npj | computational materials

Article

Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences

https://doi.org/10.1038/s41524-024-01216-7

Machine vision-based detections of transparent chemical vessels toward the safe automation of material synthesis

Check for updates

Leslie Ching Ow Tiong[1,2,5], Hyuk Jun Yoo[1,3,5], Nayeon Kim[1,4], Chansoo Kim[1], Kwan-Young Lee[3] ✉, Sang Soo Han[1] ✉ & Donghun Kim[1] ✉

Although robot-based automation in chemistry laboratories can accelerate the material development process, surveillance-free environments may lead to dangerous accidents primarily due to machine control errors. Object detection techniques can play vital roles in addressing these safety issues; however, existing detection models still suffer from insufficient accuracy in environments involving complex and noisy scenes. With the aim of improving safety in a surveillance-free laboratory, we report a deep learning (DL)-based object detector, namely, DenseSSD. For the foremost and frequent problem of detecting positions of transparent chemical vessels, DenseSSD achieved a mean average precision (mAP) over 95% based on a complex dataset involving both empty and solution-filled vials, greatly exceeding those of conventional detectors; such high precision is critical to minimizing failure-induced accidents. Additionally, DenseSSD was observed to be generalizable to other laboratory environments, maintaining its high precisions under the variations of solution colors, camera view angles, background scenes, experiment hardware and type of chemical vessels. Such robustness of DenseSSD supports that it can universally be implemented in diverse laboratory settings. This study conclusively demonstrates the significant utility of DenseSSD in enhancing safety within automated material synthesis environments. Furthermore, the exceptional detection accuracy of DenseSSD opens up possibilities for its application in various other fields and scenarios where precise object detection is paramount.

Automated material synthesis based on robotics and artificial intelligence has facilitated the material development process[1,2]. Recent works involved the synthesis of a wide range of materials, such as organic/polymer materials[3–9], quantum dots[10–13], and nanoparticles[14]. Although automation can substantially increase developmental efficiency, it is often accompanied by severe dangers in situations where corrosive or inflammable chemicals are handled without human surveillance. Any accidents would cause significant losses of life and property and result in some causalities in severe cases. To democratize automation in material synthesis laboratories, safety-related issues such as machine control errors need to be addressed.

Automated material synthesis for bath processes, especially wet chemical-based synthesis, inevitably involves frequent movements of transparent chemical vessels such as flasks[15–18], beakers[19–21], and vials[22,23]. If these vessels are incorrectly placed, any subsequent actions, such as solution stirring, may lead to undesired dangers. Some features in the robotic arm such as pressure sensing and control can help reduce the failure cases, however, do not fully eliminate them because undesired external variations can possibly occur unexpectedly. In this regard, detecting the movements of transparent chemical vessels is considered an important task for improving safety in automated chemical synthesis, and deep learning (DL)-based computer vision can play a vital role here. Computer vision has been widely used in object segmentation and detection[24–28] in industries such as autonomous vehicles[29,30], disease diagnosis[31], and rehabilitation[32]. However, object detectors for automated material synthesis systems have yet to be reported,

[1]Computational Science Research Center, Korea Institute of Science and Technology, Seoul 02792, Republic of Korea. [2]Mechatronics Research, Samsung Electronics Co., Ltd., Hwaseong-si 18448, Republic of Korea. [3]Department of Chemical and Biological Engineering, Korea University, Seoul 02841, Republic of Korea. [4]Department of Chemistry, Korea University, Seoul 02841, Republic of Korea. [5]These authors contributed equally: Leslie Ching Ow Tiong, Hyuk Jun Yoo. ✉e-mail: kylee@korea.ac.kr; sangsoo@kist.re.kr; donghun@kist.re.kr

and we aim to develop a high-performance detector that is suitable for identifying the error positions of transparent chemical vessels for safety purposes in the present study.

Generally, object detection is described as a collection of related computer vision tasks that involve identifying objects in the given image. Currently, models such as You Only Look Once (YOLO) and the single-shot detector (SSD) are the most popular DL-based object detectors introduced by Redmon et al.[33,34]. and Liu et al.[35], respectively. These models perform well regarding the speed of detection in real-life scenarios. However, one of the challenges is that neither YOLO nor the SSD can achieve sufficiently high detection accuracy in complex scenes with noisy background images. Recently, DEtection TRansformer (DETR) was introduced by Carion et al.[36], which utilized transformer-based structure to overcome these issues. In addition, these models also rarely focus on complex objects independently due to the lack of aggregating and exploring the information between the network layers. These limitations of YOLO, SSD, and DETR make it difficult to directly implement them in surveillance-free chemistry laboratories where very high detection precisions are required for safety purposes; this calls for the development of an improved detection model.

To achieve superior detection accuracy in an automated chemical synthesis laboratory, we report an object detector, namely, a densely connected single-shot detector (DenseSSD) with a densely connected mechanism[35]. The advantage of this model is that we proposed a densely connected pyramidal layer in the model structure so that it can be more robust when learning feature representations, and it achieved improved object detection performance over that of benchmark models including DETR, YOLOv3, YOLOv6, and SSD. Owing to these benefits, DenseSSD achieved 95.2% mean average precision[37] (mAP) on a complex dataset involving both empty and solution-filled vials, greatly exceeding the values produced by DETR, YOLOv3, YOLOv6, and SSD by 11.3%, 53.4%, 10.5%, and 18.9%, respectively. In addition, DenseSSD was found to be comparatively insensitive to the environmental changes, maintaining the highest precision under the variations of solution colors or testing view angles. Such robustness of DenseSSD allows the equipment settings to be more flexible in laboratory environments. Last, to fully realize the potential of DenseSSD, we designed the safety alert module to remotely and immediately notify researchers of possible dangers when any failures are detected by DenseSSD. Overall, DenseSSD demonstrates the potential for effective extension to various detection tasks where high detection accuracy is essential.

Results and Discussion

Workflow and model description

The overall workflow of vial positioning detection based on machine vision is schematized in Fig. 1a. The hardware consists of a vial storage box, a robot arm, multiple vial holders on the stirrer, and a camera taking a bird's-eye view of all these pieces of equipment and their related actions. Images of these hardware systems are provided in Supplementary Fig. 1. The robot arm is programmed to repeatedly attempt to move the vials in the storage

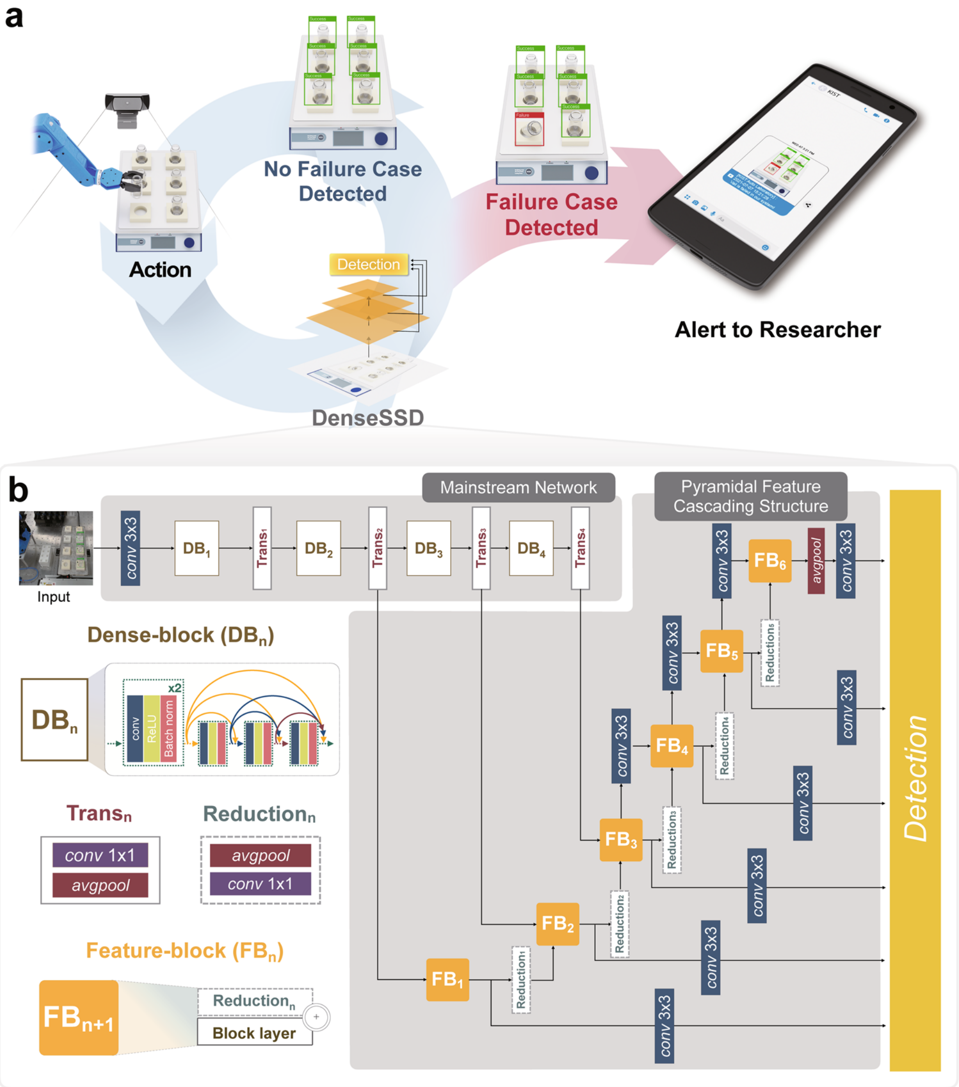

Fig. 1 | Workflow and model structure. **a** Workflow of the vial positioning detection system based on DenseSSD. An action is defined as the movement of the robot arm to relocate each vial to its holder. Green and red boxes represent the predicted bounding boxes for the success and failure cases, respectively. When a failure vial positioning case is detected by DenseSSD, an alert is remotely and immediately sent to the responsible researcher. **b** The network architectures of DenseSSD, which largely consists of mainstream structure and pyramidal feature cascading structure. The *conv* 3 × 3 and *conv* 1×1 refer to convolution layers with filter sizes 3 × 3 and 1 × 1, respectively. The *avgpool* layer means an average pooling layer with its size 2×2. Each transition layer is composed of *conv* 1×1 and *avgpool*, while each reduction layer has the reversed order composition of *avgpool* and *conv* 1×1. More detailed layer information can be found in Supplementary Table 1.

