University of Tennessee, Knoxville

TRACE: Tennessee Research and Creative Exchange

Doctoral Dissertations

Graduate School

8-2022

Deep Learning-Based Robotic Perception for Adaptive Facility Disinfection

Da Hu
University of Tennessee, Knoxville, dhu5@vols.utk.edu

Follow this and additional works at: https://trace.tennessee.edu/utk_graddiss

Part of the Civil Engineering Commons, and the Construction Engineering and Management Commons

Recommended Citation

Hu, Da, "Deep Learning-Based Robotic Perception for Adaptive Facility Disinfection. " PhD diss., University of Tennessee, 2022.
https://trace.tennessee.edu/utk_graddiss/7245

This Dissertation is brought to you for free and open access by the Graduate School at TRACE: Tennessee Research and Creative Exchange. It has been accepted for inclusion in Doctoral Dissertations by an authorized administrator of TRACE: Tennessee Research and Creative Exchange. For more information, please contact trace@utk.edu.

To the Graduate Council:

I am submitting herewith a dissertation written by Da Hu entitled "Deep Learning-Based Robotic Perception for Adaptive Facility Disinfection." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Civil Engineering.

Shuai Li, Major Professor

We have read this dissertation and recommend its acceptance:

Qiang He, Mingzhou Jin, Jindong Tan

Accepted for the Council:

Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

DEEP LEARNING-BASED ROBOTIC PERCEPTION FOR ADAPTIVE FACILITY DISINFECTION

A Dissertation Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

Da Hu
August 2022

Copyright © 2022 by Da Hu
All rights reserved.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor, Dr. Shuai Li for his support and guidance throughout my graduate studies. I am very fortunate to have him as my advisor who continuously provides keen insight and wise judgement to my research. He sparks my research passion and inspires me to think critically and creatively. He has always supported me. He provides an enabling and supportive environment for me to succeed in my PhD study and research. Without him, I wouldn’t be who I am now. Also, his aesthetic sensitivity and devotion have a profound influence on me. I would also like to thank Dr. Qiang He, Dr. Mingzhou Jin, and Dr. Jindong Tan for serving on my committee, and for their valuable suggestions to help me improve my studies.

I will always be thankful to my colleagues and lab-mates. It was a great experience to work with them on various NSF and TDOT-funded projects and share our experience and life. I also feel indebted to all my lab-mates who provided enormous supports for my research.

I especially thank my wife, Junxuan Zhao, for years of selfless love and supports. I simply couldn’t have done this without her. Thanks to my family, who are always supporting me through trying times, always believing in me, and for giving me strength. I will forever owe my achievements to my dedicated, caring, and thoughtful family.

ABSTRACT

Hospitals, schools, airports, and other environments built for mass gatherings can become hot spots for microbial pathogen colonization, transmission, and exposure, greatly accelerating the spread of infectious diseases across communities, cities, nations, and the world. Outbreaks of infectious diseases impose huge burdens on our society. Mitigating the spread of infectious pathogens within mass-gathering facilities requires routine cleaning and disinfection, which are primarily performed by cleaning staff under current practice. However, manual disinfection is limited in terms of both effectiveness and efficiency, as it is labor-intensive, time-consuming, and health-undermining. While existing studies have developed a variety of robotic systems for disinfecting contaminated surfaces, those systems are not adequate for intelligent, precise, and environmentally adaptive disinfection. They are also difficult to deploy in mass-gathering infrastructure facilities, given the high volume of occupants. Therefore, there is a critical need to develop an adaptive robot system capable of complete and efficient indoor disinfection.

The overarching goal of this research is to develop an artificial intelligence (AI)-enabled robotic system that adapts to ambient environments and social contexts for precise and efficient disinfection. This would maintain environmental hygiene and health, reduce unnecessary labor costs for cleaning, and mitigate opportunity costs incurred from infections. To these ends, this dissertation first develops a multi-classifier decision fusion method, which integrates scene graph and visual information, in order to recognize patterns in human activity in infrastructure facilities. Next, a deep-learning-based method is proposed for detecting and classifying indoor objects, and a new mechanism is developed to map detected objects in 3D maps. A novel framework is then developed to detect and segment object affordance and to project them into a 3D semantic map for precise disinfection. Subsequently, a novel deep-learning network, which integrates multi-scale features and multi-level features, and an encoder network are developed to recognize the materials of surfaces requiring disinfection. Finally, a novel computational method is developed to link the recognition of object surface information to robot disinfection actions with optimal disinfection parameters.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE
INTRODUCTION

Infrastructure facilities such as hospitals and airports become hotbeds for pathogen transmission, causing infectious disease outbreaks, which in turn cause illness and deaths. They may thereby impose significant burdens on healthcare systems, reduce productivity, and lead to enormous economic losses. For example, the COVID-19 pandemic has infected more than 259 million people and has caused more than 5.1 million deaths [1]. The number of infections and deaths continues to increase with the emergence of more infectious variants, stoking fears of future surges of infections. Surface cleaning and disinfection are essential to reducing pathogen transmissions and infection risks. However, manual disinfection processes are unproductive, necessitating efficient disinfection with robots. While many robots have been developed and tested in infrastructure facilities, they remain inadequate, given the critical need for automated, intelligent, and precise disinfection, in order to reduce pathogen transmission and exposure, thus preventing outbreaks of infectious disease and infections alike. This chapter introduces the background of this dissertation and reviews related studies as well as identify limitations concerning current practices. Then, the research objectives and contributions are elaborated.

Background

Hospitals, schools, airports, and other environments built for mass gatherings can become hot spots for microbial pathogen colonization, transmission, and exposure, greatly accelerating the spread of infectious diseases across communities, cities, nations, and the world. Outbreaks of infectious diseases lead to illness and death, imposing significant burdens on healthcare systems, reducing productivity, and leading to enormous economic losses. Healthcare facilities are particularly of concern during the pandemic, given the influx of infected patients needing treatment. In healthcare facilities, surfaces can be contaminated through touch, respiratory droplets, or bodily secretions. Surface contamination can cause cross-transmission among patients and between patients and healthcare providers, jeopardizing people's health as well as the normal operations of hospitals [2]. In fact, prior to the COVID-19 pandemic, the US Centers for Disease Control and Prevention estimated that nearly 1.7 million patients are infected during hospitalization per year, resulting in 98,000 associated deaths [3]. The annual direct cost of hospital-acquired infections (HAIs) to hospitals is estimated to range from 28 to 45 billion dollars, but could be much higher when accounting for indirect costs [4]. The disastrous impacts of infections on societies and economies are enormous, highlighting the urgency for developing effective surface disinfection methods to mitigate the spread of infectious pathogens in built environments.

Many facilities still rely on cleaning staff to carry out disinfection processes, such as applications of hydrogen peroxide and ultraviolet disinfection, which are time-consuming and labor-intensive, in addition to posing infection risks for cleaning staff [5]. Furthermore, manual disinfection is influenced by human behavioral factors, which render real-world practices highly variable [6]. Research indicates that fewer than 50% of hospital surfaces can be considered clean after disinfection procedures using the current standard methods [7]. For instance, Rutala and Weber [8] found that, in the rooms of patients infected with C. difficile, MRSA, and VRE, 10%–50% of the surfaces become contaminated. However, 51% of the surfaces in patient rooms are found not to be thoroughly cleaned or disinfected, which could lead to a 120% increase in the probability of infection for future occupants of those rooms. Therefore, there is a critical need for adpative and precise robotic disinfection, in order to reduce viral bioburdens on contaminated surfaces and thus prevent fomite-mediated transmission of infectious pathogens. Elevated concerns in response to the COVID-19 pandemic have increased the adoption of robotic technologies for infection control and environmental hygiene [9]. Despite the great potential, however, existing technologies cannot meet the needs for intelligent, precise, and environment-adaptive disinfection.

Related studies and limitations

Routine cleaning and disinfection are the most important practices for mitigating the spread of infectious pathogens in mass-gathering built environments. Many disinfection robots have been developed and tested during the COVID-19 pandemic to combat infectious diseases. However, these robots are unable to assess or respond to situations and social contexts so as to determine when, where, what, and how to disinfect. This section briefly reviews the related studies and highlights the limitations.

Limitations in activity perception in healthcare facilities. Human activity recognition (HAR) has been extensively studied in various fields, including human behavior analysis and human-robot interaction. Researchers in these fields have developed various algorithms for identifying human activities such as jumping, watching TV, calling, and more. Visually-based and sensor-based approaches are two major techniques for activity recognition utilizing different types of data [10]. For applications in robots, visually-based methods have dominated research areas concerned with enabling robots to capture and understand the surrounding human context. The visually-based HAR methods can be categorized into traditional machine learning, deep learning, and multimodal feature fusion. Traditional machine-learning-based approaches have relied heavily on the quality of handcrafted features, which are not robust and perform poorly on large-scale datasets. Methods based on deep learning and multimodal feature fusion have achieved promising results and have demonstrated the potential of convolutional neural networks (CNNs) for the task of HAR. However, existing deep-learning-based methods only rely on visual features; they do not explore rich objects or relationship information within the images, which means that they may miss critical cues for activity recognition. Therefore, there is a critical need to develop a method which utilizes objects and their mutual relationships in images, in order to achieve activity classification with a deep understanding of visual content.

Limitations in object detection and mapping. Many studies have been dedicated to automating the recognition of objects from images in healthcare facilities. Despite their advances, three knowledge gaps still need to be closed. First, the performance of deep-learning-based object detection is largely influenced by the quality and generalizability of training data. While many image datasets exist, very few of them, if any, are focused on object detection within environments that are as cluttered as healthcare facilities. Moreover, existing studies have not explored the importance of high-touch objects for the disinfection robot. As a result, no datasets have been generated with the aim of disinfection, so high-touch objects may inevitably be ignored in datasets. To overcome this limitation, this study introduces a new image dataset for indoor object detection in healthcare facilities, with a total of 57 object categories. Second, the deep-learning methods developed in most studies have complex architectures and cannot achieve satisfactory accuracy in real time. In this dissertation, a new, lightweight deep-learning network is designed by integrating spatial and channel attention mechanisms into the YOLOv5 architecture, in order to improve both accuracy and computational efficiency. A third challenge remains, namely, how to map object-detection results in RGB images to a 3D map that can be used for robot navigation and disinfection.

Limitations in the detection of potentially contaminated surfaces. The tasks of detecting and segmenting areas of potential contamination from images are related to object detection and semantic segmentation. In addition, the concept of object affordance is also relevant. Many studies have been conducted in object detection and recognition [11–15], scene classification [16,17], and indoor-scene understanding [18,19]. However, merely detecting scene elements is not enough to make intelligent decisions. Particularly for this application, the detection of a computer on an office desk does not imply a need to disinfect that computer. The existing object detection and segmentation techniques lack the capabilities for reasoning out the circumstances under which an object, or some part of it, needs specific disinfection. Understanding how humans interact with different objects will help determine potential areas of contamination. Human interactions with objects have implications for which parts of objects may be contaminated and could contaminate different parts of the human body. Studies in computer vision have been conducted to predict the affordance of whole objects [20,21]. In [22], a region proposal approach was integrated with a CNN feature-based recognition method in order to enable the detection of affordance. In [23], a method was developed to learn affordance-segmentation using weakly supervised data. In [24], several studies of object-affordance in computer vision and robotics were reviewed. Despite these advancements, there remains a gap in linking semantic segmentation and object-affordance with the areas of potential contamination.

Limitations in object surface material recognition. Healthcare facilities harbor a variety of pathogens that may colonize a wide spectrum of surfaces made from varying materials. Object-surface materials have significant impacts on pathogen colonization and transmission; thus, correspondingly varying disinfection modes, parameters, and procedures are needed to ensure complete and efficient disinfection. Material recognition is a long-standing problem, and numerous methods have been developed to address it. However, material recognition is challenging due to the diverse range of appearances and spatially invariant features. Existing studies, such as Deep-Ten [25] and DEP [26], aim to capture spatially invariant features of materials with the integration of an orderless feature-pooling layer in an end-to-end learning fashion. DSRNet [27] is another network that periodically captures recurrent features by learning the inherent spatial dependencies of multiple primitives from different directions. However, these networks cannot capture low-level texture or color information, which is important for material-recognition tasks. In the latest study, CLASSNet [28] utilized information from different layers, thereby achieving state-of-the-art performance in material recognition, even though spatially invariant features were not captured. No study, to our knowledge, has utilized multi-scale features for material representations. Furthermore, there is a lack of a network either that is capable of learning or that enables the learning of both multi-level and multi-scale features simultaneously and either encoding or the encoding of these features in an orderless manner to capture spatially invariant features.

Limitations in robotic arm disinfection planning. The fifth knowledge gap is the lack of methods for area coverage path-planning with contextual data that can inform UV surface disinfection with appropriate dosages. In the absence of such methods, The UV dosages applied to some surfaces may be insufficient. Current practice focuses primarily on solving coverage-planning for the disinfection robot with a UV light column mounted on a mobile robot—which cannot be deployed in the presence of humans. Therefore, there is a critical need for an intelligent area-coverage path-planning method that enables the disinfection robot to appropriately adapt to different environments and ensure complete and efficient disinfection.

