPS. We use the mIoU of the labeled images captured during the k^{th} adaptation period as the accuracy ϕ_k to calculate the immediate reward $r(s, a)$. The required mIoU is set to 0.7, i.e., $\phi_{\text{req}} = 0.7$.

7.3.2 DRL Training Results. The weights λ_1 and λ_2 in Eq. (1) affect the trade-off between camera energy consumption and compliance to the accuracy requirements. We evaluate the convergence of the DRL agent training under various settings for λ_1 and λ_2. Fig. 10 shows the training traces of the reward, mIoU (i.e., accuracy) and total energy consumption of the wireless cameras over 500 training epochs, each of which consists of 250 adaptation periods. The λ_1 is set to 1 while the λ_2 varies from 1, 5, to 10. From Fig. 10(a), we can see that with $\lambda_2 = 5, 10$, the reward increases and then saturates at around a similar value along the training epochs. Moreover,



(a) mIoU. **(b) Energy.** **(c) Latency.**

Figure 11: Comparison of DRL ($\lambda_1 = 1$ and $\lambda_2 = 10$) with baseline approaches. The dotted lines in (a) represents the required mIoU $\phi_{\mathbf{req}} = 0.7$.

Figs. 10 (b)-(c) show that the accuracy and energy consumption have increasing and decreasing overall trends when $\lambda_2 = 5, 10$. The results show that the training of the DRL agent can be convergent with $\lambda_2 = 5, 10$. Differently, with $\lambda_2 = 1$, the reward, accuracy and energy consumption mostly remains stable during the training.

7.3.3 DRL Execution Results. We compare the execution of the trained DRL agent with three baseline approaches which are the single-camera BubCam, greedy, and round-robin (RR) approaches. The single-camera BubCam only uses the main camera with a constant frame rate of 30 FPS. Similar to the proposed DRL approach, the greedy and RR approaches additionally deploy the wireless cameras to assist the main camera. Specifically, in the greedy approach, all wireless cameras are always activated with a fixed frame rate of 30 FPS, while the RR approach activates one wireless camera in an adaptation period.

Fig. 11 presents the mIoU, total energy consumption, and latency of the single, greedy, RR, and DRL approaches over 50 execution adaptation periods. From Fig. 11(b), due to the use of the wireless cameras, the multi-camera BubCam with DRL can achieve higher mIoU than the single-camera BubCam. The greedy, RR, and DRL approaches can satisfy the mIoU accuracy requirement. Moreover, Fig. 11(b) represents the total energy consumption of the wireless cameras under the greedy, RR, and DRL approaches. We can see that our DRL approach achieves the lowest energy consumption while the greedy approach has the highest energy consumption due to the activation of all wireless cameras over time. Furthermore, Fig. 11 (c) also shows the latency of all approaches. The single-camera BubCam can achieve the lowest latency since it does not need to process the images captured by the wireless cameras. Among the three multi-camera approaches, the proposed DRL approach can achieve the lowest latency. In summary, compared with the single-camera BubCam, the multi-camera BubCam can achieve better accuracy at the cost of higher latency and energy consumption.

8 CONCLUSION

This paper presents the design and implementation of a smart camera system, called BubCam for the automated quality inspection of the ink bags manufactured in the HP's ink production lines. BubCam employs a DL-based image segmentation and fusion pipeline to accurately estimate the volume of the air bubbles in the inspected ink bags under the complex settings and dynamic environment conditions in the factories. Furthermore, BubCam additionally deploys multiple wireless cameras to achieve better accuracy based on the multi-view visual sensing information. To save battery power of the wireless cameras, a DRL-based configuration approach is proposed to adapt the configuration for the camera's activation mode and frame rate in responses to the changes of the presence of the air bubble and light reflection. Extensive evaluation based on testbed experiments in the lab and factory environments, as well as comparison with six baseline approaches are conducted to show the effectiveness of the proposed BubCam systems.

ACKNOWLEDGMENT

This study is supported under the RIE2020 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner, HP Inc., through the HP-NTU Digital Manufacturing Corporate Lab.