box right into the vial holders on the stirrer. In automated synthesis environments where researcher surveillance is absent, vial movement attempts may fail, possibly due to robot malfunctioning, programming errors, and any external environmental changes. Even if they may occur with a very low probability, such errors may cause dangerous situations in synthesis laboratory environments; any subsequent actions must be immediately halted. In other words, when the machine vision process detects a failure case in the vial movement actions, the system instantly stops and is designed to remotely send an alert to responsible researchers. Otherwise, the system continues to perform the following action of moving the remaining vials. Supplementary Video 1 demonstrates the overall workflow of our system including hardware movements, object detection steps, and alert modules.

We propose an object detection model, namely, DenseSSD. The original SSD model was introduced by Liu et al.[35] for object detection and recognition, and the concept of a densely connected network[38] is incorporated in DenseSSD to extract comparatively richer feature representations for achieving enhanced vial positioning detection performance. In this study, we demonstrate that DenseSSD outperforms the benchmark models, including YOLOv3, DETR, SSD, and YOLOv6 models, for the task of vial positioning detection. YOLOv3 and SSD are conventional object detection models. The YOLOv3 model was the first attempt at building a DL-based object detector, and it was proposed by Redmon et al.[34] As illustrated in Supplementary Fig. 2, the YOLOv3 model utilizes DarkNet[39] as a backbone structure by adding four convolutional (*conv*) layers to explore the features of the entire input image and predict each bounding box candidate. This means that the model globally explores the full image and all the objects in the image without performing the region proposal step. However, the limitation is that such a model lacks the ability to recognize irregularly shaped objects or groups of small objects due to its lack of exploring the global and local features of the image. To overcome this limitation, Liu et al.[35] proposed SSD by using pyramidal feature representation layers to explore and correlate the global and local features for efficiently detecting objects. Here, the global features describe the entire image with general information, such as shape information, and the local features describe the image patches with specific details, such as texture information. SSD model was inspired by the Visual Geometry Group (VGG)[40] model, which was designed as the base model for extracting useful image features. Specifically, as shown in Supplementary Fig. 2, SSD adds several feature layers with decreasing sizes; these layers are defined as pyramidal representations of images at different scales.

Recently, Li et al.[41] proposed the newest version YOLO model, namely YOLOv6, to leverage the pyramidal network for extracting global and local features information, as illustrated in Supplementary Fig. 2. Such a pyramidal structure performs better than YOLOv3 and SSD in capturing global and local features from the different scales of the representation layers to target objects of various sizes. However, this model hardly focuses on complex objects independently due to the lack of aggregating and exploring the information between the subsequent layers.

More specifically in terms of object type, many computer vision models have been developed to deal with transparent object, such as 3D shape estimation[42], segmentation[43–45], and detection[46]. For example, ClearGrasp[42] was designed for 3D shape estimation and transparent object grasp. This model requires a difficult input of depth channel from RGB-D sensor to perform the 3D shape reconstruction, which causes a longer (a few seconds) decision making process. This feature may significantly limit the usability of this model in automated chemical labs where the superfast detection speed is critical. Besides ClearGrasp, some segmentation models for transparent objects were also reported, including Trans2Seg[43], TransMatting[44], Trans4Trans[45]; however, these models were developed for segmentation purposes, rather than detection purposes, thus being unsuitable as comparative models in our study. Instead, we consider adopting the state-of-the-art transformer-based object detectors such as DETR[36] and TTG-Net[46] as reference models, which were used for detecting both transparent and opaque objects.

Considering the limitations of previous models above, we introduce a densely connected mechanism in the pyramidal cascading structure to explore and aggregate the correlations of relevant features among the subsequent layers in SSD model, namely, DenseSSD. The network architecture of DenseSSD is illustrated in Fig. 1b. DenseSSD utilizes the design concept of a densely connected mechanism[38], in which all layers are densely connected to extract comparatively global and local feature representations; in contrast, in the original YOLOv3, SSD, YOLOv6, and DETR models (Supplementary Fig. 2), the features in each *conv* layer are used as inputs for the next layer without communication. DenseSSD contains two components: a mainstream network and a pyramidal feature structure. The mainstream network consists of four dense block (DB) layers and four transition layers. The DB layers use different connectivity patterns by introducing direct connections from any layer to all subsequent layers, which improves the information flow between layers. Each layer has access to all the preceding feature maps (FMs) in its blocks and thus to the network's collective knowledge. Next, the densely connected mechanism is additionally deployed in the pyramidal feature structure to extract the multiscale FMs from different layers. The structure consists of six feature block (FB) layers as pyramidal layers and five reduction layers. The FB layers are devised to aggregate the multiple features derived from different regions and progressively explore the global and local features at each pyramidal layer. The reduction layer consists of an average pooling layer (*avgpool*) and *conv* 1×1 layer to remain the depth of FB representations. Owing to the densely connected FB layers, DenseSSD is expected to collect rich information while maintaining low feature complexity. The details of the network configurations are provided in Supplementary Table 1.

Object detection performance

To perform the training, a large-scaled data is created in this study. Our dataset only focused on standard type of vial with 20 ml, which is commonly used. The dataset was collected in the real scene of our automated material synthesis environments, which contains 10 short recorded videos. During recording, we set the camera view with 45°, as this view angle can easily cover the entire environments. A total of 789 images are extracted from the videos. Each image contained several vials, which were manually labeled as either success or failure cases, as described in Fig. 2a and Supplementary Fig. 3. Note that success cases referred to vials correctly located in their holder, whereas failure cases referred to vials in any undesired places, namely, those in *fall-out*, *lie-down*, *lean-in*, and *stand-on* situations. These four types of failure modes were chosen based on appearances in the real use cases experiments, as shown in Supplementary Video 2. Since collecting the scene for different environment conditions is difficult, data augmentations methods, including random flipping, brightness, saturation, hue, and Gaussian filter, were applied only to the images in the learning dataset to overcome the imbalanced environmental factors (Supplementary Fig. 4). We divide the dataset with the ratio 60:40 as learning (including training and validation) and testing set. Note that no images for learning set overlapped with the testing set. As a result, for the detector performance evaluations, 8,764 vial cases were used for learning, and 1,502 cases were used for testing. The details about the dataset construction process are provided in Supplementary Tables 2 and 3.

DL experiments based on the image datasets were carried out to compare the vial positioning detection performance between DenseSSD and other benchmark models (DETR[36], YOLOv3[33], SSD[35] and YOLOv6[41]). To assess the performance of these object detection algorithms, we followed the standard procedure and utilized average precision (AP) and mean average precision (mAP) as evaluation metrics[37,47,48]. AP and mAP are widely used in object detection tasks as they provide a comprehensive measure of the algorithms' precision and recall in detecting objects. In Fig. 2b, DenseSSD achieved APs of 99.9% and 90.5% for each success and failure class and mAP of 95.2%, greatly exceeding the mAPs of the YOLOv6 (86.6%), SSD (82.1%), DETR (83.9%), and YOLOv3 (61.4%). For the details, the mAP values of YOLOv3, YOLOv6, SSD, DETR, and DenseSSD are provided in Supplementary Table 4. In particular, DenseSSD outperformed the original SSD,