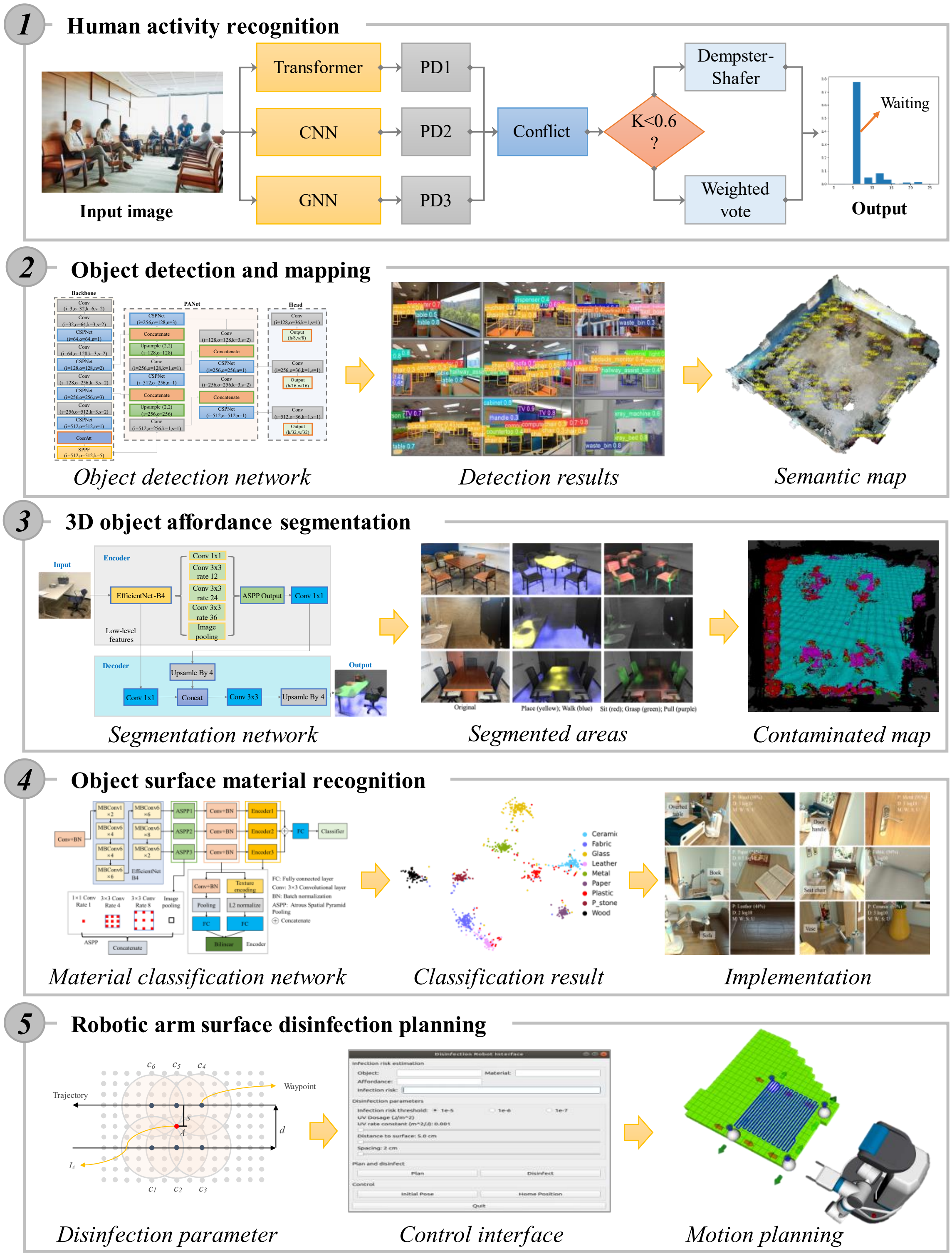
Research objectives and contributions

The overarching goal of this research is to develop an intelligent and adaptive robotic disinfection framework for efficient and precise disinfection in built environments. The proposed methods are expected to allow the disinfection robot to recognize human contexts, detect high-touch objects, segment potentially contaminated areas, recognize object-surface materials, and generate disinfection motion with optimal parameters. Figure 1.1 illustrates the overview of the research. Five specific objectives are pursued in order to achieve the overarching research goal.

1) The first objective is to recognize human activities in healthcare facilities as they relate to robot disinfection decisions. The accomplishment of this objective will facilitate the continuous perception of human contexts, thus enabling the robot to make informed disinfection decisions.

2) The second objective is to detect and classify objects in healthcare facilities and project them onto a 3D map. The accomplishment of this objective will enable the robot to identify high-touch objects for efficient disinfection.

3) The third objective is to detect and segment object affordance indicating potentially contaminated areas and to then project them onto a 3D semantic map. The accomplishment of this objective will endow the disinfection robot with the ability to recognize potentially contaminated areas for precise disinfection.



Disinfection Robot Interface

Infection risk estimation

Object: 　Material:

Affordance:

Infection risk:

Disinfection parameters

Infection risk threshold: ● 1e-5 ○ 1e-6 ○ 1e-7

UV Dosage (J/m^2)

UV rate constant (m^2/J): 0.001

Distance to surface: 5.0 cm

Spacing: 2 cm

Plan and disinfect

Plan　Disinfect

Control

Initial Pose　Home Position

Quit

Figure 1.1. Research overview

4) The fourth objective is to recognize material types for the object surfaces requiring disinfection. The accomplishment of this objective will facilitate the recognition of surface material to adapt disinfection parameters.

5) The fifth objective is to computationally link the recognition of contaminated surfaces, objects, and materials with the robotic disinfection actions, thus completing the loop from robotic assessment to robotic actions. The accomplishment of this objective will enable the robot to generate efficient disinfection trajectories with appropriate dosages.

This novel and original research is expected to establish a roadmap towards realizing adaptive and intelligent robotic disinfection, in order to promote health in built environments. There are five specific contributions of this research.

1) This research proposes a new image-to-scene-graph pipeline to synthesize the scene graph dataset with node and edge attributes for the image-classification dataset, thus enabling rapid deployment and testing for scene-graph classification networks. Scene graphs are extracted from images in order to capture high-level semantic information consisting of relationships between humans and objects in the scene. A graph neural network (GNN) is then designed based on a graph-attention mechanism in order to classify scene graphs associated with images. Finally, a multi-classifier hybrid decision fusion method is proposed for combining outputs from a CNN, a visual transformer, and a GNN, which can capture various types of features in images.

2) This research creates a new image dataset with well-classified object categories in healthcare facilities, providing a benchmark for the task of object recognition in healthcare environments. Note that most of high-touch objects and equipment in healthcare facilities are annotated in the newly created dataset. Then, a novel, deep-learning-based object-recognition method was designed by integrating spatial- and channel-attention mechanisms into the YOLOv5 architecture, so as to improve both accuracy and computational efficiency. A novel mapping

framework, consisting of object-detection, object-coordinate estimation, and object-clustering, is developed to project detected objects onto 3D maps.

3) This research develops a novel deep-learning method to segment object-affordance from RGB-D images and map the segments to areas of potential contaminations onto a 3D map. Areas with frequent human interactions, which may be colonized by a variety of pathogens, can be automatically detected and segmented. Using the visual simultaneous localization and mapping technique, segmented areas of potential contamination are mapped onto a 3D space. The 3D semantic occupancy map and the locations of the areas of potential contamination are exploited for the robot's disinfection planning.

4) This research develops a novel deep-learning network for recognizing materials captured by the disinfection robot in infrastructure facilities. The proposed network integrates multi-level CNN features, thus leveraging both low- and high-level information to capture semantic and textural information. High-level features can capture semantic features, which are abstract representations of the material. Low-level features can capture more subtle details, such as textural information. An Atrous Spatial Pyramid Pooling (ASPP) module can extract multi-scale features by resampling feature maps at multiple rates. The ASPP module increases the size of the receptive field without compromising feature-map resolution. Our network further integrates an encoder component, which combines both orderless and local spatial-feature pooling. The encoder can preserve textural and ordered spatial information from different layers, which can better capture the spatially-invariant features of materials.

5) This research proposes a new method for computationally linking the recognition of materials, affordance, and objects with robotic disinfection actions. The innovation lies in the computational modeling of the interactions between surfaces, pathogens, and disinfection modes as well as parameters for responsively adapting robotic disinfection actions, which has not been achieved

by existing studies or current systems. The developed methods could lead to an intelligent robotic disinfection paradigm that goes well beyond existing systems, which are perceived as roaming UV lights for coarse disinfection. Intelligent and precise robotic disinfection can be implemented in critical infrastructure facilities such as hospitals, airports, school buildings, and food-processing plants in order to significantly improve environmental and public health.

Dissertation organization

The remainder of this dissertation is organized as follows. Chapter 2 presents the novel, deep learning-based method for human-context recognition. Chapter 3 presents a novel 3D object-mapping framework. Chapter 4 presents a deep learning-based 3D method for generating contamination maps for robotic navigation. Chapter 5 presents a new deep learning network designed to enable robots to recognize various material types of object surfaces. Chapter 6 presents a computational method to link the recognition of object surface information to robot disinfection action. Finally, Chapter 7 concludes the dissertation by summarizing its findings.

CHAPTER TWO
HUMAN ACTIVITY RECOGNITION

Introduction

Mass-gathering built environments such as healthcare facilities are often hotspots for pathogen transmission. The United States Centers for Disease Control and Prevention (CDC) report shows a large number of outbreaks of infectious diseases in hospitals [29]. These disease outbreaks impose a substantial burden on the healthcare system and are one of the leading causes of death [30]. For example, CDC estimates that 1.7 million people could be infected by infectious diseases while being treated in hospitals every year. These hospital-acquired infections (HAIs) are associated with 99,000 deaths annually [3]. In addition, HAIs have an estimated direct medical cost of $30 billion each year and more than $10 billion in costs to society from early deaths and lost productivity. More recently, Coronavirus disease 2019 (COVID-19) has infected more than 449 million and caused more than 6 million deaths worldwide [1]. Worse still, the number of infections continues to aggressively increase due to the emergence of more infectious variants. Routine cleaning and disinfection are one of the most important practices to limit the transmission of infectious diseases in healthcare facilities [31].

Nowadays, traditional manual cleaning and disinfection is still the primary practice in healthcare facilities, which is labor-intensive and time-consuming. Furthermore, there is potential for a person to miss or insufficiently disinfect contaminated surfaces due to fatigue [8]. The robot holds great potential to address these limitations given its ability to operate 24/7 without putting humans at risk [32]. The pandemic of COVID-19 has accelerated the adoption of disinfection robots for environmental cleaning and disinfection [9]. Despite the potential, existing disinfection robot systems are unable to recognize human activity in healthcare facilities, which largely impacts their efficiency and hinders their deployment. This is because a variety of human activities are happening within healthcare facilities, which pose different implications for the robot regarding how to proceed with the disinfection task. For example, when a doctor meeting or consulting is going on in the room, the robot should be better moving to the next place needing disinfection, instead of interrupting human activities. Therefore, human activity recognition is the premise of the successful deployment of disinfection robots in healthcare facilities.

Recognizing human activity is a challenging task and has been an active research area for two decades. A variety of algorithms have been developed for the classification of human activity based on videos or a single image [33]. Considering that the disinfection robot needs to continuously move and disinfect within the building, our work focuses on classifying human activity from static images to provide a timely understanding of the surrounding human context. While video data can provide both image and temporal information, a lot of activities can also be classified in still images or video frames [34]. In addition, video data sizes are relatively large, requiring a high computational power for model training and inference [35]. Therefore, recognizing human activity from still images is more suitable for the application of the disinfection robot. It should be noted that human activity recognized from still images in different views and time frames can also be merged for a more robust classification. However, classifying human activity in a single image remains a challenging task, especially for images with disturbance and cluttered backgrounds.

To address this challenge, this study develops a multi-classifier decision fusion method, which integrates scene graphs and visual information to achieve a reliable and robust human activity recognition in healthcare facilities. This study features three contributions to the body of knowledge. First, a new image-based human activity dataset is created, consisting of common human activities in hospitals with ground-truth activity annotations. This provides the first benchmark for quantitative evaluation of models to classify human activity from images in healthcare facilities. Second, a new image-to-scene-graph pipeline is proposed to synthesize the scene graph dataset with node and edge attributes for the image classification dataset, which enables rapid deployment and testing for scene graph classification networks. Scene graphs are extracted from images to capture high-level semantic information consisting of relationships among humans and objects in the scene. A Graph Neural Network (GNN) is then designed based on a graph attention mechanism to classify scene graphs associated with images. Finally, a multi-classifier hybrid decision fusion method is proposed to combine outputs from Convolutional Neural Network (CNN), Visual Transformer (ViT), and GNN, which can capture different types of features in images. The hybrid fusion approach alleviates the high-conflict issue in decision fusion by integrating the Dempster-Shafer theory and weighted majority vote method.

Literature review

Human activity recognition (HAR) has been extensively studied in various fields such as human behavior analysis and human-robot interaction. Researchers in these fields have developed a variety of algorithms to identify human activities such as jumping, watching TV, calling, etc. Visually-based and sensor-based approaches are two major techniques for activity recognition, which utilize different types of data [10]. Specifically, visually-based methods utilize image and video data collected using camera systems such as RGB and RGB-D cameras. On the other hand, sensor-based systems use different types of sensors such as accelerometers, WiFi, and Radar. For the application of robots, visual-based methods have dominated the research areas which enable the robot to capture the surrounding human context. The visual-based HAR methods can be categorized into traditional machine learning, deep learning, and multimodal feature fusion.

Related studies on traditional machine learning for HAR

Machine learning algorithms, such as support vector machine (SVM) and random forest, have been developed to identify human activities from still images. These traditional machine learning methods achieved promising results under relatively simple scenarios with small sample size. For example, Wang et al. [36] developed an unsupervised learning approach to classify action classes present in images by matching the coarse shape of a human to reference images. The human shape was obtained from the canny edge detector with a series of post-processing. The spectral clustering approach was used to cluster the images with the same actions based on the distance between image pairs. In [37], Ikizler et al. improved the method by modeling human pose by applying the Histogram of Oriented Rectangles to rectangular patches. The linear discriminant analysis was adapted for feature reduction, and the output was fed into a classifier for action classification.

Yao et al. [38] proposed a new human action classification benchmark dataset "Stanford 40 Action". In their work, action attributes and parts were jointly modeled by learning sparse bases for image representation. The Locality-constrained Linear Coding (LLC) method on dense SIFT features was used to train an action attribute classifier. The action parts consist of object and poselet that were extracted from pre-trained Deformable Parts Model and poselet detector. The extracted attributes and parts features were finally fed into a linear SVM classifier. Yun et al. [39] utilized image patches centered at each hand of a person, where hand positions can be extracted from human skeleton. The image patch feature combined with its distance to human torso was used as feature representation for each human activity. The feature representation was then fed into an SVM classifier based on geodesics on Riemannian manifolds to classify human activity. However, traditional machine learning-based approaches heavily relied on the quality of handcrafted features, which are not robust and perform poorly on a large-scale dataset.

Related studies on deep learning for HAR

With the advancement of computational power, more recent attention has focused on deep learning-based HAR methods. Deep learning-based methods can automatically learn feature representation for human activities, which can significantly improve classification performance and reduce feature selection efforts. For example, Oquab et al. [40] investigated how mid-level image representation learned with CNN on ImageNet can be transferred to the task of HAR and improve its performance. It should be noted that ground-truth person bounding boxes were needed during training and testing. Gkioxari et al. [41] adapted RCNN method to use both person and contextual information for classification. The person information was used as primary region, which is provided by the manually annotated bounding box in the images. Contextual information was obtained through bottom-up region proposals, which are candidate secondary regions. The candidate secondary region with the highest probability is treated as the most important contextual information and is added to primary region to generate action representation. Zhao et al. [42] proposed a deep learning-based framework for activity prediction, which consists of part localization and part action networks. In their method, a keypoint prediction network was first adapted to detect human joints and generate bounding boxes of defined parts. The part action network combined features from global body and local part actions. The global body action was obtained through annotated person bounding boxes. All of these methods achieved good performance on popular action classification benchmarks such as PASCAL VOC 2012 and Stanford 40 action. However, manually annotated person bounding boxes are required for these methods, which are typically not available in real-world applications.