REFERENCES

[1] Abolfazl Abdollahi and Biswajeet Pradhan. 2021. Integrated technique of segmentation and classification methods with connected components analysis for road extraction from orthophoto images. *Expert Systems with Applications* (2021).
[2] Xiao Bian, Ser Nam Lim, and Ning Zhou. 2016. Multiscale fully convolutional network with application to industrial inspection. In *IEEE WACV*. 1–8.
[3] Fei Chang, Mingyu Dong, Min Liu, Ling Wang, and Yunqiang Duan. 2020. A lightweight appearance quality assessment system based on parallel deep learning for painted car body. *IEEE Trans. Instrum. Meas.* (2020).
[4] Zihao Chu, Lei Xie, Tao Gu, Yanling Bu, Chuyu Wang, and Sanglu Lu. 2022. Edge-Eye: Rectifying Millimeter-level Edge Deviation in Manufacturing using Camera-enabled IoT Edge Device. In *IPSN*.
[5] Jos de Jong, Roger Jeurissen, Huub Borel, Marc van den Berg, Herman Wijshoff, Hans Reinten, Michel Versluis, Andrea Prosperetti, and Detlef Lohse. 2006. Entrapped air bubbles in piezo-driven inkjet printing: Their effect on the droplet velocity. *Physics of Fluids* 18, 12 (2006), 1–7.
[6] Yucheng Fu and Yang Liu. 2019. BubGAN: Bubble generative adversarial networks for synthesizing realistic bubbly flow images. *Chem. Eng. Sci.* (2019).
[7] Tim Haas, Christian Schubert, Moritz Eickhoff, and Herbert Pfeifer. 2020. BubCNN: Bubble detection using Faster RCNN and shape regression network. *Chemical Engineering Science* (2020).
[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. *arXiv* (2015).
[9] Haixin Huang, Xueduo Tang, Feng Wen, and Xin Jin. 2022. Small object detection method with shallow feature fusion network for chip surface defect detection. *Scientific reports* 12, 1 (2022), 1–9.
[10] Jarmo Ilonen, Tuomas Eerola, Heikki Mutikainen, Lasse Lensu, Jari Käyhkö, and Heikki Kälviäinen. 2014. Estimation of bubble size distribution based on power spectrum. In *Iberoamerican Congress on Pattern Recognition*. 38–45.
[11] Intel. 2022. L515. https://www.intelrealsense.com/lidar-camera-l515/. (2022).
[12] Liangzhi Li, Kaoru Ota, and Mianxiong Dong. 2018. Deep learning for smart industry: Efficient manufacture inspection system with fog computing. *IEEE Trans. Ind. Informat.* 14, 10 (2018), 4665–4673.
[13] Giuseppe Lisanti, Svebor Karaman, Daniele Pezzatini, and Alberto Del Bimbo. 2018. A multi-camera image processing and visualization system for train safety assessment. *Multimedia Tools and Applications* 77, 2 (2018), 1583–1604.
[14] Wenyan Pai, Jin Liang, MingKai Zhang, Zhengzong Tang, and Leigang Li. 2022. An advanced multi-camera system for automatic, high-precision and efficient tube profile measurement. *Optics and Lasers in Engineering* 154 (2022), 106890.
[15] Christian Salim, Abdallah Makhoul, Rony Darazi, and Raphaël Couturier. 2019. Similarity based image selection with frame rate adaptation and local event detection in wireless video sensor networks. *Multimed. Tools. Appl.* (2019).
[16] Judi E See, Colin G Drury, Ann Speed, Allison Williams, and Negar Khalandi. 2017. The role of visual inspection in the 21st century. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 61. 262–266.
[17] Nataliya Strokina, Jiri Matas, Tuomas Eerola, Lasse Lensu, and Heikki Kälviäinen. 2012. Detection of bubbles as Concentric Circular Arrangements. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*. 2655–2659.
[18] Kentaro Wada. 2022. Labelme. https://github.com/wkentaro/labelme. (2022).
[19] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. 2017. Pyramid scene parsing network. In *CVPR*.
[20] Siyuan Zhou, Duc Van Le, Rui Tan, Joy Qiping Yang, and Daren Ho. 2022. Configuration-Adaptive Wireless Visual Sensing System with Deep Reinforcement Learning. *IEEE Trans. Mobile Comput.* (2022).