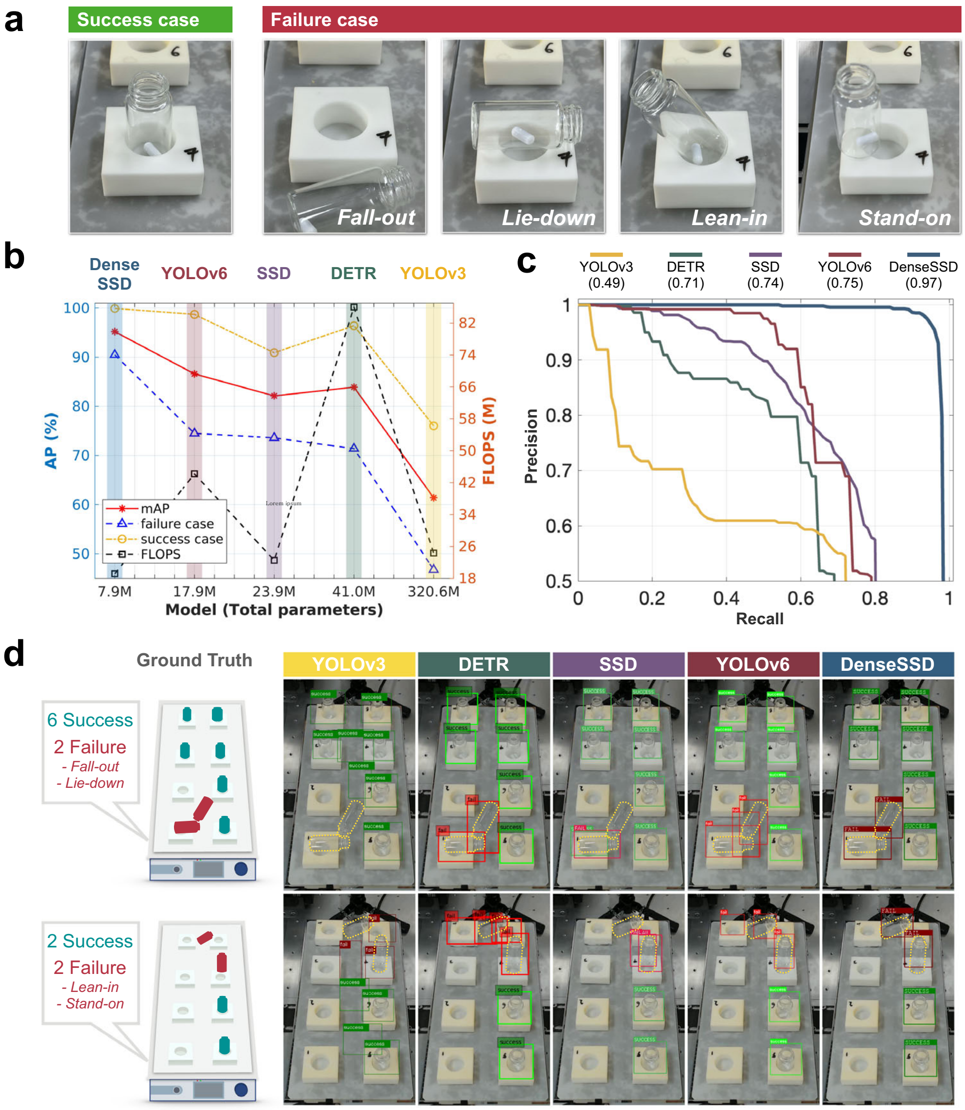

Fig. 2 | Comparisons of vial positioning detection performance. a Definitions and types of success and failure cases. Failure vial positioning involves four statuses of *fall-out*, *lie-down*, *lean-in*, and *stand-on*. **b** Detection performance of the DenseSSD, YOLOv6, SSD, DETR, and YOLOv3 models. AP, FLOPS, and total number of parameters of three models were compared. For AP evaluation, the threshold of 0.5 for the intersection over union (IoU) was used. **c** PR curves of the YOLOv3, DETR, SSD, YOLOv6 and DenseSSD models. **d** Exemplary detection results of YOLOv3, DETR, SSD, YOLOv6 and DenseSSD, achieved for two scenes. One scene contains eight vials including six success cases and two failure cases with the *fall-out* and *lie-down* statuses, while the other one contains four vials including two success cases and two failure cases with the *lean-in* and *stand-on* statuses. The failure vial cases are highlighted in yellow dotted lines for clarity.

DETR, and YOLOv6 by 16.0—16.9% for failure cases, and such high detection precision for failure cases is required to minimize failure-induced accidents. Supplementary Video 3 describes a specific example where a scene involving eight vials was used to test three models, including DenseSSD. We observed that DenseSSD performed perfectly, while some detections provided by YOLOv3, SSD, DETR, and YOLOv6 were misleading. For the YOLOv6 and DETR, multiple bounding boxes with failure labels were made for a *stand-on* and *lie-down* vial. Several errors in SSD model were that two bounding boxes with different labels were made for a *stand-on* vial. In YOLOv3, many error types were found as follows: (1) some vials were missed in the detection results, (2) failure cases were incorrectly classified as success cases, and (3) the bounding boxes were poorly made (small IoU value). In addition to the detection accuracy, the computational and memory efficiency levels of these models are compared in Fig. 2b. DenseSSD was superior to the YOLOv6, SSD, DETR, and YOLOv3 in terms of both space complexity (total parameters) and time complexity (FLOPS: floating-point operations per second). The numbers of total parameters and FLOPS were 7.9 M and 19.2 M, respectively, which were significantly smaller than those of YOLOv6 (parameter size = 17.9 M, FLOPS = 44.2 M), SSD (parameter size = 23.9 M, FLOPS = 22.5 M), DETR (parameter size = 41.0 M, FLOPS = 86.0 M), and YOLOv3 (parameter size = 320.6 M, FLOPS = 24.3 M). In DenseSSD, the complexity is substantially relieved by optimizing the parameters and simplifying the connectivity between layers because it is unnecessary to learn redundant FMs. These comparisons indicated that the object detection process is extremely fast and efficient in our DenseSSD model.

To analyze the stability of the models, precision-recall (PR) curves were obtained, as shown in Fig. 2c. The analysis was performed by measuring the area under the curve (AUC)[49]. For reference, stability is a crucial factor that increases safety by reducing the false alarm probability, which refers to incorrectly identifying a failure case as a success case. As seen in Fig. 2c, DenseSSD effectively maintained its performance when applied to real-time vial positioning detection by achieving 0.97 as its AUC value; however, YOLOv6, SSD, DETR, and YOLOv3 only achieved 0.75, 0.74, 0.71, and 0.49 as their AUC values, respectively. This means that DenseSSD maintained its AP with less degradation than other models and accurately predicted bounding boxes for detections. Through this analysis, we conclude that DenseSSD exhibited significantly higher stability in detecting failure cases than the benchmark models and thereby can minimize false alarm probability.

Some detection examples are provided in Fig. 2d and Supplementary Fig. 5, where most of the failure cases were accurately detected by DenseSSD without false alarm issues, in contrast, YOLOv3, DETR, SSD, and YOLOv6 did not perform well. As can be seen in Fig. 2d and Supplementary Fig. 5a-5c, we noticed that vials positioned between vial holders were difficult to capture with YOLOv3, DETR, SSD, and YOLOv6 due to their transparency,

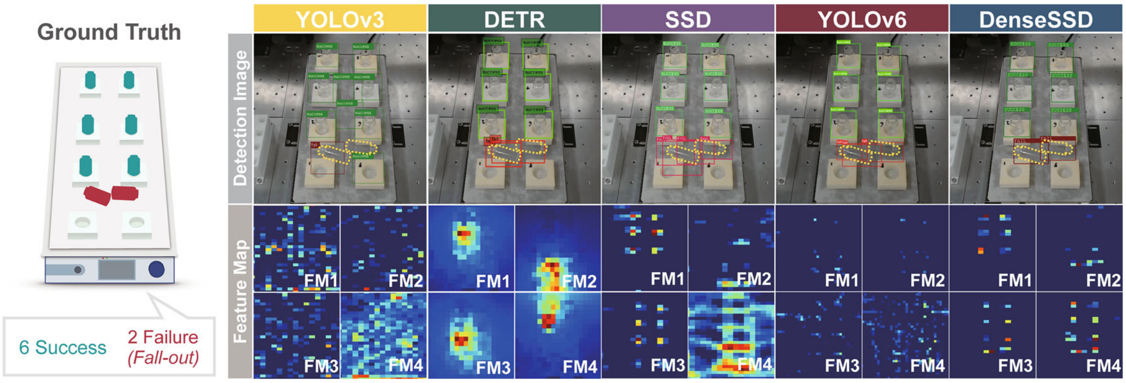

Fig. 3 | Visualization of the FM representations. The ground-truth scene contains eight vials including six success cases and two failure cases with the *fall-out* status. The failure vial cases are highlighted in yellow dotted lines for clarity. The detection results and four FMs yielded by YOLOv3, DETR, SSD, YOLOv6, and DenseSSD are compared.

but they were successfully detected by DenseSSD. This indicates that DenseSSD is advantageous in differentiating the transparent objects exhibiting low brightness from the noisy background. In addition, we also find in Fig. 2d and Supplementary Fig. 5, that vials positioned on vial holders (either *lean-in* or *stand-on* failure cases) were erroneously detected by SSD with two bounding boxes made for each object; the phenomenon was fixed with DenseSSD. YOLOv6 and DETR also suffered the errors by capturing additional bounding boxes on overlapped positions between failure cases. We speculate that these benefits were likely derived from densely connected mechanism and pyramidal feature cascading structure in DenseSSD, which could explore and correlate the global and local features for object detections regardless of low brightness and noisy background.