To address this limitation, a large and growing body of literature has developed HAR methods without any manual person bounding box annotations. For example, Khan et al. [43] proposed an action-specific person detection method for action classification. The outputs of the person detector were fed into a CNN to extract deep features, The extracted features were further used to identify human activity using SVM with linear kernel. Similarly, Siyal et al. [44] utilized CNN features from the last pooling layer of ResNet-18, and trained an SVM classifier on extracted features to classify action in an image. The main drawback of these methods is that CNN features and classifiers are learned separately, which could compromise efficiency and performance. To overcome this problem, Bera et al. [45] proposed an end-to-end CNN based on a keypoints-based attention mechanism for action recognition in a single image. Specifically, keypoints of an image were first generated using scale-invariant feature transform (SIFT) algorithm. The Gaussian Mixture Model (GMM) clustering algorithm was then applied to these keypoints to group them into a number of clusters, which were further converted to salient regions. The salient regions were fed into an attentional module to learn the importance of each salient region in activity classification. More recently, Bas and Ikizler-Cinbis [46] proposed a top-down and bottom-up attentional deep multiple instance learning network for action recognition. In their method, the top-town attention module was designed to identify action-related regions in images. The bottom-up attention layer was used to generate a pixel-level action map by removing irrelevant pixels. All of these methods achieved promising results and demonstrated the potential of CNN for the task of HAR. However, these methods only rely on visual features without exploring rich objects and relationship information that appears in the images.

Related studies on multimodal feature fusion for HAR

Multimodal data have been demonstrated to be beneficial for the interpretation of complicated human activities by providing rich semantic knowledge [10]. For example, Khaire et al. [47] combined RGB, depth, and skeletal data for activity classification. In their work, a new skeleton data processing approach was created to extract human skeletons from RGB images according to human physical structures. The depth and skeletal data are more robust in undesired lighting conditions compared to RGB images. Guo et al. [48] developed a novel approach to sense human activity combining WiFi-based and vision-based features, which has the potential to classify activity under unfavorable indoor environments such as occlusion and weak light. Singh et al. [49] proposed an approach to combine multiple CNN streams for human activity recognition. The inputs of the proposed network consist of dynamic images generated from RGB images and depth map from three different dimensions. Their method achieved state-of-the-art performance for the time on three human activity challenging datasets, i.e., MSR daily activity, UTD MHAD, and CAD 60. However, while these methods achieved promising results, they require data information from supplemental sensors. There still lacks a method to utilize objects and their mutual relationships in the image for activity classification, which generally enables a deep understanding of visual content.

Methodology

In this study, a multi-classifier decision fusion approach is proposed to integrate visual and scene graph information for the recognition of human activity in healthcare facilities. Figure 2.1 presents an overview of the methodology that consists of two steps. In the first step, Convolutional Neural Network (CNN), Visual Transformer (ViT), and Graph Neural Network (GNN) are adapted to classify human activity in images captured by the disinfection robot. The three classifiers have the potential to capture different information in images given completely different network architectures, thus a combination of them can complement each other and improve the reliability of activity recognition. Specifically, the state-of-the-art CNN network ConvNeXt [50] is adopted in this study. The CNN method is good at capturing spatial local patterns in an image. On the contrary, ViT method has been demonstrated to be powerful to capture global contextual information. In this study, Swin Transformer [51] is adapted, which has achieved great performance in image classification. The GNN is designed to classify scene graphs based on object and relationship features. The scene graph is generated using the unbiased scene graph generation approach [52], which contains rich human-object and human-human relationships. The outputs from these three classifiers are probability distributions (PDs), which give the likelihood of each activity category. In the second step, a hybrid multi-classifier fusion method integrating Dempster-Shafer theory and weighted majority vote is proposed to combine the outputs from Swin Transformer, ConvNeXt, and GNN. The conflict between the three PDs is estimated. If the conflict value is smaller than 0.6, Dempster-Shafer theory will be used. Otherwise, weighted majority vote will be used. The methodology is detailed in the following subsections.

Scene graph-based activity classification

The unbiased scene graph generation method developed in [52] is adopted to generate scene graphs for images. In this method, counterfactual causality is used to infer the effect of bad bias in the trained graph to be removed. The method is based on the casual inference with total direct effect analysis, which can be integrated with other scene graph generation models such as Iterative Message Passing (IMP) [53] and VCTree [54].

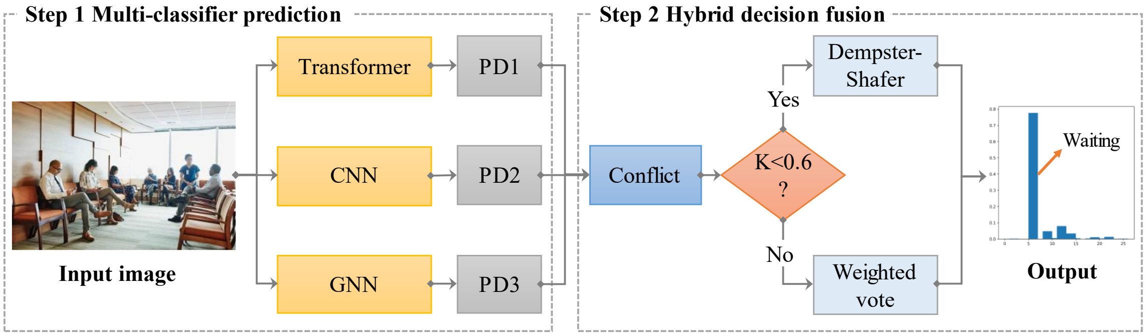


Figure 2.1. Methodology overview

In this study, MOTIFS model is selected, and it is trained on Visual Genome [55], which is a large benchmark for the scene graph generation. The trained model combined with casual inference is used to detect scene graphs in images.

To form a scene graph, the relationship pairs with confidence scores higher than 0.5 are selected as edges and corresponding objects as nodes. Furthermore, 300-dimensional and 100-dimensional vectors are generated for each node and edge, respectively based on predicted relationship and object categories. Word2Vec algorithms are used to generate vector representation for nodes and edges. The word embeddings are treated as initial node and edge features for scene graph classification. The image category is served as graph category. Figure 2.2 shows some example results of generated scene graphs for the images. Note that objects with top-10 confidence and top-10 relationship among these objects are selected to construct scene graph for a better illustration. In the classification task, more objects and relationships could be extracted deepening on image context. As indicated, the scene graph can capture some high-level relationships in the image, which can provide critical information for human activity recognition. For example, a group of doctors is sitting around the table for a meeting, which is reflected in the generated scene graph.

A graph classification network is designed to recognize human activity from scene graphs. Figure 2.3 shows the flowchart of the proposed GNN that consists of feature encoding and feature classifier. The graph attention mechanism is adapted to learn graph-structured data by leveraging attention over a node’s neighbor. The Graph Attention Network (GAT) was first introduced in 2017 by Veličković et al. [56], which then becomes one of the most popular GNN architectures and is considered the state-of-the-art architecture for representation learning with graphs. GAT learns the hidden features of graph nodes by computing the node similarity in a graph. In recent, a dynamic graph attention variant was proposed with a simple modification of the order of operations, which was demonstrated to be more effective in fitting the training data [17]. In this study, the graph attention mechanism is used to encode graph features.

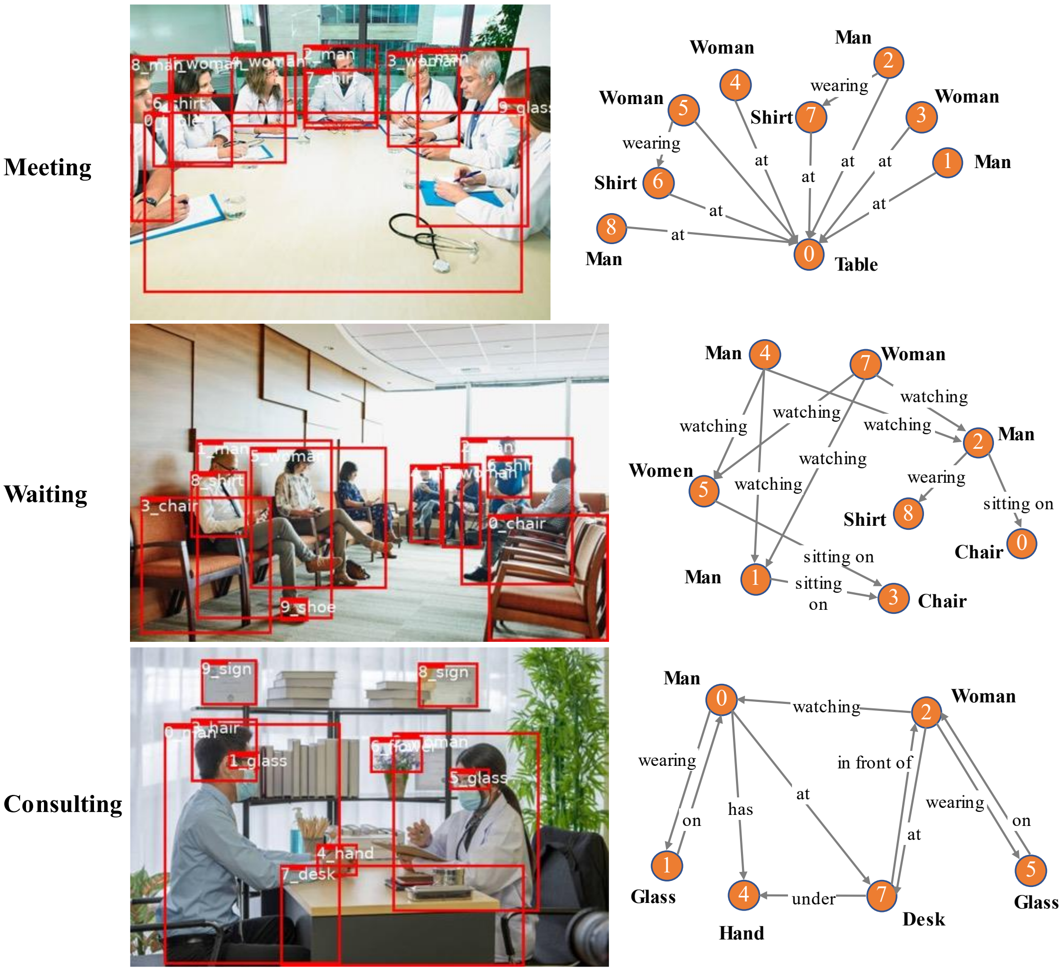


Figure 2.2. Example of generated scene graphs for images

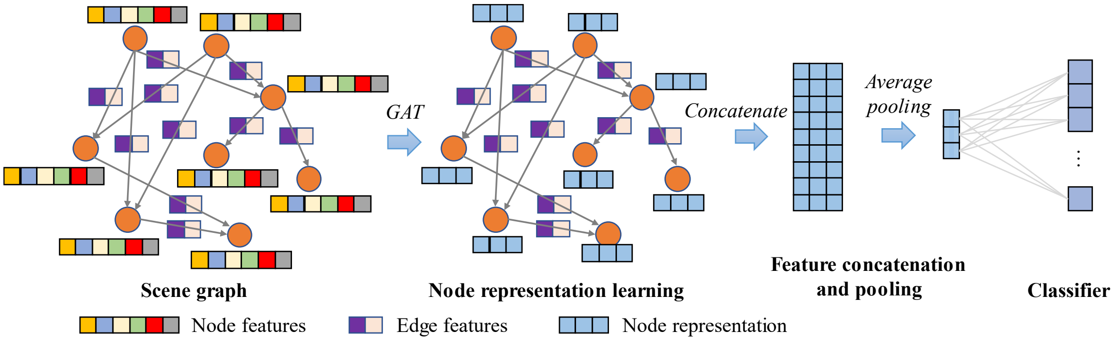


Figure 2.3. Architecture of the proposed GNN

The input of GAT operator is a set of node features $h = \{h_1, h_2, \dots, h_n\}, h_i \in \mathbb{R}^F$ and a set of edge features $e_{i,j} \in \mathbb{R}^D$, where N is the number of nodes in graph, *F* is the number of features in each node, *D* is the number of features in each edge, and $e_{i,j}$ only exists when there is an edge between node *i* and node *j*. The output of GAT operator is high-level features $h' = \{h'_1, h'_2, \dots, h'_N\},\ h'_i = \mathbb{R}^{F'}$. The GAT operator consists of three steps. First, a shared linear transformation method is applied to every node in the graph. Eq. (1) defines attention coefficient, where $\mathbf{W} \in \mathbb{R}^{F' \times (2F+D)}$ is the weight matrix, a μ_{ij} represents the importance of node *j*'s feature to node i, and the operator || represents the concatenation.

$$\mu_{ij} = \mathbf{W}\left(h_i \parallel h_j \parallel e_{i,j}\right) \tag{1}$$

The attention coefficients are only computed for first-order neighbor nodes $k \in \mathcal{N}_i$, which is designed to retain the topological information of the scene graph. The attention coefficients for each node are normalized using Eq. (2), where LeakyReLU is the activation function with a negative slope of 0.2, **a** $\mathbb{R}^{F'} \times \mathbb{R}^{F'}$ is the learnable weight vector.

$$\alpha_{ij} = \frac{\exp\left(\mathbf{a}^T \mathrm{LeakyReLU}(\mu_{ij})\right)}{\sum_{k \in \mathcal{N}_i} \exp\left(\mathbf{a}^T \mathrm{LeakyReLU}(\mu_{ik})\right)} \tag{2}$$

In this study, a multi-head attention mechanism is used to enrich the model capability and to stabilize the learning process. Specifically, we compute two different attention maps and then average their outputs in Eq. (3), Where *K* is the number of independent attention processes, σ is the ReLU activation function.

$$h'_i = \sigma\left(\frac{1}{K} \sum_{k=1}^{K} \sum_{j \in \mathcal{N}_i} \alpha_{ij}^{k} \mathbf{W}^{\boldsymbol{k}} h_j\right) \tag{3}$$

The output from the GAT operator is fed into a global average pooling layer to average node features across node dimensions. Finally, a fully connected layer is used to classify pooled graph features. The GNN has only 0.11 million parameters to learn, which is very fast for training and inference.

Transformer-based activity classification

Swin Transformer is adopted in this study to learn visual features for activity classification given its superior performance [51]. Figure 2.4 presents an overview of Swin_L architecture, which is a large version of Swin Transformer. The input image is first divided into multiple non-overlapping patches using a patch splitting module. The raw pixel values are served as features for each patch. A patch size of 4 × 4 is used in the network, resulting in a feature dimension of 48. The 48-dimensional feature is then fed into four different stages with transformer blocks. From shallow to deep stages, the number of channels doubles and the dimension of feature map reduces to half. Particularly, stage 1 is composed of a linear embedding layer and two swin transformer blocks. The linear embedding layer converts the 48-dimensional feature into a feature dimension of 192.

From stage 2 to 4, the patch merging module in each stage first reduces the dimension of feature map to half. It is based on feature concatenation on each group of 2 × 2 neighboring patches followed by a linear operation. The output from the patch merging layer is fed into the transformer block to generate a hierarchical representation. The number of swin transformer blocks are 2, 18, and 2 for stage 2 to 4, respectively. Specifically, the swin transformer block is built based on a shifted window-based multi-head self-attention (MSA) module. Two Multilayer Perceptron (MLP) layers are integrated which are connected by the GELU activation function. In addition, layer normalization is used before MSA and MLP layers.

CNN-based activity classification

The state-of-the-art ConvNeXt is selected as CNN-based activity classification method. The ConvNeXt is an improvement of the popular ResNet architecture with significant modifications. Figure 2.5 shows the architecture of ConvNeXt_L. The architecture of ConvNeXt_L consists of a convolutional operation, four stages with ConvNeXt blocks, and a classifier. The input image is first fed into a 4×4 convolution with a stride of 4, which is used to reduce the feature map dimension.

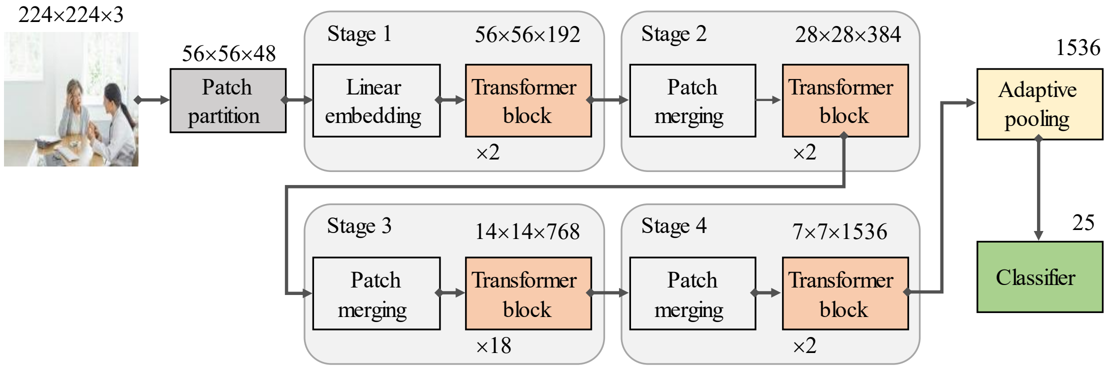


Figure 2.4. Flowchart of the Swin_L architecture

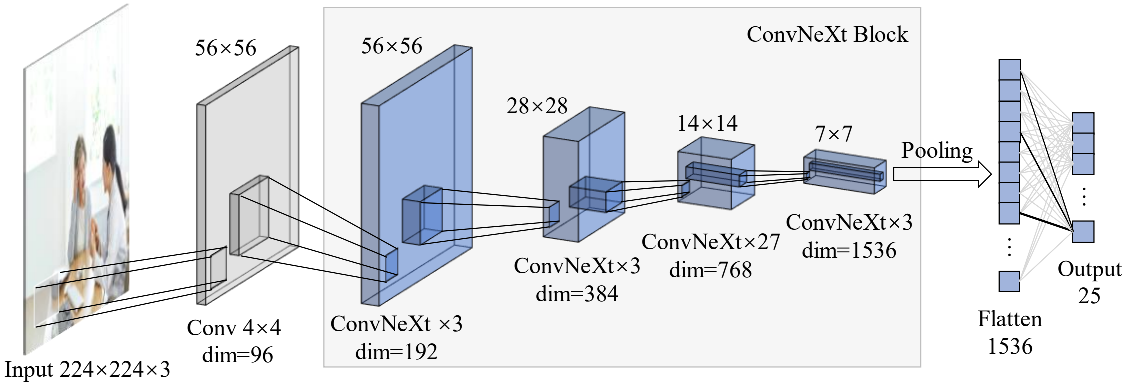


Figure 2.5. Flowchart of the ConvNeXt_L architecture

The number of ConvNeXt blocks for the four stages are 3, 3, 27, and 3, respectively. Note that the dimension of feature map reduces to half and the channel number doubles, as the network gets deeper. The output from the fourth stage is fed into an adaptive pooling layer to reduce the feature map dimension to 1×1. Finally, a fully connected layer is used to classify the extracted features. Readers are referred to [50] for a detailed description of the network.

Decision fusion

This section elaborates on the decision fusion approach aiming to combine the decisions from the GNN, Swin_L, and ConvNeXt_L into a common decision. A hybrid multi-classifier fusion approach combining Dempster-Shafer theory (DST) and weighted majority vote is proposed in this study. For the DST, a high conflict among classifiers could lead to counterintuitive results and make evidence fusion approach insignificant. The hybrid approach could alleviate such an issue with a combination of weighted majority vote.

The DST proposed in [58] is adapted for the decision fusion in our study. Different from conventional DST, the classification abilities of each classifier are integrated to provide reliable information for the classifier. The advantage of the method is the ability to punish overconfident and overtrained classifiers. Specifically, the classification probabilities of each classifier on the training set are used as the prior knowledge, which is represented as decision template for each class. The decision template is defined as the most typical profile in the classification outputs. Eq. (4) defines the decision template DT_j for class w_j, where **Z** represents the dataset, N_j is the number of elements of **Z** from w_j, *DP* is the decision outputs from the classifier with probabilities, and *c* is the total number of classes.

$$DT_j = \frac{1}{N_j}\sum_{\boldsymbol{z}_k \in \boldsymbol{w}_j, \boldsymbol{z}_k \in \boldsymbol{Z}} DP(\boldsymbol{z}_k) \quad j = 1,2, \dots, c \tag{4}$$

The decision profile *DP*(x) and decision template DT_j are *L*x*c*-dimensional matrices. *L* is the number of classifiers. Let DT_j^i denote the *i*th row the decision template DT_j.

Let $D_i(\mathbf{x})$ = [$d_{i,1}$(x), …, $d_{i,c}$(x)] denote the *i*th row of the decision profile *DP*(**x**) from the classifier The proximity between DT_j^i and the output of classifier D_i for the input **x** is given in Eq. (5), where ||·|| represents L1 norm, and $\phi_{j,i}(\mathbf{x})$ is the proximity for class *j* = 1, 2, …, *c* and classifier *i* = 1, 2, …, *L*.

$$\phi_{j,i}(\mathbf{x}) = \frac{\left(1+\left\|DT_j^i - D_i(\mathbf{x})\right\|\right)^{-1}}{\sum_{k=1}^{c}\left(1+\left\|DT_k^i - D_i(\mathbf{x})\right\|\right)^{-1}} \tag{5}$$

The belief degree can be calculated using Eq. (6).

$$b_j\left(D_i(\mathbf{x})\right) = \frac{\phi_{j,i}(\mathbf{x})\prod_{k\neq j}\left(1-\phi_{k,i}(\mathbf{x})\right)}{1-\phi_{j,i}(\mathbf{x})\prod_{k\neq j}\left(1-\phi_{k,i}(\mathbf{x})\right)} \tag{6}$$

The final degrees of support are given in Eq. (7), where β is the normalizing constant.

$$\mu_j(\mathbf{x}) = \beta\prod_{i=1}^{L} b_j\left(D_i(\mathbf{x})\right) \tag{7}$$

The weighted majority vote will replace DST for high conflict decisions to improve the reliability of evidence fusion approach. The conflict value *K* is defined in Eq. (8), where $D_i^{k_i}$ denotes the *ith* row and *k*th column of the decision profile *DP*(**x**).

$$K = \sum_{k_1\cap\ldots\cap k_c=\emptyset} D_1^{k_1}(\mathbf{x})\cdot D_2^{k_2}(\mathbf{x})\cdot\ldots\cdot D_L^{k_c}(\mathbf{x}) \tag{8}$$

Under the assumption that conflict value K greater than 0.6 indicates a large conflict between different classifiers [59], the weighted majority vote is given in Eq. (9), where **M** is the 1 x *L* weight matrix calculated based on the accuracy of classifier on the validation set, $DP^k(\mathbf{x})$ represents *k*th column of the decision profile *DP*(**x**). A high classification accuracy on the validation set indicates the reliability of the classifier, which will have a higher weight in the final decision fusion.

$$\mu_j(\mathbf{x}) = \mathbf{M}\times DP^k(\mathbf{x}) \tag{9}$$

We use a synthetic simple example with three classes and three classifiers to illustrate the process of decision fusion. The three decision templates are given in Eq. (10).

$$DT_1 = \begin{bmatrix}0.6 & 0.3 & 0.1\\0.7 & 0.2 & 0.1\\0.8 & 0.1 & 0.1\end{bmatrix}\ DT_2 = \begin{bmatrix}0.2 & 0.6 & 0.2\\0.1 & 0.7 & 0.2\\0.2 & 0.7 & 0.1\end{bmatrix}\ DT_3 = \begin{bmatrix}0.1 & 0.3 & 0.6\\0.2 & 0.2 & 0.6\\0.2 & 0.1 & 0.7\end{bmatrix} \tag{10}$$

The decision profile for input **x** the three classifiers is given in Eq. (11)

$$DP(\mathbf{x}) = \begin{bmatrix}0.4 & 0.4 & 0.2\\0.3 & 0.6 & 0.1\\0.4 & 0.1 & 0.5\end{bmatrix} \tag{11}$$

The conflict value K is calculated as 0.55 using Eq. (8), which is smaller than 0.6. In this case, DST will be used. Using Eq. (5), the proximities for each decision template are calculated and shown in Table 2.1.

Table 2.2 presents the belief degrees and final degrees of support calculated using Eqs. (6) and (7). In this example, the fusion method gives a slight preference toward w_2.

Experiment and results

Data description

The construction of hospital activity dataset is composed of two steps that are image downloading and data cleaning. We select 25 common activities that occur in healthcare facilities, which could have different implications for robot disinfection. For example, when doctor meeting or patient consultations is going on in the room, the robot should better move to next places needing disinfection instead of interrupting human activities. For short-duration activities such as injecting and checking temperature, the disinfection robot can wait until these activities are finished and then proceed to disinfect the room.

In the first step, images were downloaded from Getty and Shutterstock, which are two popular online stock photography websites with hundreds of millions of images. The image on the website is uploaded with textual description written by the image composer based on visual content. As such, candidate images related to hospital human activities can be downloaded using relevant query words. Initially, around 1,000 images were downloaded for each human activity. In the second step, a verification task was performed to clean and refine the dataset. Specifically, if images do not belong to the category, they will be deleted or moved to the correct class. Low-quality images like blurry or images with unobvious human activity are also removed from the dataset. Furthermore, images within each category have large variations in background, human pose, and appearance. The resulting positive images were sent for further verification by another human labeler to ensure the quality of the dataset. Figure 2.6 shows some examples of images.

Table 2.1. Proximity measures

Class	$\phi_{j,1}(\mathbf{x})$	$\phi_{j,2}(\mathbf{x})$	$\phi_{j,3}(\mathbf{x})$
w_1	0.3600	0.3139	0.3222
w_2	0.3600	0.4036	0.2636
w_3	0.2800	0.2825	0.4142

Table 2.2. Belief and final degrees of support

Class	$b_j\left(D_1(\mathbf{x})\right)$	$b_j\left(D_2(\mathbf{x})\right)$	$b_j\left(D_3(\mathbf{x})\right)$	$\mu_j(\mathbf{x})$
w_1	0.2058	0.1637	0.1701	0.3352
w_2	0.2058	0.2499	0.1244	0.3741
w_3	0.1374	0.1388	0.2609	0.2907

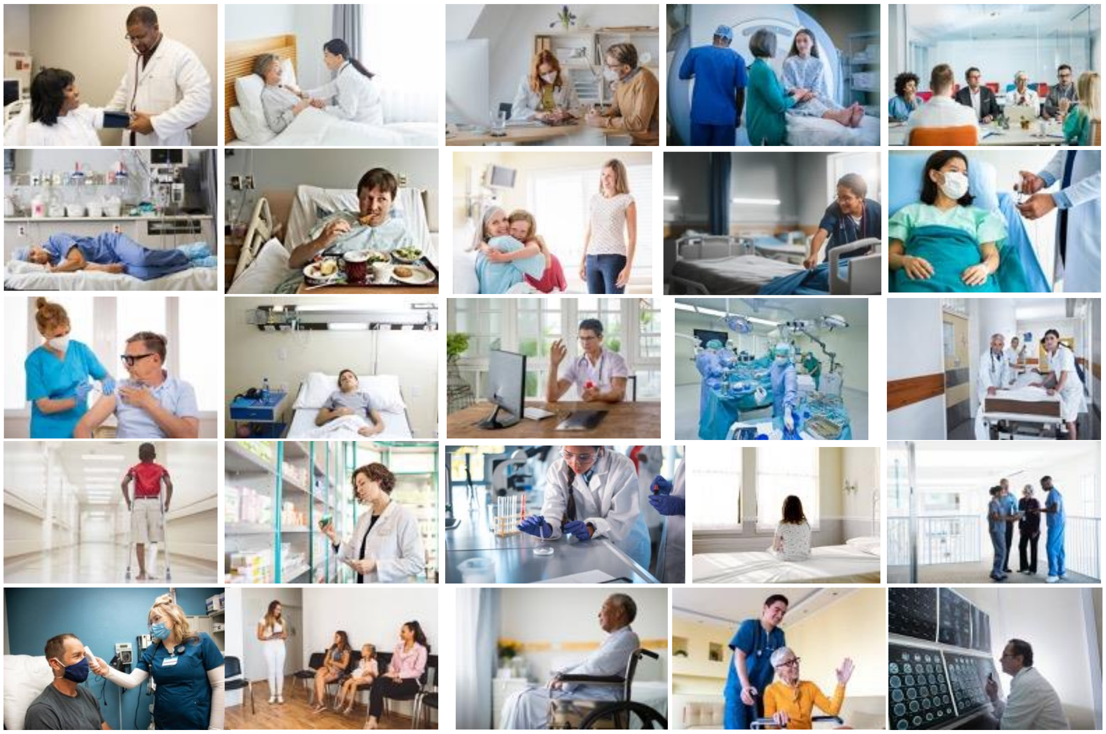


Figure 2.6. Example images in hospital human activity dataset

We collected 180∼396 images for each class. In total, 5,770 images were collected in the dataset. Table 2.3 presents the number of images for each class with a brief description.

To further analyze the similarities between different activities, Barnes-Hut t-SNE [60] algorithm is adopted to visualize raw image features in 2D space. t-SNE is a nonlinear dimensionality reduction technique, which is used to embed high-dimensional data into lower-dimensional data. t-SNE starts by converting similarity between image samples to conditional probabilities. Conditional probabilities of low-dimensional embedding are identified by minimizing the Kullback-Leibler divergence with high-dimensional data. Since the number of image features is very high, principal component analysis (PCA) is first adopted to reduce its dimension to 50 as recommended in [61]. The 50-dimensional features are then fed into t-SNE algorithm to convert them into 2D dimension. Note that input images are resized to 224 × 224 with three channels, resulting in a total number of 150,528 features for each image. The perplexity value is set to 30.

Figure 2.7 shows t-SNE visualization of hospital activity dataset in 2D space. It is apparent from this figure that image features for different activities are overlapped with each other. This phenomenon indicates that images are very similar across human activities, which could be a challenge for the robot to recognize. The similarity is mainly attributed to that those images are collected within healthcare facilities with similar environmental backgrounds.