To further understand the reasons why DenseSSD outperformed the other methods, we visualize the FMs of the three models in Fig. 3 and Supplementary Fig. 6. The FMs were captured in the FBs of the DenseSSD model, the attention layer of DETR, and the *conv* layers of YOLOv3, SSD and YOLOv6 models (Fig. 1b and Supplementary Fig. 2), and they look substantially different. For DenseSSD, the FMs focus on the localized regions of vials, and in particular, FM2 of DenseSSD successfully distinguishes two failure cases from the remaining success cases. For the DETR, the attention layer can capture the feature consistently; however, the mechanism is still sensitive to background noise and tiny-size appearance of vial. In contrast, the FMs of SSD and YOLOv6 are obviously less clear, and those of YOLOv3 are irregular and noisy, which is consistent with the higher performance of our DenseSSD model. These comparisons indicate that DenseSSD built clearer and richer FM representations that enhanced the pyramidal feature cascading structure to differentiate between the multiple vial positions. This is likely because unlike the competing approaches, DenseSSD receives direct supervision and reuses the feature patterns from the previous layers in its FBs, which share collective knowledge to improve the detection performance of the overall method.

Application involving solution-filled vial datasets

Since vials are often filled with solutions in chemistry laboratories[50–53], the study needed to be expanded to datasets involving solution-filled vials. The positioning failures of solution-filled vials would be more dangerous than those of empty vials because the liquids in these vials may pour out and jeopardize neighboring equipment. Therefore, we performed object detection experiments by constructing more complex datasets involving both empty and solution-filled vials. Figure 4a describes the composition of the training dataset used for the detection experiments. While the empty vial dataset was the same as the one used in the experiments in Fig. 2, we additionally collected 359 images for the dataset of solution-filled vials. Here, the colors of the filled solutions were randomly designed to make the detectors more robust to color changes. The vials in each image were labeled as either success or failure cases. The data augmentation procedures were also applied to the images in the learning dataset, as described in Fig. 4b and Supplementary Fig. 7. As a result, a total of 17,174 vial cases were used for detector training, and 2,282 cases were used for testing. More details about the dataset construction process are provided in Supplementary Tables 2 and 3.

We first measured the precisions achieved when the models were trained only with the empty vial dataset, as shown in the upper panel of Fig. 4c. DenseSSD, YOLOv6, SSD, DETR, and YOLOv3 achieved mAPs of only 81.2%, 80.2%, 75.5%, 72.5%, 30.0% respectively, indicating that the empty vial dataset is not sufficient on its own for detecting the positions of solution-filled vials. In particular, DenseSSD only achieved a 67.5% AP for the failure cases, which is far from the satisfactory level. Next, we measured the precisions achieved when the models were trained with the full datasets containing both empty and filled vials. The mAP value of DenseSSD reached 95.2%, greatly exceeding 84.7%, 76.3%, 84.5%, and 41.8% of YOLOv6, SSD, DETR, and YOLOv3. Very interestingly, the precision of failure cases was enhanced dramatically for DenseSSD, whereas it was limitedly improved for SSD, DETR, and YOLOv6. The AP of DenseSSD for the failure cases was improved to 90.8%, which was significantly higher than those of SSD, DETR, and YOLOv6. Those high precision for the failure cases is critical to minimizing failure-induced accidents. The mAP values of YOLOv3, DETR, SSD, and YOLOv6 are provided in Supplementary Table 5. The detection results obtained in the example scenes (Fig. 4d) reveal the specific incorrect detection cases. For example, some solution-filled vials with the *stand-on*, *lean-in* status, and even success cases were missed in YOLOv3 and SSD detections. Also, DETR, SSD, and YOLOv6 show the frequent problems of double bounding boxes with different labels for a single object, causing misleading detection results. Overall, DenseSSD was demonstrated to be the most effective vial positioning detection method for both empty and solution-filled vials, greatly outperforming the existing models.

Testing view angle sensitivity

For wide utilization of DenseSSD in diverse lab environments, the discussion of camera setting is needed to get consistent detection performance globally. One of the important things in camera setting is view angle variations. In Figs. 2 and 4, overlapped objects between transparent vials sometimes occur, and induce detection confusion of model. Indeed, degradation of detection performance from overlapped objects is affected by sensitivity of camera angles. Therefore, we modified the datasets by adding images taken from different angles to evaluate the robustness of DenseSSD to camera angle variations. Several view angles of investigation included 30°, 45°, 60°, and 90°, as illustrated in Fig. 5a. The modified dataset included 2,377 original images across different angles. The data augmentation processes, such as image flipping, brightness, hue, and Gaussian blurring augmentation, were applied only to the images in the learning set (Supplementary Fig. 4). As a result, for the detector performance evaluations,

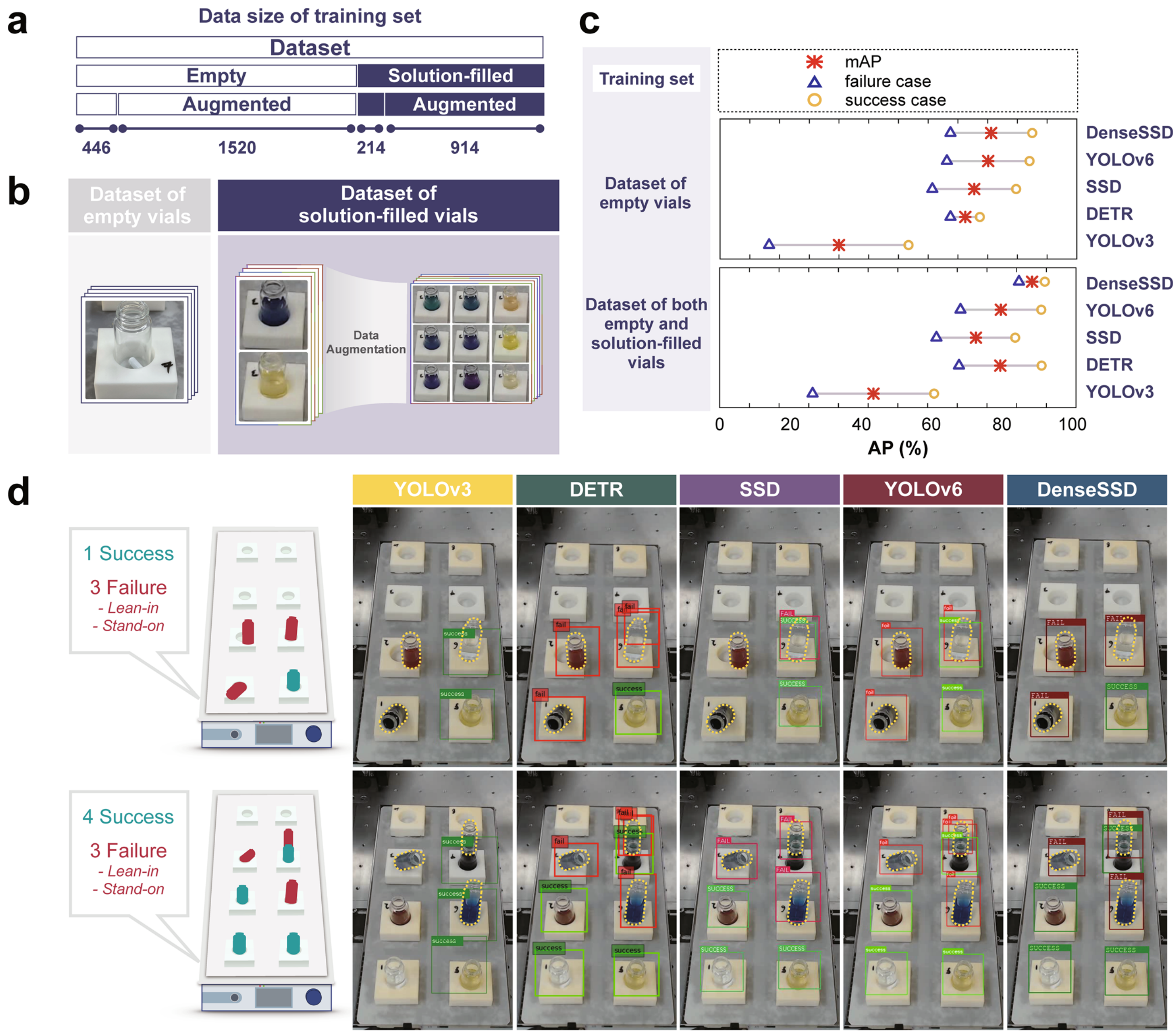

Fig. 4 | Application of DenseSSD to solution-filled vial datasets. a Compositions of the training datasets used for the detection experiments. The number refers to the number of images in the training dataset. **b** The appearances of empty vials and solution-filled vials. Image processing techniques based on hue and saturations were applied to the images involving solution-filled vials for data augmentations. **c** Detection performance (AP) comparisons between YOLOv3, DETR, SSD, YOLOv6, and DenseSSD under different training set construction procedures: one that conducts training with only the empty vial dataset and another that conducts training with the dataset containing both empty and filled vials. **d** Exemplary detection results with other benchmark models and DenseSSD, achieved for two scenes involving multiple solution-filled vials. One scene (Case 1) contains four vials including one success cases and three failure cases with the *lean-in* and *stand-on* statuses, while the other one (Case 2) contains seven vials including four success cases and three failure cases with the *stand-on* and *lean-in* statuses. The failure vial cases are highlighted in yellow dotted lines for clarity.