Implementation details

The deep learning networks were constructed using PyTorch [62] and trained on an Ubuntu 16.04 workstation using dual NVIDIA Quadro P5000. The pretrained weights on ImageNet were used to initialize Swin_L and ConvNeXt_L. The training hyperparameters for GNN, Swin_L, and ConvNeXt_L are set to be the same. Specifically, Stochastic Gradient Descent (SGD) [63] was used to optimize the network. The weight decay and momentum for SGD optimizer were set to 5e-4 and 0.9, respectively.

Table 2.3. Data statistics

Activity	**Description**	**Shortened**	**Number of people**	**Count**
Measuring blood pressure	Medical staff is helping a patient to measure blood pressure.	Measuring	2	250
Comforting	Medical staff is comforting or expressing concern to patients.	Comforting	2 and above	189
Carrying patients	Critically ill patients are being carried by first responders, doctors, or nurses.	Carrying	2 and above	231
Consulting	Patients are consulting a professional doctor or nurse about their condition.	Consulting	2 and above	197
Scanning	Doctors are helping patients with MRIs or full-body scans in large instruments.	Scanning	1 and above	220
Doctor meeting	Doctors are doing in-person academic meetings around the table	Meeting	2 and above	203
Analyzing samples	Doctors are analyzing blood or drug sample in hospital laboratory analysis room.	Analyzing	1 and above	247
Doctor sleeping	Doctors or first responders are temporarily sleeping and resting in hospital.	Sleeping	1 and above	293
Eating	Inpatients in wheelchairs or beds are eating.	Eating	1 and 2	198

Table 2.3. Continued

Family visiting	Family members are visiting a patient who lying or sitting on the bed.	Visiting	3 and above	202
Cleaning	Cleaning staff is disinfecting and cleaning within a hospital.	Cleaning	1 and above	262
Patient sitting in a wheelchair	A patient is sitting alone in a wheelchair.	Wheelchair_1	1	253
Patient sitting in a wheelchair with assistance	A patient is sitting in a wheelchair with an or multiple assistant or nurse next to it.	Wheelchair_2	2 and above	202
Infusing	A patient is being infused while lying or sitting on the bed.	Infusing	1 and above	165
Injecting	A patient is being injected by the doctor.	Injecting	1 and above	197
Lying in a bed	Patients are lying in the nursing bed and might be questioned by medical staff.	Lying	1 and above	203
Online meeting	A doctor or paramedic is meeting online using a computer.	O-meeting	1	205
Performing surgery	Doctors and nurses are performing major surgery on patients in a professional operating room.	Operating	2 and above	193
Patient walking	A patient is walking alone in hospital hallway or with one/couple of Nursing staff.	Walking	1 and above	180
Working in pharmacy	Doctors in pharmacy are sorting, helping clients, or recording.	Working	1 and above	248

Table 2.3. Continued

Patient sitting on a bed	Patient or doctor sitting on the bed in a hospital.	Sitting	1	194
Discussing	A group of doctors or paramedics is standing together and discussing.	Discussing	2 and above	181
Checking temperature	Nursing staff is using thermometer guns to measure the temperature of the patients.	Checking	2 and above	343
Patient waiting	Patients are waiting in a waiting room or hallway for treatment or examination.	Waiting	1 and above	396
Examining x-ray	Doctors are examining the X-ray pictures by himself/herself or with a colleague or nurse.	Examining	1 and 2	318

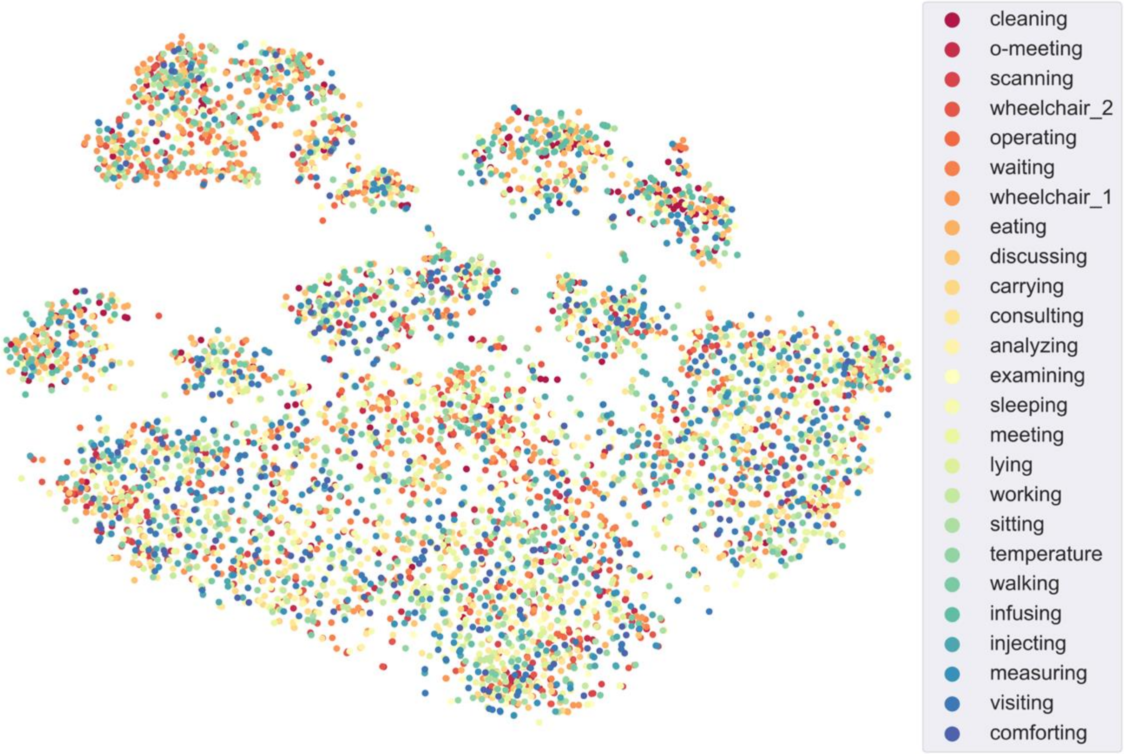


Figure 2.7. t-SNE visualization of hospital activity datasets. Each point represents an image, and its associated color represents activity category

The initial learning rate of SGD was 1e-3 and decreases by half for every 20 epochs. The batch size was set to 24, and the number of training epochs was 50. The cross-entropy loss was selected to adjust model weights during training, which is widely used as the loss function when optimizing classification models. The hospital activity dataset was randomly split into 80% training, 10% validation, and 10% testing. In total, five random splits were generated to evaluate the performance of the proposed method. The best performance reported on the validation set is saved for subsequent analysis. The prediction results on the validation set are used as training data in the process of decision fusion, and the final fusion is performed and evaluated on the test dataset.

Evaluation metrics

In this study, accuracy, recall, precision, and F1 metrics are used to evaluate the performance of the proposed method. The classification accuracy is calculated as the ratio of the number of correct predictions to the total number of samples and is given in Eq. (12), where TP denotes true positives (i.e., the prediction and ground-truth are both positives); TN denotes true negatives (i.e., the prediction and ground-truth are both negatives); FP represents false positives (i.e., ground-truth is negative, but the prediction is positive); FN represents false negatives (i.e., ground-truth is positive, but the prediction is negative)

$$\text{accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{12}$$

Precision is used to measure the correctly classified positive against the total number of classified positives and is defined in Eq. (13). Recall measures the predictive power of the network in identifying all the positive elements and is given in Eq. (14).

$$\text{precision} = \frac{TP}{TP+FP} \tag{13}$$

$$\text{recall} = \frac{TP}{TP+FN} \tag{14}$$

F1 metric combines the precision and recall by taking an evenly harmonic mean of them and is defined in Eq. (15).

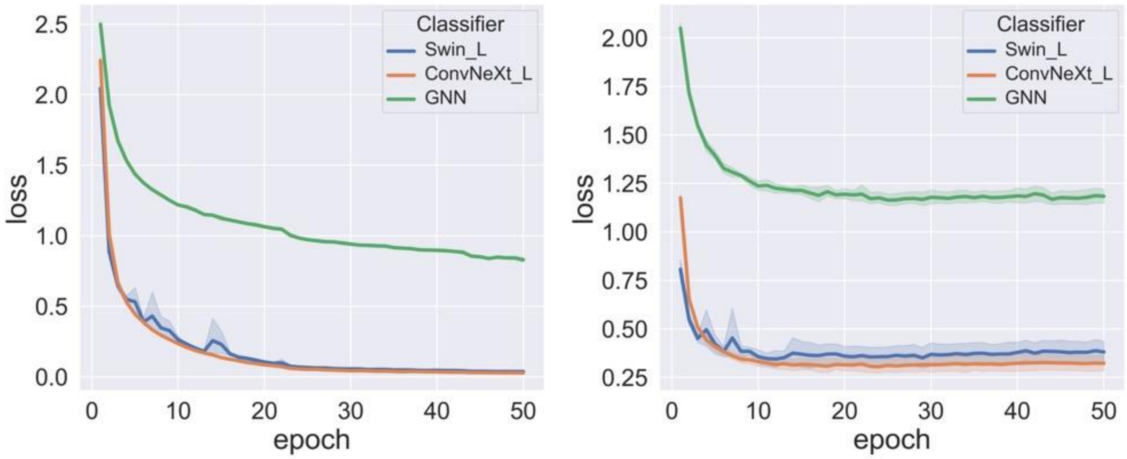
$$\text{F1} = 2 \times \frac{\text{precision}\times\text{recall}}{\text{precision}+\text{recall}} \tag{15}$$

Results

This section elaborates on experimental results. Figure 2.8 presents the variation of loss values on training and validation sets during network training for Swin_L, ConvNeXt_L, and GNN. On the training dataset, the loss curves for ConvNeXt_L and Swin_L are nearly coincident. In addition, ConvNeXt_L and Swin_L reach very small loss values, and become stable around 30 epochs. The small loss indicates the capability of the network to learn visual features for action representation. The loss value for GNN is greater than that of ConvNeXt_L and Swin_L, but is continuously dropping within 50 epochs. With the increasing number of epochs, the loss values on the training set could be much smaller. On the validation dataset, the ConvNeXt_L has a relatively smaller loss value compared to that of Swin_L, and becomes stable around 15 epochs. The loss values for the GNN become stable around 20 epochs. The loss curves indicate that GNN has a relatively lower performance compared to ConvNeXt_L and Swin_L.

Table 2.4 presents the model performance for each classifier and the proposed method on the testing set. Note that the metrics reported in the table represent the mean and standard deviation, which are calculated based on the results of five random splits. The ConvNeXt_L achieves the best performance with overall accuracy, F1, precision, and recall of 89.51%, 89.50%, 90.17%, and 89.30%, respectively. The proposed graph neural network (GNN) achieves the lowest performance among the three classifiers with overall accuracy, F1, precision, and recall of 63.84%, 62.78%, 63.54%, and 63.16%, respectively. The relatively low performance of GNN can be explained as follows. Our study focuses on human activities in healthcare facilities, and thus environmental backgrounds can be very similar across different categories. The scene graph is a structured representation of an image that consists of objects and relationships between objects in the scene, which could be very similar due to similar environment backgrounds.

The confusion matrix of the three classifiers on the testing dataset is created. Note that the results on the five random splits are added together to create the confusion matrix.



(a) Training

(b) Validation

Figure 2.8. The loss variation on the training and validation datasets during training. The solid lines are average over five random splits and translucent bands indicate the range

Table 2.4. Model performance on hospital human activity testing dataset

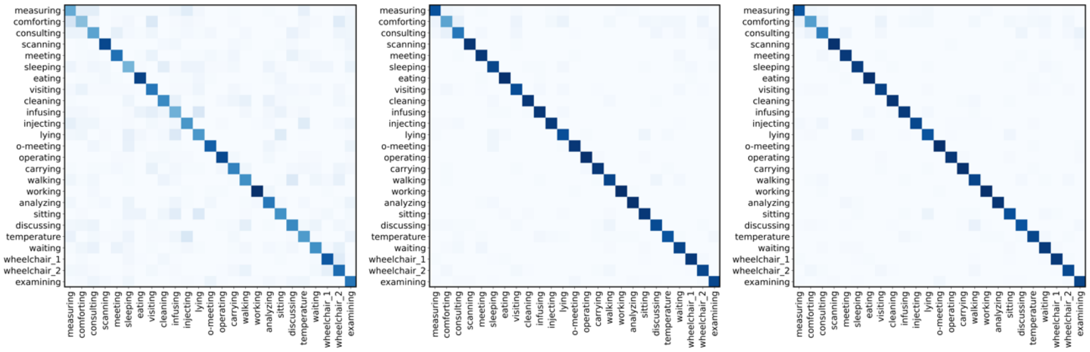
Model	Accuracy		F1		Precision		Recall	
	mean	st.d.	mean	st.d.	mean	st.d.	mean	st.d.
GNN	63.84	2.33	62.78	2.21	63.54	2.02	63.16	2.20
Swin_L	89.34	1.19	89.27	1.25	90.00	1.06	89.00	1.36
ConvNeXt_L	89.51	0.91	89.50	0.98	90.17	1.01	89.30	0.94
Proposed	**90.59**	1.18	**90.54**	1.33	**91.16**	1.22	**90.31**	1.38

Figure 2.9 presents the confusion matrix of the three classifiers. The confusion matrix is normalized by the row, and the diagonal can be used to represent the model's predictive power on positive classes. The recall for GNN has a large variation for different human activities. In particular, measuring blood pressure and comforting patients are the two most misclassified activities with a recall of 0.26 and 0.33, respectively. With respect to Swin_L, the model achieves the best performance on six activities that are infusing, operating, carrying, pharmacy, waiting, and sitting with a recall of 1. For the ConvNeXt_L, a recall of 1 is achieved for activities including infusing, online meeting, operating, carrying, pharmacy, analyzing, and waiting. This indicates that Swin_L and ConvNeXt_L can learn distinct features in images, which could lead to different performances.

The proposed method achieves an overall accuracy, F1, precision, recall of 90.59%, 90.54%, 91.16%, and 90.31% by averaging results on the testing set of five random splits, which has an improvement of 1.08%, 1.04%, 0.99%, and 1.01% for accuracy, F1, precision, and recall, respectively compared to ConvNeXt_L. Figure 2.10 presents the confusion matrix of the proposed method, resulting in recall values between 0.62 and 1. In particular Comforting patients achieves the lowest performance with a recall of 0.62. This may be attributed to that comforting patient activity is typically the interaction between a doctor and a patient, which is similar to activities such as infusing and injecting.

Ablation study

This section conducts an ablation study to evaluate the performance of a combination of either of the two classifiers. The proposed hybrid decision fusion method is applied to the combination of two classifiers. Table 2.5 summarizes the results. The results indicate that a combination of Swin_L and ConvNeXt_L achieves an overall accuracy of 90.49%, which is an improvement of 1.15% and 0.98% from Swin_L and ConvNeXt_L. In addition, a combination of ConvNeXt_L and GNN results in an accuracy improvement of 0.1% compared to ConvNeXt_L.