32,715 vial cases were used for learning, and 3,648 cases were used for testing. More details about the dataset construction process are provided in Supplementary Tables 2 and 3.

Transfer learning was performed by using the pretrained weights from the dataset with only 45° angles. In Fig. 5b, we observe that DenseSSD achieved the best mAPs on the testing datasets corresponding to different angles compared to those of SSD, DETR, YOLOv6 and YOLOv3: 88.5% (30°), 94.8% (45°), 93.8% (60°), and 84.9% (90°). The mAP values of SSD, DETR, YOLOv6 and YOLOv3 are provided in Supplementary Table 6. Considerable mAP reductions at 30° and 90° was observed for all models. DenseSSD maintained its high mAP (over 93%) between 45° and 60°, indicating that its performance is highly insensitive to the testing view angle, and thus the model is robust to angle variations. Unlike that of DenseSSD, the mAPs of the other methods, including YOLOv3 and SSD, hit their highest values at only 45° and noticeably dropped at both 30° and 60°; these techniques are comparatively sensitive to the provided view angles. Interestingly, DETR is performed well consistently for all the angles within the mAP range of 82–84%. Overall, this test revealed that DenseSSD is robust to environmental changes. This insensitivity of DenseSSD is beneficial in a chemical laboratory because it allows the utilized equipment settings to be more flexible. The detection precisions achieved by DenseSSD with angle variations can further be understood with specific examples in Fig. 5b and c, involving two failure vial positioning cases with the *lean-in* and *stand-on* statuses. At 30°, two vials appeared to overlap, making the differentiation of these two objects highly difficult; as a result, the *lean-in* vial was incorrectly detected as the success case. Similarly, at 90° angle, the *lean-in* vial was detected as success, likely due to the very similar appearances of these two types of vials from the top view. Testing view angles between 45° and 60° could prevent these limitations and led to enhanced mAPs exceeding 93.8%. In terms of camera settings, testing view

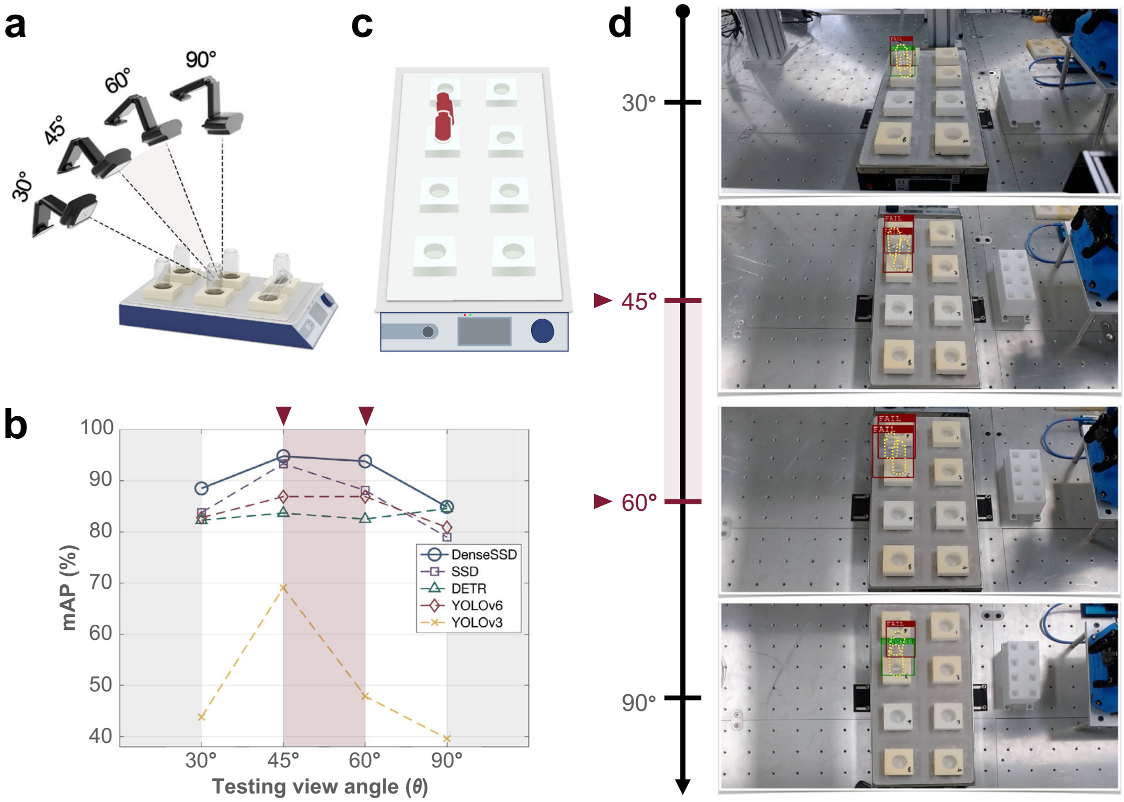

Fig. 5 | Testing view angle sensitivity. a A scheme illustrating four camera angles (30°, 45°, 60°, and 90°). The origin for angle measurement is the center position of the stirrer. **b** Vial positioning detection performance (mAP) as a function of the testing view angle. **c** A scheme illustrating a ground-truth image involving two failure cases, one each with the *lean-in* and *stand-on* statuses. **d** The scenes and their corresponding detection results with DenseSSD for each tested camera angle. The failure vial cases are highlighted in yellow dotted lines for clarity. At 45° and 60°, both failure cases were correctly detected. At 30° and 90°, the vial in the *lean-in* status was incorrectly predicted as the success case.

angles between 45° and 60° prevent performance degradation from transparent overlapped objects.

Generalizability to other laboratory environments

To expand the impact of our proposed model, we conducted further deep learning experiments to assess its generalizability in various laboratory environmental settings. The primary objective of these experiments was to enhance the performance of the DenseSSD, YOLOv6, and DETR models specifically for the environmental settings. However, it is important to note that the generalization capability of the model heavily relies on the quality and diversity of the datasets used for training[54]. The concept of quality and diversity in our study encompassed different aspects of the laboratory environment, including camera views, indoor locations, experiment desks, illumination conditions, and hardware variations (such as different stirring machines). To ensure robust performance and adaptability, we focused on expanding the training datasets to include a wide range of scenarios that cover both success and failure cases in unconstrained environments. By doing so, we not only aim to improve model performance, but also enhance generalization, capture variability, and effectively address rare events.

In terms of training protocols, we used existing datasets along with a more diverse and generalized dataset. We carefully rearranged and collected a comprehensive training dataset that encompasses various types of stirring machines. We categorized the stirring machines into three types based on the allocated object distance between the vials: “sparse”, “semi-sparse”, and “dense”, which are illustrated in Fig. 6a. The “sparse” type refers to configurations where there is a significant distance between individual vials, resulting in fewer vials within the machine. On the other hand, the “dense” type represents configurations with closely packed vials, resulting in a higher density of vials within the machine. The “semi-sparse” type falls in between, with a moderate level of vial spacing and density. These categories provide a systematic way to account for variations in vial arrangements and help the models learn and adapt to different levels of vial density in real-world scenarios. Additionally, we considered the coverage of different environmental settings and illumination conditions, encompassing both bright and dark environments in the external laboratories. In terms of camera view, illumination conditions from backgrounds affect confusion of camera focus which degrade clarity of transparent objects. For visual confirmation and further details, please refer to Fig. 6b, which provides a comprehensive overview of the experimental setups. This approach allowed us to capture different configurations and variations in the positioning of the vial holders, enabling the models to learn and adapt to a wider range of real-world scenarios. Conversely, for the testing protocols, we also collected additional datasets that encompassed different environmental settings (particularly featured by vial holders self-designed by 3D printers in Fig. 7b), enabling us to evaluate the model’s performance in previously unseen environments.