(a) Graph neural network

(b) Swin_L

(c) ConvNeXt_L

Figure 2.9. Confusion matrix of three classifiers: (a) graph neural network; (b) Swin_L; and (c) ConvNeXt_L

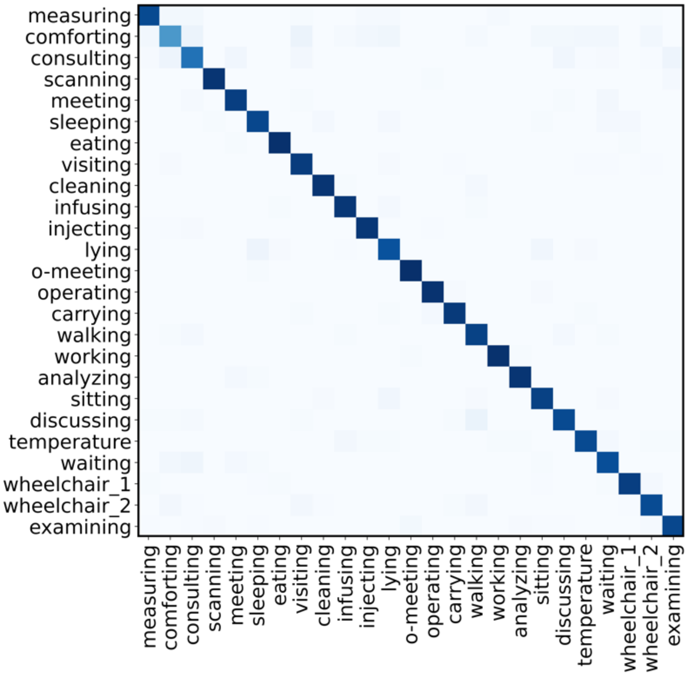


Figure 2.10. Confusion matrix of the proposed decision fusion method

Table 2.5. Results of ablation study

Model	Accuracy		F1		Precision		Recall	
	mean	st.d.	mean	st.d.	mean	st.d.	mean	st.d.
GNN + Swin_L	89.47	1.00	89.43	1.07	90.16	0.95	89.12	1.17
GNN + ConvNeXt_L	89.61	0.93	89.54	0.98	90.09	1.05	89.36	0.95
Swin_L + ConvNeXt_L	90.49	1.16	90.44	1.30	91.11	1.15	90.21	1.35
Proposed	**90.59**	1.18	**90.54**	1.33	**91.16**	1.22	**90.31**	1.38

The fusion of GNN and Swin_L outputs achieves an accuracy improvement of 0.13% compared to Swin_L. Furthermore, the proposed method with a combination of three classifiers achieves the best performance. Therefore, it can be concluded that objects and object relationships in scene graphs can improve classification accuracy. Furthermore, the proposed hybrid decision fusion approach effectively combines outputs from diverse classifiers and achieves a better performance.

Discussion

Human activity recognition in healthcare facilities is critical for the robot to efficiently conduct disinfection tasks within mass-gathering facilities. The core idea behind this study is to develop a new method to enable the robot to recognize human activity in hospitals. To this end, we first prepared the first hospital human activity dataset using images, which cover a variety of activities. Then, we developed a workflow to generate scene graphs with node and edge attributes from images and designed a GNN architecture for scene graph classification. Finally, a novel hybrid multi-classifier decision fusion was proposed to combine outputs from GNN, Swin_L, and ConvNeXt_L and achieved promising results on the hospital activity dataset. The proposed method has great potential to facilitate the deployment of disinfection robots in the presence of humans in healthcare facilities. The following sections discuss how robots could use human context information to make informed disinfection decisions, compare our approach with state-of-the-art methods, and discuss limitations and future studies.

Robot implementation

The proposed human activity recognition method enables the robot to understand the human context in the healthcare facilities, which can be integrated into existing disinfection robot platforms to guide robot disinfection decisions. In this study, the robot decision is divided into three categories that are “disinfect”, “wait”, and “leave”. For “disinfect” decision, the robot will greet the patient and family or visitors, give a brief introduction, and notify the time needed to disinfect the room. For “wait” decision, the robot will wait until humans finish their activities and then proceed with “disinfect” decision. For “leave” decision, the robot will move to next places needing disinfection instead of interrupting human activities.

The robot decision is characterized based on surface cleaning and disinfection procedures and techniques in healthcare facilities. For example, if doctor, nurse, or other clinical person is working with patients, the robot should “leave” and go back later if it is a long-duration human activity, otherwise, the robot can “wait” if it is a short-duration activity. For an activity like eating, the robot should also better move to the next places needing disinfection. For activities such as patient waiting, lying in a bed, or family visiting, the robot can directly proceed with disinfection after greeting, introduction, and notification. Table 2.6 summarize the robot disinfection for each human activity. It should be noted that the robot needs to observe human activity for a certain duration to ensure it is accurately predicted, thus making informed disinfection decision.

Comparison to state-of-the-art methods

In this section, the proposed method is evaluated by comparing it to other methods on two action datasets, which are described as follows.

Stanford 40 action dataset. This dataset contains 9,532 images with 40 action categories. These actions are daily human activities, such as running, walking, and applauding, and each of these activities includes 180 to 300 images. The provided train-test split is used, i.e., training set contains 100 images per action.

PASCAL VOC 2012. The dataset contains a total of 10 human actions, such as playing an instrument, riding a bike, and using a computer. The original dataset consists of 2,296 training and 2,292 validation images. To be consistent with compared methods, the model performance is evaluated on the validation set. Note that images with multiple actions are excluded in the training and validation dataset, and our networks are trained and tested with images associated with a single action.

Table 2.6. Robot disinfection decision based on human activities

Activity	**Robot decision**
Measuring blood pressure	Wait
Comforting	Disinfect
Carrying patients	Wait
Consulting	Leave
Scanning	Leave
Doctor meeting	Leave
Analyzing samples	Leave
Doctor sleeping	Disinfect
Eating	Leave
Family visiting	Disinfect
Cleaning	Leave
Patient sitting in a wheelchair	Disinfect
Patient sitting in a wheelchair with assistance	Disinfect
Infusing	Disinfect
Injecting	Wait
Lying in a bed	Disinfect
Online meeting	Leave
Performing surgery	Leave
Patient walking	Disinfect
Working in pharmacy	Disinfect
Patient sitting on a bed	Disinfect
Discussing	Wait
Checking temperature	Wait
Patient waiting	Disinfect
Examining x-ray	Leave

Performance on Stanford 40 action dataset

Most of the existing action recognition methods incorporated bounding-box annotations, which are provided by the Stanford 40 action dataset. For example, R*CNN, Top-Down Pyramid, and Semantic Part Action detected contextual information and semantic parts around or within person bounding boxes. Some other methods such as person detection and Multiple Spatial Clues Network (MSCNet) needs to detect action-specific human pose, human body regions, or action-specific parts. Whereas our method neither requires bounding-box annotations nor detecting humans and objects in images. Instead, our method automatically learns visual and graph features for action representation. For a fair comparison, the same metric mean average precision (mAP) is used. Table 2.7 presents the comparison of our method to SotA methods on Stanford 40 action, and our method achieves the best performance with an mAP of 96.6%. In particular, our model has an improvement of 0.4% compared to SotA Attend and Guide which requires keypoints detection to generate semantic regions. The MSCNet achieved the second-best performance among compared methods with an mAP of 94.6%, which, however, needs to learn human bodies and action-specific semantic parts. Compared to the MSCNet, our method significantly improves mAP by 2%. Compared to the rest of the methods, our model gains a significant margin with a minimum improvement of 5.4%.

Performance on PASCAL VOC 2012 action dataset

Table 2.8 shows the comparison of our methods with SotA methods on PASCAL VOC 2012 action dataset. As mentioned above, many existing action recognition methods require manually annotated bounding boxes to increase predictive power. For example, R*CNN achieved an mAP of 87.9% with annotated human bounding box. The performance decreased to 84.9% without integrating the manual annotations. However, such annotation is not very practical in real-world implementations.

The performance on PASCAL VOC 2012 dataset has not achieved significant improvement until the Top-Down + Bottom-Up Attention network [46] is proposed, which achieved an mAP of 95.0%.

Table 2.7. Comparison to state-of-the-art methods on Stanford 40 action dataset

Method	mAP (%)
R*CNN (2015) [64]	90.9
Person Detection (2015) [43]	75.4
Action Masks (2016) [65]	82.6
Top-Down Pyramid (2016) [66]	80.6
AttSPP-Net (2017) [67]	81.6
Semantic Part Action (2017) [42]	91.2
Multi-Branch Attention (2017) [68]	85.2
Attend and Guide (2021) [45]	96.2
Top-Down + Bottom-Up Attention (2022) [46]	91.0
MSCNet (2022) [69]	94.6
Ours	**96.6**

Table 2.8. Comparison to state-of-the-art methods on PASCAL 2012 action dataset

Method	mAP (%)
R*CNN (2015) [64]	87.9
Action Part (2015) [70]	80.4
Action Masks (2016) [65]	82.2
AttSPP-Net (2017) [67]	76.2
Semantic Part Action (2017) [42]	90.0
Generalized Symmetric Pair Model (2017) [34]	71.1
Multi-Branch Attention Network (2017) [68]	87.1
Top-Down + Bottom-Up Attention (2022) [46]	95.0
Ours	**95.0**

Our method achieves the same level of performance as the Top-Down + Bottom-Up Attention network, which can be viewed as SotA methods. On the other hand, our method achieves a much better performance on Stanford 40 action dataset with an improvement of 5.6% compared to the Top-Down + Bottom-Up Attention network. The comparison with SotA methods demonstrates the efficiency of our method for human action recognition.

Limitation and future studies

There remain limitations in this study that could be addressed in future studies. First, scene graphs extracted from images contain some irrelevant information such as "woman wearing shirt" and "man has hand", which could have an impact on the performance of GNN. This is because scene graph generation method was trained on the large Visual Genome dataset with dense annotations of objects and relationships in each image. A natural progression of scene graph classification is to identify what kind of objects and relationships are indispensable and what are non-essential. Second, graph classification network is built based on a single graph attention network. It remains a challenge to train deep graph neural networks due to two main reasons: node features become very similar and hard to distinguish when the network gets deeper [72]; another reason is the bottleneck could result in over-squashing of exponentially growing amount of information in a fixed-sized node representation [73]. Increasing the depth of graph neural networks for better performance would be a fruitful area for further work. Third, since the main purpose and scope of this study is the creation and evaluation of the human activity recognition algorithm, how should the disinfection robot respond to different activities is not systematically studied. In the future, the interaction between humans and robots needs to be investigated to improve disinfection efficiency.

Conclusion

This paper set out to propose a hybrid multi-classifier decision fusion method for recognizing human activity with a particular emphasis on integrating graph and visual features. A new hospital activity dataset is provided, which contains a total of 5,770 images with 25 activity categories. The scene graph generation method was adapted to generate scene graphs for images. Scene graphs were then fed into a graph neural network to train a graph-based classification model. Swin_L and ConvNeXt_L were trained to classify human activity based on visual features in images. The decision outputs from the graph-based classification model, Swin_L, and ConvNeXt_L were combined using the proposed hybrid decision fusion approach, which integrates Dempster-Shafer theory and weighted majority vote. The proposed method achieved an accuracy of 90.59%, F1 of 90.54%, recall of 90.16%, and precision of 90.31%. Furthermore, compared to other methods, the proposed human activity recognition method achieved state-of-the-art performance.

CHAPTER THREE
OBJECT DETECTION AND MAPPING

Introduction

In healthcare facilities, it has been demonstrated that contaminated environmental surfaces and equipment are linked to pathogen transmission, leading to nosocomial outbreaks of infectious diseases. Researchers have focused on investigating the role of inanimate objects nearby patients in infectious disease transmission. It is now well established from a variety of studies that surfaces in the immediate vicinity of a patient can be a reservoir for nosocomial pathogens [74]. Infectious pathogens can transmit directly or indirectly to surfaces near patients from the hands of a healthcare worker. The cleaning staff is the main force to clean high-touch objects in healthcare facilities to limit the spread of fomite transmission. However, manual cleaning can be insufficient due to fatigue and other human behavior factors [75]. In addition, cleaning staff are exposed to an infectious environment and may have direct contact with these high-touch objects, causing a relatively high infection risk. Disinfection robots have been deployed in infrastructure facilities, achieving promising environmental surface disinfection results.

Despite the great potential of robots for environmental surface disinfection, existing disinfection robot platforms are incapable of recognizing high-touch objects and mapping these objects to a 3D map in healthcare facilities, which significantly compromises their disinfection efficiency. The lack of such a solution stems from two challenges. First, while many image datasets have been created for object detection in a variety of scenarios, few datasets, if any, were developed specifically for high-touch object recognition tasks in healthcare facilities. Recognizing high-touch objects is the basis of the disinfection efficiency of the robots, which can pinpoint high-transmission-risk surfaces requiring an adequate dosage of disinfectant. Second, how to map object detection results in RGB images to a 3D map that can be used for robot navigation remains a challenge to be addressed.

To address these challenges, this paper developed a deep learning-based method to detect and classify objects in healthcare facilities and project detected objects onto a 3D map for robot navigation. The contribution of this research is three-fold. First, a new image dataset with well-classified object categories in healthcare facilities is proposed, providing a benchmark for the task of object recognition in healthcare environments. Note that high-touch objects in healthcare facilities are annotated in the newly created dataset. Second, a novel deep learning-based object recognition method was designed by integrating spatial and channel attention mechanisms into the YOLOv5 architecture to ensure both accuracy and computational efficiency. Third, a novel mapping framework consisting of object detection, object coordinate estimation, and object clustering was developed to project detected objects onto a 3D map that can be used for robot navigation. The developed method was tested and validated on real data collected from a campus building at the University of Tennessee, Knoxville, demonstrating its feasibility and applicability.

Literature review

Related studies on object recognition in healthcare facilities

This section reviews algorithms and datasets for the task of object recognition in healthcare facilities. A number of datasets have been created for object detection, such as Microsoft COCO, Open Images, and PASCAL VOC2007. These datasets consist of both indoor and outdoor environments and have been widely used as benchmarks to evaluate the performance of deep learning networks. However, scant few datasets have been specifically developed for object recognition in healthcare facilities. In recent years, Bashiri et al. [76] proposed an object classification dataset named MCIndoor20000 using images collected from the Marshfield Clinic. The dataset contains a total of 2,055 images with three object categories: doors, stairs, and hospital signs. More recently, Ismail et al. [77] created an image classification dataset (MYNursingHome) using a total of 37,500 images collected in several nursing homes. The dataset contains 25 indoor object categories such as basket bin, bench, cabinet, chair, and wheelchair. The main drawback of MCIndoor20000 and MYNursingHome datasets is that surrounding backgrounds and objects were removed for each object category. As such, every image contains only one object category, which is unsuitable for object detection in a cluttered indoor environment.

With the advancement of computational power, deep learning methods have been widely used in computer vision tasks. These methods have been shown to be effective in a variety of fields such as material recognition and affordance segmentation. For object detection in healthcare facilities, Vasquez et al. [78] integrated a fast region proposal method into a Fast R-CNN network to increase object detection efficiency and speed. The network showed promise in detecting patients with mobility aids such as patients with a wheelchair. The results of object detection were further smoothed using a probabilistic position, velocity, and class estimator generated using a hidden Markov model.
Kinash et al. [79] designed a "you only look once" (YOLO)-based object detector to identify hospital beds. A centroid tracking approach was proposed to address the low-confidence detection by calculating displacement compared to the object size. Lie et al. [80] adopted YOLOv2 to detect 19 different object categories in the hospital environment.

Despite the achievements, two knowledge gaps remain to be closed. First, the performance of deep learning-based object detection is largely influenced by the quality and generalization of the training data. While there are many existing image datasets, very few of them, if any, are focused on object detection in a cluttered healthcare facility's environment. Moreover, existing studies have not explored the importance of high-touch objects for the disinfection robot. As a result, no datasets were generated with the aim of disinfection, thus ignoring high-touch objects in the data. To overcome this limitation, this study introduces a new image dataset for indoor object detection in healthcare facilities with a total of 57 object categories. Second, the deep learning methods developed in most studies have complex architecture and cannot achieve satisfactory accuracy in real time. In this study, a new lightweight deep learning network is designed by integrating spatial- and channel- attention mechanisms into the YOLOv5 architecture to ensure accuracy and computational efficiency

Related studies on 3D object mapping

Creating a 3D object map requires the integration of simultaneous localization and mapping (SLAM) and object detection. SLAM can track camera movement and provide a camera pose, and the object detection method can detect and classify objects. The two components are essential for accurate object mapping. Object information has been widely integrated into SLAM to improve localization and mapping accuracy. For example, various studies have developed methods to improve SLAM accuracy in a dynamic environment with moving objects such as a human. Zhong et al. [81] integrated SLAM with a deep learning-based object detector to improve robot performance in a dynamic environment. Specifically, features from moving objects recognized by the deep learning network were treated as unreliable features. Zhong et al.’s SLAM method was found to be significantly improved and robust to dynamic objects. Bescos et al. [82] proposed a DynaSLAM based on ORB-SLAM2 with the integration of dynamic object detection and background inpainting. In Bescos et al.’s method, Mask R-CNN is used to detect and segment dynamic objects (e.g., person, dog, and cat) in RGB images. The segmented objects were not used for tracking and mapping, and segmented areas were inpainted with static information from previous views. Object detection has also been integrated into SLAM for identifying landmarks. For instance, Nicholson et al. [83] proposed an object-oriented SLAM method leveraging object detections as landmarks. The object detection results from multiple camera views were combined to estimate the 3D quadric surface for each object in a 3D landmark representation. A geometric error model was developed to constrain quadric parameters using 2D object detections. However, the aim of this research was mainly to estimate the camera pose instead of localizing objects on the map. For robot action, it is critical to localize objects and project them onto a 3D map so the disinfection robot can plan its action accordingly.

Several studies have combined SLAM and object detection to estimate the 3D coordinates of recognized objects. For example, Liu et al. [84] proposed a method to estimate the target position in a 3D map using a combination of ORB-SLAM2 and object detection. The authors elected to adapt the YOLOv4 network for target recognition. ArUco markers were used to provide positions of objects and enable the extraction of a more accurate object mask. The object position was obtained on an ORB-SLAM2-generated sparse map using the camera transformation. However, this method requires ArUco markers for camera calibration, object size estimation, and pose estimation, limiting its application in a new indoor environment. Furthermore, in [84], the object position was calculated from a single camera view. As a result, which camera position should be used for target position estimation and how the camera position affects the position estimation were not addressed. Another direction of 3D object mapping research is to detect objects on a 3D reconstructed map. For example, in research conducted by Rosinol et al. [85], the reconstructed 3D object mesh was fitted to known object shapes, thus, the object category could then be determined. The 3D pose of objects was also estimated by extracting an axis-aligned bounding box. However, a well-reconstructed 3D map with accurate 3D reconstructed object shapes is a prerequisite of an object matching algorithm, which is hard to obtain in a real-world application. This research aims to address this knowledge gap.

Methodology

Figure 3.1 shows a flowchart of the proposed method, which can be divided into four steps. First, a deep learning-based network is developed to detect and classify objects in RGB images. To train the network, a newly labeled dataset was used totaling 67,430 object instances covering 57 object categories in healthcare facilities. Second, SLAM was used to estimate the camera pose and generate a 3D map. Third, the camera poses, generated point clouds, and recognized objects are combined to estimate the 3D coordinates of objects. Finally, objects detected from multiple camera views are clustered using Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm. The centroid for each cluster is used to represent the position of the object.

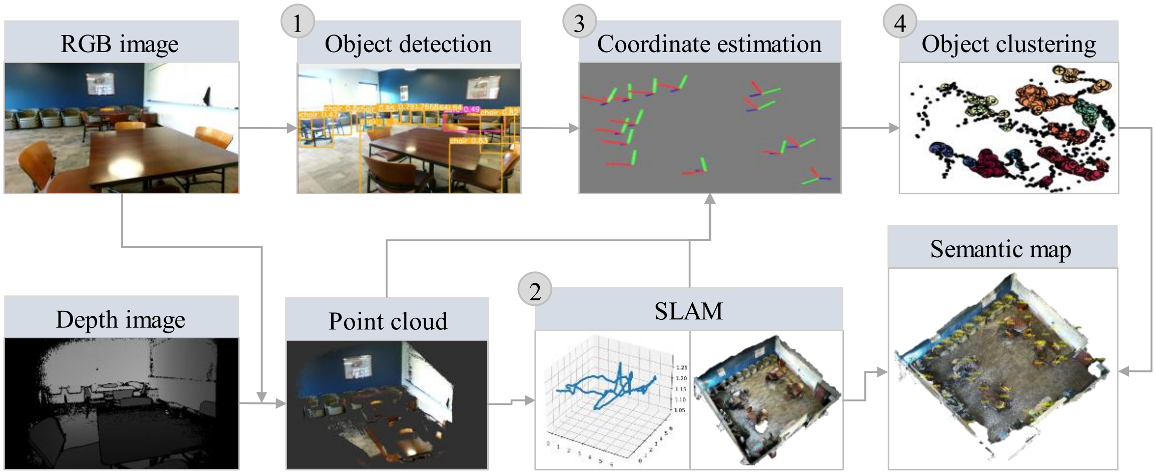


Figure 3.1. Methodology overview

The objects are projected into 3D to create a semantic 3D map. The 3D semantic map can then be exploited for robot navigation and information registration.

Dataset preparation

Data collection

In this study, image data are collected from two sources: online image search engines (Google Images, Bing Images, Getty, Shutterstock) and extracting individual frames from video data. The images and videos were obtained using a list of keywords related to healthcare facilities, such as “intensive care unit,” “operating room,” “hospital consulting room,” “hospital tour,” “hospital waiting room,” and more. A total of 1,782 images were collected from online image search engines, and 63 videos were collected from online websites (e.g., YouTube). The video data were first converted to individual frames. One frame was then extracted from at least 30 consecutive frames to achieve a more visually heterogeneous dataset. Note that images that are blurry or don’t contain objects are removed from the dataset. In total, 3,818 images were extracted from collected videos.

Object ground-truth label annotation was done by crowdsourcing the task to human labelers on the Scale AI platform. First, Scale AI labelers were given detailed instructions for every type of object with a definition and example annotations. Scale AI platform provides automated benchmarks to measure and maintain labeler quality, and thus ensure the annotation quality and reliability. The annotation from Scale AI served as our draft version of the dataset, requiring further auditing.

For the auditing task, six graduate students were recruited and comprehensively trained as data inspectors and examiners from the University of Tennessee, Knoxville. Specifically, four students were recruited as inspectors and two as examiners. The inspectors were asked to audit Scale AI annotations and make corrections. The average auditing speed is around 40 images per hour according to the report from the four inspectors. After the auditing process, the annotation quality has been significantly improved. Finally, the examiners conducted a verification process of the dataset to ensure the quality of the hospital object dataset.

Data description

The annotated dataset is named “healthcare facilities indoor object detection” (HFIOD). The HFIOD dataset consists of a total of 5600 images with 67,430 object instances. Figure 3.2 shows the statistics of the HFIOD dataset, which shows a long-tailed distribution. The handle object has the highest number of instances with a total of 7,662, followed by the chair object with 5,962 instances. Wheelchair and elevator panels have only 24 and 32 instances, respectively. The long-tail distribution poses a significant challenge for object detection.

Object detection network

This section elaborates on the network for indoor object detection in healthcare facilities. YOLO architecture is adopted in our study, which is a fast real-time multi-object detection algorithm [86]. Object detection in YOLO is done as a regression problem to estimate bounding box coordinates and class probabilities. CNN is employed to detect objects with a single forward propagation through the network, which can be trained in an end-to-end manner.

The proposed deep learning method is adapted from the YOLOv5 network. YOLOv5 is the latest upgrade from YOLOv3 with significant modifications such as adding mosaic augmentation, and customizing backbone network with Cross Stage Partial Network (CSPNet) and Spatial Pyramid Pooling – Fast (SPPF) [87]. YOLOv5 architecture is divided into YOLOv5s (small), 5m (medium), 5l (large), and 5x (extra-large) based on the number of learnable parameters in the network. The number of learnable parameters is controlled by two parameters that are depth multiple and width multiple. YOLOv5s is the smallest model among the four variants with a depth multiple of 0.5 and a width multiple of 0.33. Typically, the predictive power of the family YOLOv5 models improves with the increasing size of the network.

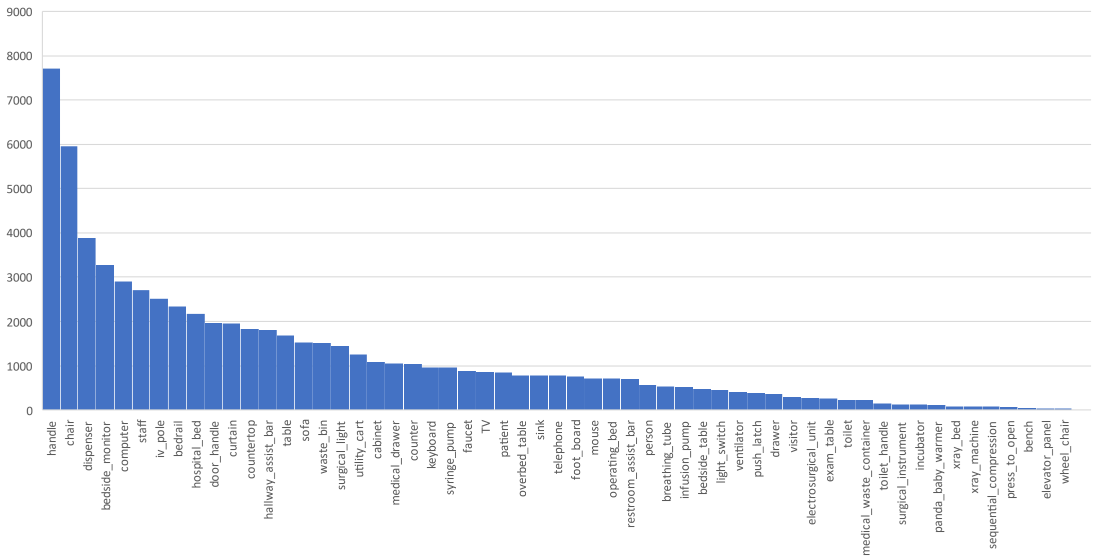


Figure 3.2. HFIOD dataset statistics

In this study, YOLOv5s architecture is selected to ensure the inference speed of the network. The YOLOv5s consists of three components that are backbone network, detection neck, as well as three detection heads. The architecture is detailed as follows.

The input images are first preprocessed using the mosaic method, which is a data augmentation method to improve network performance on small objects. The backbone network is used to extract features at various levels from images, which is built based on CSPNet [88]. The CSPNet integrates the gradient changes into the feature map from beginning to end. As such, the CSPNet could reduce the computation cost while maintaining the inference power of the network. Each CSPNet network consists of three convolutional layers cascaded by various bottlenecks. SPPF is included as the last layer backbone, aiming to extract fine and coarse information by simultaneously pooling on multiple kernel sizes (5, 9, 13). The detection neck is built based on Path Aggregation Network (PANet) [89] to boost information flow at different levels. PANet is an improvement of Feature Pyramid Network (FPN) with an additional bottom-up pathway. The detection neck aims to get feature pyramids. The feature pyramid is used to identify objects of various sizes and scales. The detection neck consists of four CSPNet blocks. The three feature maps with different scales are used to predict targets of various sizes. Finally, these feature maps are divided into grids, and each grid consists of three anchors to predict the bounding box for the object. Two improvements were introduced that are adding an attention mechanism and replacing bounding box regression loss.

Adding attention mechanism. Attention mechanism was developed by studying humans’ cognitive process in visual perception. Specifically, humans selectively focus on particular regions in the scene while ignoring some other regions (always known as backgrounds). For example, humans will learn to concentrate on useful objects that appear in the scene for an image classification task. This mechanism enables humans to quickly perceive and understand the visual context. Attention mechanism has been widely used in computer vision and demonstrated to be effective [90]. For CNN, every channel of feature map could be representative of different objects [91]. Based on this characteristic, channel attention mechanism was proposed to capture channel-wise relationships and improves the representation ability of the network. Squeeze-and-Excitation Network (SENet) [92] is the pioneering work for channel attention modeling by recalibrating weight for feature map channels. The main drawback of SENet is the ignorance of the positional information. The coordinate attention [93] was developed to address this limitation by embedding positional information into channel attention. In this study, coordinate attention mechanism is added to the detection neck as shown in Figure 3.3. This attention module is lightweight and enables the YOLOv5s network to focus on important regions at the expense of little computational cost.