The comparison results for the experimental settings are presented in Fig. 7a and Supplementary Table 7. A total of 594 scenes from the different experimental settings was used as the test set, including different types of stirring machines accommodating 1 vial (“sparse” type), 9 vials (“semi-sparse” type), and 25 vials (“dense” type), respectively. As shown in Fig. 7a and Supplementary Table 8, our DenseSSD model (without fine-tuning) consistently achieved the highest mAP values across all the test scenarios. Importantly note that DenseSSD and other comparative models applied in these testing experiments are not fine-tuned by transfer learning. Specifically, for the “sparse” type of stirring machine, DenseSSD achieved a mAP of 84.9% and 90.4% in bright and dark environments, respectively. When it came to the “semi-sparse” type of stirring machine, DenseSSD achieved a mAP of 75.1% in the bright environment and 90.1% in the dark

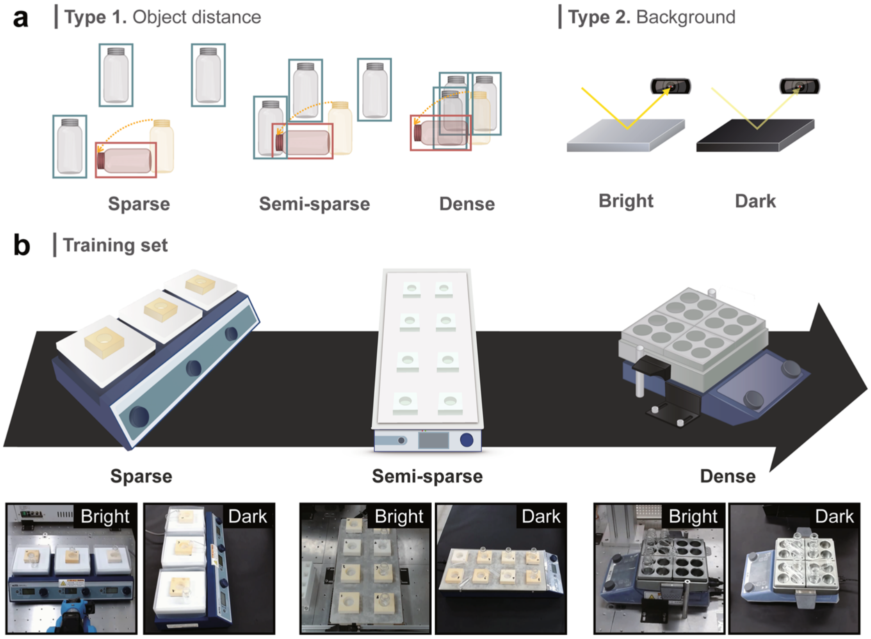

Fig. 6 | The designation of training dataset in unconstrained environments. a Definition of three types of stirring machine based on the allocated object distance between the vials: “sparse”, “semi-sparse”, and “dense”; the coverage of different environmental settings and illumination conditions, encompassing both bright and dark environments in the chemistry laboratories. **b** Visualization on several examples of training dataset in the unconstrained environments.

environment. Additionally, for the “dense” type of stirring machine, DenseSSD achieved mAP values of 62.8% and 79.9% in the bright and dark environments, respectively. The comparative models, including DETR and YOLOv6, achieved mAP values of less than 79% for the “sparse” type, less than 70% for the “semi-sparse” type, and less than 49% for the “dense” type, respectively. To sum up detection results, DenseSSD substantially outperforms YOLOv6 and DETR in all tested cases. Interestingly, DenseSSD shows consistently high performances in dark environment due to clarity of transparent objects. As can be seen in Fig. 7b and Supplementary Figs. 8–10, the lightness of transparent objects in “dark” backgrounds is higher than “bright” backgrounds, so that DenseSSD can find it easier to locate transparent objects and classify success vs. failure cases, leading to the much enhanced detection precisions of DenseSSD over YOLOv6 and DETR. In Supplementary Figs. 8–10, the comparisons of real detection scenes visually illustrate the exceptional performance of our DenseSSD model, showcasing its remarkable ability to accurately detect and precisely localize vials in various experimental configurations.

The results shed light on the significant impact that the DenseSSD model can have in preventing failures within automated laboratory systems. Through rigorous testing and evaluation, we demonstrated that the DenseSSD model achieved exceptional performance in accurately detecting and localizing vials across various laboratory setups. Notably, the advanced pyramidal feature cascading structure employed within DenseSSD played a pivotal role in generating clearer and more detailed feature maps, empowering the model to effectively discern between multiple vial positions in the unconstrained environments. These findings not only highlight the tangible benefits of implementing algorithms such as DenseSSD in laboratory automation environments for chemical experiments but also provide valuable insights for generalizing its effectiveness to other environments.

Although we have confirmed thus far that our DenseSSD model performs excellently in terms of vial positioning detections, the benefits of this work could possibly be reduced in flexible automation systems. Recently, the community for lab automation is shifting away from hard-coded manipulation path toward more flexible automations. In this work, the movement of robotic arm are hard-coded and the background scene remains same. On the other hand, in the flexible automation systems, the hardware movements and background scenes could unexpectedly be variant, where the training of machine learning models could be much more difficult. However, our results provide insights into the generalizability of DenseSSD to other laboratory environments that involve complex hardware settings or different chemical vessels. We showcase exemplary detection scenes, including cuvette positioning, in Supplementary Fig. 11, comparing the detection results of DenseSSD with those of YOLOv6 and DETR. These results support the notion that with appropriate transfer learning techniques, our DenseSSD model can be extended to handle not only complex hardware settings, but also various types of chemical vessels in diverse laboratory environments.

Last, to fully realize its potential in surveillance-free chemistry laboratory environments, we additionally introduced a safety alert module that was designed to remotely notify researchers of possible dangers immediately after failure cases are detected by DenseSSD. Supplementary Fig. 12 shows the scheme of the alert module. Upon the detection of failure cases, any hardware system operations are immediately halted, and the alert module remotely sends the scene image and related text such as the event time and problematic vial’s information to the user’s messenger based on TCP/IP network communications. Such an alert module is an indispensable component of an automated laboratory environment because it is critical to minimizing failure-related losses. Supplementary Fig. 13 describes our alert module, which was tested with several popular messengers around the globe, including Facebook Messenger and Telegram.

In summary, we developed a DL-based object detector, namely, DenseSSD, which was demonstrated to accurately detecting chemical vials’ positions in chemistry laboratory environments. Recently, the automation of chemical synthesis has garnered much attention mainly due to its potential to significantly increase material development efficiency; however, safety issues have rarely been addressed today. In the present study, DenseSSD significantly outperformed the previous detectors of YOLOv3, DETR, SSD, and YOLOv6 exhibiting detection precisions over 95% for the complex datasets involving both empty and solution-filled vials. The enhanced precision will no doubt contribute to minimizing the possible losses of life and property in surveillance-free laboratory environments. Also, DenseSSD is found to be robust to environmental changes. It maintains the high precisions under the variations of solution colors, testing view angles, background scene, and types of experiment hardware and chemical vessels, which supports the generalizability of DenseSSD in diverse laboratory environments. Furthermore, since DenseSSD model is a general object detector, we expect this will not be limited to chemistry-relevant datasets; thus, it will be useful for other detection tasks, such as self-driving vehicles, medical imaging, and remote sensors, where high detection accuracy is of utmost importance.

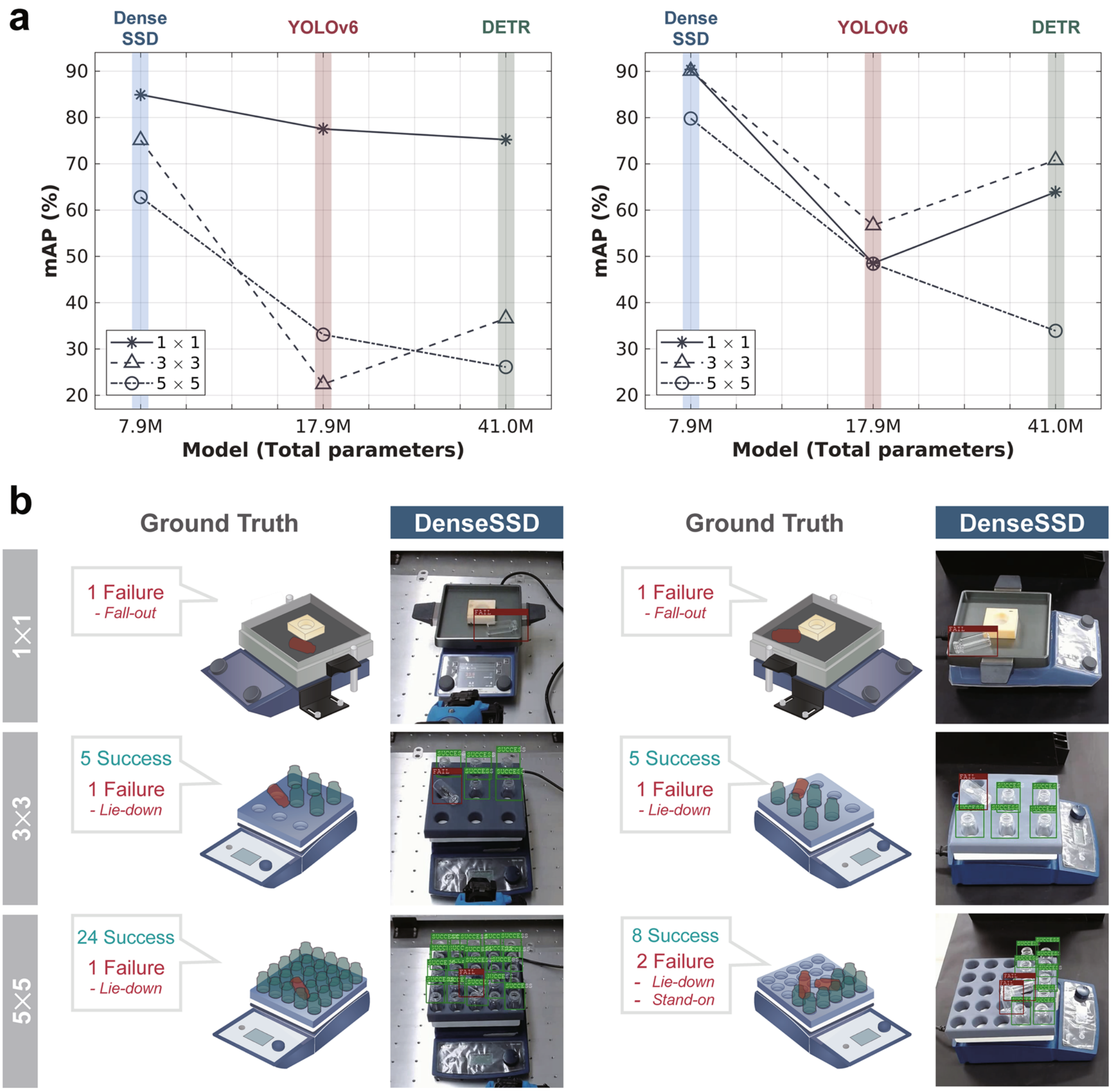

Fig. 7 | Generalizability of DenseSSD to other environmental settings. **a** Comparisons of detection performance on between DenseSSD, YOLOv6 and DETR models with different types of stirring machine and different environments. **b** Exemplary detection scenes on DenseSSD model with different types of stirring machine and environments. More exemplary detection scenes are provided in Supplementary Figs. 8–10.

Methods

Hardware

The hardware systems consisted of a vial storage box, a robot arm, and eight vial holders on a stirrer, as described in Supplementary Fig. 1. A camera was also set up for monitoring the equipment and their interactions. The vial storage box was prepared to supply vials, and the robot arm was programmed to relocate the vials in the box to the holders on the stirrer.

Object detection models

Regarding the configuration of DenseSSD, the model contains five FB layers and five reduction layers, as shown in Fig. 1b. Let an FB be a FM block with l layers of H that are composed of DBs and rectified linear unit layers:

$$\mathrm{FB} = H_l([b_0, b_{l-1}]) \tag{1}$$

where b_0 and b_{l-1} represent the transition and DB layers, respectively. The operator $[\cdot]$ is defined as a concatenation operator. Then, a reduction layer is implemented in the early stage of the FB and performs an average pooling operation and a 1 × 1 *conv* layer to aggregate the FM representations with the same dimensionality. Here, the 1 × 1 *conv* states that the filter size of the *conv* layer is 1 × 1. All the configurations are listed in Supplementary Table 1. To achieve the training objective, we define a total loss[34] (L_{total}) function as a weighted sum of the localization (L_{loc}) loss and the confidence loss (L_{conf}) as follows:

$$L_{\mathrm{total}}(x,c,l,g) = \frac{1}{N}\left(L_{\mathrm{conf}}(x,c) + \alpha L_{\mathrm{loc}}(x,l,g)\right) \tag{2}$$

where x, c, l, and g are defined as the input image, multiclass confidence scores, predicted box, and ground-truth box, respectively. N is defined as the number of matched boxes, and α is the weight for the localization loss. In this work, α is set as 0.5. Specifically, L_{loc} is presented as follows:

$$L_{\mathrm{loc}}(x,l,g) = \sum_{i\in Pos}^{N}\sum_{m} x_{ij}^{k}\, smooth_{L1}\left(l_i^m - g_j^m\right) \tag{3}$$

where $smooth_{L1}(\cdot)$ is defined as the smooth L1 loss[55] to calculate the localization loss between the predicted boxes (l) of m and the ground-truth (g) boxes of m. Here, $m \in \{cx, cy, w, h\}$, where cx and cy are defined as the center of the default bounding box; w and h are defined as the width and height of the bounding box, respectively. In addition, x_{ij}^{k} is an indicator for matching the ith predicted box to the jth ground-truth box of category k. Next, L_{conf} is the softmax loss over multiple classes of confidence I:

$$L_{\mathrm{conf}}(x, c) = -\sum_{i\in Pos}^{N} x_{ij}^{k}\log\left(\hat{c}_i^{p}\right) - \sum_{i\in Neg}^{N}\log\left(\hat{c}_i^{0}\right) \tag{4}$$

$$\hat{c}_i^{p} = \frac{\exp\left(c_i^{p}\right)}{\sum_{p}\exp\left(c_i^{p}\right)} \tag{5}$$

where $\hat{c}_i^{p}$ is known as the softmax loss function; p is defined as a predicted box with a specific class, and 0 refers to a predicted box as a negative sample or background information.

During the training process, we applied the stochastic gradient descent (SGD) optimizer[56] with a learning rate of 1.0×10^{-3}, a weight decay of 1.0×10^{-8} and a momentum of 0.9. In our experiments, the batch size was set to 64, and the training procedure was carried out for 100 epochs. Training was conducted on our dataset and was performed by following the protocols that were mentioned in the "Model structure and performance comparison" section. We also randomly divided the images in the training set for cross-validation purposes by taking 20% of the images as the validation set and the remaining images as the training set. Note that the model was trained using an Nvidia Tesla V100 GPU.

For YOLOv3, SSD, DETR, and YOLOv6, we utilized all the existing models that were provided by their respective authors. We therefore made our best effort to modify the networks from the existing models by following DETR[36], YOLOv3[34], SSD[35] and YOLOv6[41] to perform fine-tuning by conducting training with our dataset, as mentioned earlier. We also applied the SGD optimizer with a learning rate of 1.0×10^{-3}, a weight decay of 1.0×10^{-8} and a momentum of 0.9. In these experiments, the batch size was set to 64, and the training process was carried out for 100 epochs. The models were trained using an Nvidia Tesla V100 GPU.

Evaluation metrics for object detection

AP is a popular metric for measuring the accuracy of object detection models. The calculation of AP involves only one class. The metric can be calculated using the following equation:

$$\mathrm{AP} = \sum_{\mathrm{i}=1}^{\mathrm{n}-1}\left(\mathrm{r}_{\mathrm{i}+1} - \mathrm{r}_{\mathrm{i}}\right)\mathrm{pre}\left(\mathrm{r}_{\mathrm{i}+1}\right) \tag{6}$$

where $r_1, r_2, \ldots, r_n$ are the recall levels at which the precision (pre) is first interpolated. Next, the mAP is the average AP across all classes, which can be defined as follows:

$$\mathrm{mAP} = \frac{1}{\mathrm{M}}\sum_{\mathrm{i}=1}^{\mathrm{M}}\mathrm{AP}_{\mathrm{i}} \tag{7}$$

where M is defined as the number of classes.

Data availability

Our datasets follow the COCO format which is standard protocol in computer vision. Several examples of the real-time scenes are available in the GitHub repository (https://github.com/KIST-CSRC/DenseSSD/tree/main/dataset/test_sample). The full dataset can be accessed via the link: https://doi.org/10.5281/zenodo.7687879.

Code availability

The code for the pre-trained model of DenseSSD is available in the GitHub repository (https://github.com/KIST-CSRC/DenseSSD). This repository also provides entire code explanation, and educational version code to utilize for other scope, such as Jupyter notebook. All the codes are written in Python 3.7, and they work well in Python versions of 3.7, 3.8, and 3.9. The architecture of DenseSSD is implemented using PyTorch 1.7.

Received: 28 September 2022; Accepted: 25 January 2024;
Published online: 22 February 2024

References

1. Häse, F., Roch, L. M. & Aspuru-Guzik, A. Next-generation experimentation with self-driving laboratories. *Trends. Chem.* **1**, 282–291 (2019).
2. Soldatov, M. A. et al. Self-driving laboratories for development of new functional materials and optimizing known reactions. *Nanomaterials* **11**, 619 (2021).
3. Granda, J. M., Donina, L., Dragone, V., Long, D.-L. & Cronin, L. Controlling an organic synthesis robot with machine learning to search for new reactivity. *Nature* **559**, 377–381 (2018).
4. Adamo, A. et al. On-demand continuous-flow production of pharmaceuticals in a compact, reconfigurable system. *Science* **352**, 61–67 (2016).
5. Steiner, S. et al. Organic synthesis in a modular robotic system driven by a chemical programming language. *Science* **363**, eaav2211 (2019).
6. Coley, C. W. et al. A robotic platform for flow synthesis of organic compounds informed by AI planning. *Science* **365**, eaax1566 (2019).
7. Burger, B. et al. A mobile robotic chemist. *Nature* **583**, 237–241 (2020).
8. Langner, S. et al. Beyond ternary OPV: High-throughput experimentation and self-driving laboratories optimize multicomponent systems. *Adv. Mater.* **32**, 1907801 (2020).
9. Du, X. et al. Elucidating the full potential of OPV materials utilizing a high-throughput robot-based platform and machine learning. *Joule* **5**, 495–506 (2021).
10. Li, J. et al. Autonomous discovery of optically active chiral inorganic perovskite nanocrystals through an intelligent cloud lab. *Nat. Commun.* **11**, 2046 (2020).
11. Higgins, K., Valleti, S. M., Ziatdinov, M., Kalinin, S. V. & Ahmadi, M. Chemical robotics enabled exploration of stability in multicomponent lead halide perovskites via machine learning. *ACS Energy Lett* **5**, 3426–3436 (2020).
12. Li, J. et al. AIR-Chem: Authentic intelligent robotics for chemistry. *J. Phys. Chem. A.* **122**, 9142–9148 (2018).
13. Epps, R. W. et al. Artificial Chemist: An autonomous quantum dot synthesis Bot. *Adv. Mater.* **32**, 2001626 (2020).
14. Salley, D. et al. A nanomaterials discovery robot for the Darwinian evolution of shape programmable gold nanoparticles. *Nat. Commun.* **11**, 2771 (2020).
15. Seo, J., Lee, S., Koo, B. & Jung, W. Controlling the size of Pt nanoparticles with a cationic surfactant, $\mathrm{C_nTABr}$. *Cryst. Eng. Comm.* **20**, 2010–2015 (2018).
16. Shejawal, K. P. et al. Green synthesis of silver, iron and gold nanoparticles of lycopene extracted from tomato: their characterization and cytotoxicity against COLO320DM, HT29 and Hella cell. *J. Mater. Sci.: Mater. Med.* **32**, 19 (2021).
17. Marișca, O. & Leopold, N. Anisotropic gold nanoparticle-cell interactions mediated by collagen. *Materials* **12**, 1131 (2019).
18. Nthunya, L. N. et al. Green synthesis of silver nanoparticles using one-pot and microwave-assisted methods and their subsequent embedment on PVDF nanofiber membranes for growth inhibition of mesophilic and thermophilic bacteria. *New. J. Chem.* **43**, 4168–4180 (2019).
19. Dhand, V. et al. Green synthesis of silver nanoparticles using coffea arabica seed extract and its antibacterial activity. *Mater. Sci. Eng. C* **58**, 36–43 (2016).
20. Tailor, G., Yadav, B. L., Chaudhary, J., Joshi, M. & Suvalka, C. Green synthesis of silver nanoparticles using Ocimum canum and their anti-bacterial activity. *Biochem. Biophys. Rep.* **24**, 100848 (2020).

21. Alshehri, A. A. & Malik, M. A. Facile one-pot biogenic synthesis of Cu-Co-Ni trimetallic nanoparticles for enhanced photocatalytic dye degradation. *Catalysts* **10**, 1138 (2020).
22. Miller, P. J. & Shantz, D. F. Covalently functionalized uniform amino-silica nanoparticles. Synthesis and validation of amine group accessibility and stability. *Nanoscale Adv.* **2**, 860–868 (2020).
23. Tripathi, R. M., Yoon, S.-Y., Ahn, D. & Chung, S. J. Facile synthesis of triangular and hexagonal anionic gold nanoparticles and evaluation of their cytotoxicity. *Nanomaterials* **9**, 1774 (2019).
24. Porzi, L., Bulo, S. R., Colovic, A. & Kontschieder, P. Seamless scene segmentation. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition.* 8277–8286 (2019).
25. Orr, I., Cohen, M. & Zalevsky, Z. High-resolution radar road segmentation using weakly supervised learning. *Nat. Mach. Intell.* **3**, 239–246 (2021).
26. Fan, Q., Zhuo, W., Tang, C.-K. & Tai, Y.-W. Few-shot object detection with attention-RPN and multi-relation detector. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition.* 4013–4022 (2020).
27. Shiri, P. et al. Automated solubility screening platform using computer vision. *iScience* **24**, 102176 (2021).
28. Eppel, S., Xu, H., Bismuth, M. & Aspuru-Guzik, A. Computer vision for recognition of materials and vessels in chemistry lab settings and the vector-lab pics data set. *ACS. Cent. Sci.* **6**, 1743–1752 (2020).
29. Yurtsever, E., Lambert, J., Carballo, A. & Takeda, K. A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access* **8**, 58443–58469 (2020).
30. Ye, M., Xu, S. & Cao, T. HVNet: Hybrid voxel network for LiDAR based 3D object detection. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition.* 1631–1640 (2020).
31. Shen, L. et al. Deep learning to improve breast cancer detection on screening mammography. *Sci. Rep.* **9**, 12495 (2019).
32. Debnath, B., O'Brien, M., Yamaguchi, M. & Behera, A. A review of computer vision-based approaches for physical rehabilitation and assessment. *Multimed. Syst.* **28**, 209–239 (2022).
33. Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR).* 779–788 (2016).
34. Redmon, J. & Farhadi, A. YOLOv3: An incremental improvement. Preprint at https://arxiv.org/abs/1804.02767 (2018).
35. Liu, W. et al. SSD: Single shot multi-box detector. In *European Conference on Computer Vision (ECCV)*, 21–37 (Springer, 2016).
36. Carion, N. et al. End-to-end object detection with transformers. *European Conference on Computer Vision (ECCV)*. 213–229 (2020)..
37. Beitzel, S. M., Jensen, E. C. & Frieder, O. Average Precision Histogram. In *Encyclopedia of Database Systems* (ed. Liu, L., Özsu, M.T.) 194 (Springer, 2009).
38. Huang, G., Liu, Z., van der Maaten, L. & Weinberger, K. Q. Densely connected convolutional networks. *IEEE Conf. Comput. Vis. Pattern. Recognit. (CVPR)*, 4700–4708 (2017).
39. Redmon, J. & Farhadi, A. YOLO9000: Better, faster, stronger. *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 6517–6525 (2017)..
40. Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. *Int. Conf. Learning Representations (ICLR)* 1–14 (2015).
41. Li, C. et al. YOLOv6: A Single-Stage Object detection framework for industrial applications. Preprint at https://arxiv.org/abs/2209.02976 (2022).
42. Sajjan, S. S. et al. ClearGrasp: 3D Shape estimation of transparent objects for manipulation. *International Conference on Robotics and Automation (ICRA)*. 3634–3642 (2020).
43. XIE, E. et al. Segmenting transparent object in the wild with transformer. Preprint an https://arxiv.org/abs/2101.08461 (2021).
44. Cai, H., Xue, F., Xu, L. & Guo, L. TransMatting: Enhancing transparent objects matting with transformers. *European Conference on Computer Vision (ECCV)*. 253-269 (2022).
45. Zhang, J. et al. Trans4Trans: Efficient transformer for transparent object and semantic scene segmentation in real-world navigation assistance. *IEEE Trans. Intell. Trans. Syst.* **23**, 19173–19186 (2022).
46. Zhang, J. et al. Transformer based feature pyramid network for transparent objects Grasp. *IEEE Int. Conf. Intell. Robot. Appli. (IJIRA)* **13456**, 417–429 (2022).
47. Zhao, Z.-Q. et al. Object detection with deep learning: A review. *IEEE Trans. Neural Netw. Learn. Syst* **30**, 3212–3232 (2019).
48. Padilla, R. et al. A survey on performance metrics for object-detection algorithms. *IEEE Int. Conf. Syst. Signals & Image Proc. (IWSSIP)*, 237–242 (2020).
49. Davis, J. & Goadrich, M. The relationship between precision-recall and ROC curves. *Proc. 23rd Conf. Mach. Learn. (PMLR)*. 233–240 (ACM Press, 2006).
50. Betke, A. & Kickelbick, G. Bottom-up, wet chemical technique for the continuous synthesis of inorganic nanoparticles. *Inorganics* **2**, 1–15 (2014).
51. Wang, L., Sun, Y., Li, Z., Wu, A. & Wei, G. Bottom-up synthesis and sensor applications of biomimetic nanostructures. *Materials* **9**, 53 (2016).
52. Reverberi, A. P., Kuznetsov, N. T., Meshalkin, V. P., Salerno, M. & Fabiano, B. Systematical analysis of chemical methods in metal nanoparticles synthesis. *Theor. Found. Chem. Eng.* **50**, 59–66 (2016).
53. Jamkhande, P. G., Ghule, N. W., Bamer, A. H. & Kalaskar, M. G. Metal nanoparticles synthesis: An overview on methods of preparation, advantages and disadvantages, and applications. *J. Drug Delivery Sci. Technol.* **53**, 101174 (2019).
54. Kawaguchi, K., et al. Generalization in deep learning. *Preprint at* https://arxiv.org/abs/1710.05468 (2017).
55. Rezatofighi, H. et al. Generalized intersection over union: A metric and a loss for bounding box regression. *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 658–666 (2019).
56. Smith, S. L., Dherin, B., Barrett, D. & De, S. On the origin of implicit regularization in stochastic gradient descent. *Int. Conf. Learn Represent (ICLR)* (2021)..

Acknowledgements

This work was supported by the National Research Foundation of Korea funded by the Ministry of Science and ICT (NRF-2021M3A7C2089739 and NRF-2022M3H4A7046278), and IITP (No. 2021-0-02076) by the Ministry of Science and ICT of Korea government.

Author contributions

D.K and S.S.H conceived the idea and supervised the project. L.C.O.T and H.J.Y developed DenseSSD. L.C.O.T, H.J.Y, and N.K performed data collections, augmentations, labeling, and object detection experiments with DenseSSD, DETR, SSD, YOLOv3, and YOLOv6. H.J.Y. developed the safety alert module. All authors contributed to results analysis and manuscript writing. L.C.O.T and H.J.Y contributed equally to this work.

Competing interests

The authors declare no competing financial or non-financial interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41524-024-01216-7.

Correspondence and requests for materials should be addressed to Kwan-Young Lee, Sang Soo Han or Donghun Kim.

Reprints and permissions information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.