Figure 3.4 shows the schematic flowchart of coordinate attention module. The attention mechanism consists of two steps. First, two spatial extents of pooling kernels are used to encode each channel of the feature map along with the horizontal and vertical directions, respectively. The output is a pair of direction-aware feature maps. Eq. (1) and Eq. (2) define the two pooling operations, where **X** is the input feature map, and $\mathrm{GAP}^{\mathrm{h}}$ and $\mathrm{GAP}^{\mathrm{w}}$ represent vertical and horizontal directions, respectively.

$$\mathbf{z}^{h} = \mathrm{GAP}^{h}(\mathbf{X}) \quad (1)$$

$$\mathbf{z}^{w} = \mathrm{GAP}^{w}(\mathbf{X}) \quad (2)$$

In the second step, direction-aware feature maps are first concatenated followed by a 1 x 1 convolutional operation. The output from the convolutional operation is split into two separate tensors along the spatial dimension. Then, two convolutional operations with kernel size 1 x 1 are applied to the two tensors, respectively. This process can be written in Eqs. (3) – (7), where δ is a non-linear activation operation, σ is the sigmoid function, F_1 represents 1 x 1 convolutional operation, F_h and F_w represent convolutional transformations on $\mathbf{f}^{h}$ and $\mathbf{f}^{w}$, respectively.

$$\mathbf{f} = \delta\Big(F_1([\mathbf{z}^{h}, \mathbf{z}^{w}])\Big) \quad (3)$$

$$\mathbf{f}^{h}, \mathbf{f}^{w} = \mathrm{split}(\mathbf{f}) \quad (4)$$

$$\mathbf{g}^{h} = \sigma\Big(F_h(\mathbf{f}^{h})\Big) \quad (5)$$

$$\mathbf{g}^{w} = \sigma\big(F_w(\mathbf{f}^{w})\big) \quad (6)$$

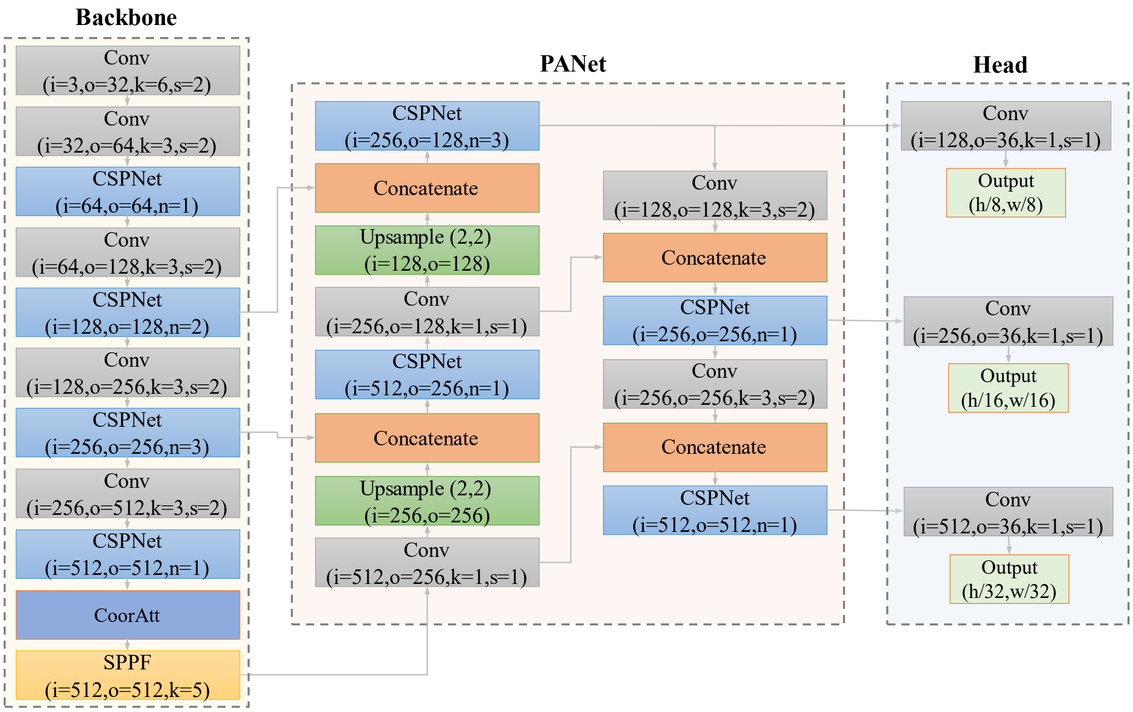


Figure 3.3. Architecture of the proposed network

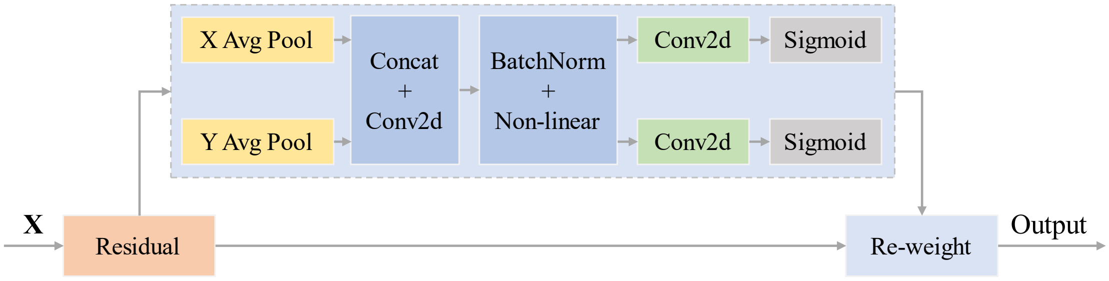


Figure 3.4. Flowchart of coordinate attention mechanism

$$\mathbf{Y} = \mathbf{X}\mathbf{g}^{h}\mathbf{g}^{w} \tag{7}$$

Replacing bounding box regression loss. The default bounding box regression loss function used to train YOLOv5 is Complete-IoU (CIoU). CIoU was developed based on Distance-IoU (DIoU) by imposing the consistency of aspect ratio. In this study, CIoU is replaced with alpha-IoU loss [94] to train the network. The alpha-IoU is a family of power IoU losses designed for bounding box regression, which has been demonstrated to be effective in small datasets and noisy bounding boxes.

The alpha-IoU is defined in Eq. (8), where $\boldsymbol{b}$ and $\boldsymbol{b}^{\mathrm{gt}}$ denote the central points of predicted bounding box B and ground-truth bounding box B^{gt}, ρ is the Euclidean distance, c is the diagonal length of the smallest enclosing box, β is a positive trade-off parameter, v is used to measure the consistency of aspect ratio, and α is the modulating parameters. When α is equal to 1, $L_{\alpha\text{-CIoU}}$ becomes CIoU loss function. $L_{\alpha-\mathrm{CIoU}}$ has a more emphasis on high-IoU objects and learns faster on these objects when α > 1. In this study, α is set to 3 to increase the loss and gradient on high-IoU objects for accurate object localization.

$$L_{\alpha-\mathrm{CIoU}} = 1 - \mathrm{IoU}^{\alpha} + \frac{\rho^{2\alpha}(\boldsymbol{b},\boldsymbol{b}^{gt})}{c^{2\alpha}} + (\beta v)^{\alpha} \tag{8}$$

3D object mapping

After recognizing objects in 2D RGB images, it is necessary to project the labels to a 3D grid map for robot navigation and disinfection. The classical pinhole camera model [95] is used to calculate the point cloud of the environment using the depth images. An example of point cloud obtained from a depth image is shown in Figure 3.5. The 3D object mapping consists of three steps, namely object coordinate calculation, SLAM, and object clustering.

Object coordinate calculation

To calculate the coordinates of an object detected in an image, the 3D point cloud corresponding to this object needs to be first identified. The depth image is aligned with the RGB image for common RGB-D cameras such as Kinect and RealSense. The object is detected as a bounding box in an image.

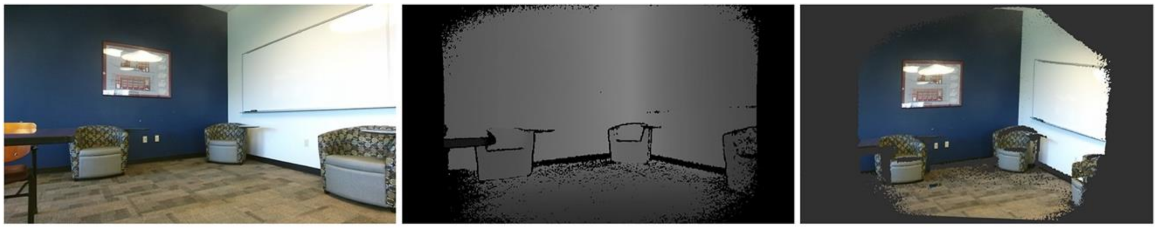

RGB image Depth image Point cloud

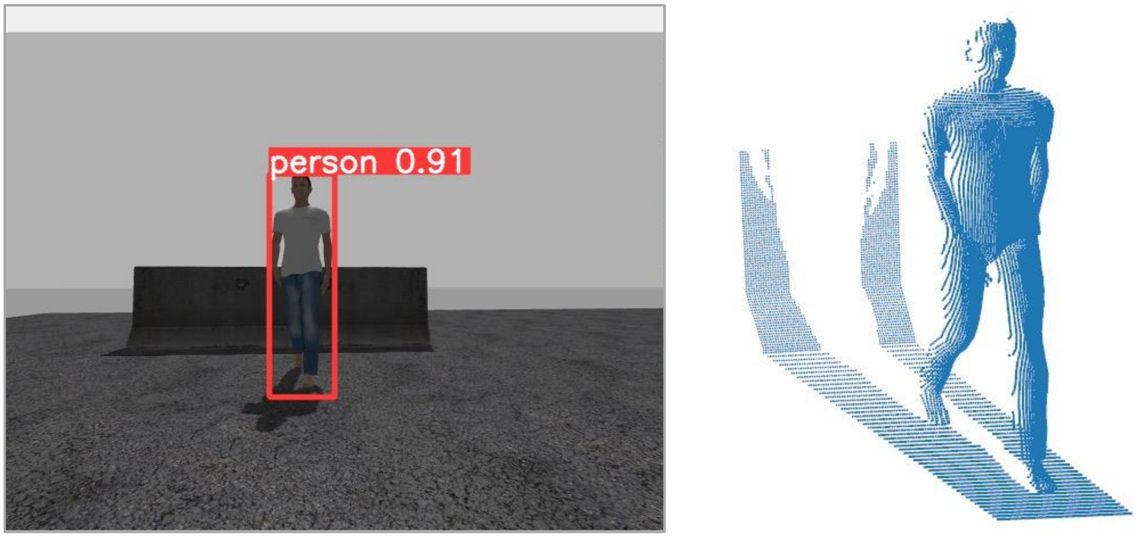
Figure 3.5. An example of point cloud generation

The 2D pixel coordinates within the bounding box are projected to a 3D point cloud. The object label is converted to cluster point indices with an assumption of 0 as the background. A set of point indices are generated and each point index represents an object label.

The output of the decomposition operator is a subset of point clouds with an object label for each detected object. In that the object is labeled as a bounding box in images, it is inevitable that some points that do not belong to this object could be included in the subset of the point cloud. These noisy points may come from the ground surface and other backgrounds, which have an impact on the accuracy of the coordinate estimation. Figure 3.6 shows an example of a clustered point cloud for a human with noisy points. As indicated, the point cloud contains noisy points from the ground plane and the barrier behind the human. The noisy points must be removed for accurate object position estimation. In this study, two point cloud filters are applied to the point cloud to remove noisy points, thus alleviating their effects on coordinate estimation. Specifically, the PassThrough filter is first applied to directly filter out points that do not meet the threshold.

The ground surface points are filtered out using a range threshold in the *z*-direction dimension. Next, a statistical outlier removal approach is used to remove points that are further away from their neighbors compared to the average for the point cloud. The method can be divided into the following steps.

- Set k, an integer, representing the number of closest points around point P_i,
- Set a standard deviation multiplier α,
- For every point P_i in the 3D point cloud
  - Find the location of k nearest neighbors to point P_i,
  - Compute the average distance d_i from point P_i to its k nearest neighbors,
- Compute the mean μ_d of the distance d_i,
- Compute the standard deviation σ_d of the distance d_i,



(a) (b)

Figure 3.6. Example point cloud corresponding to the detected person. (a) human detected by the network; and (b) point cloud

- Compute the threshold $T = \mu_d + \alpha{\cdot}\sigma_d$,
- Eliminate points in the cloud for which the average distance to its k neighbors is at a distance $d > T$.

Figure 3.7 shows the filtered point cloud using range threshold and statistical outlier removal. The ground surface points can be eliminated with the range threshold. The points that come from the background are successfully eliminated, resulting in a clean point cloud for the human. The filtered point cloud is then used to estimate the 3D coordinate of the object.

After filtering, the centroid of point cloud P_c can be estimated using Eq. (9), where N is the total number of points in the filtered point cloud, and (x_i, y_i, z_i) are point coordinates.

$$P_c = \frac{1}{N}\left(\sum_{i=0}^{N} x_i\,, \sum_{i=0}^{N} y_i\,, \sum_{i=0}^{N} z_i\right) \tag{9}$$

The 3D bounding box of the point cloud can also be estimated from the filtered point cloud. Note that the z axis of a 3D bounding box is perpendicular to the ground plane. The direction of x and y for the 3D bounding box is estimated using principal component analysis (PCA). For PCA estimation, the 3D point cloud is projected onto the ground plane. The covariance matrix is first calculated in Eq. (10), where M ($\mathbf{x}_p$, $\mathbf{y}_p$) are coordinates of projected points on the plane.

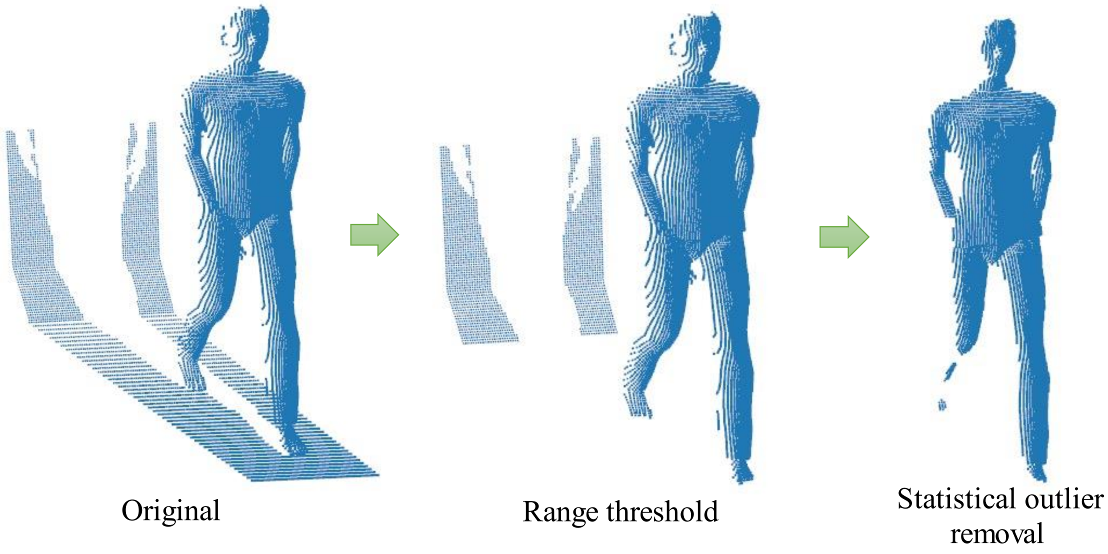
$$A = \begin{bmatrix} cov(\mathbf{x}_p, \mathbf{x}_p) & cov(\mathbf{x}_p, \mathbf{y}_p) \\ cov(\mathbf{x}_p, \mathbf{y}_p) & cov(\mathbf{y}_p, \mathbf{y}_p) \end{bmatrix} \tag{10}$$

Given an eigenvalue of λ, an eigenvector V associated with λ for the covariance matrix, A should satisfy Eq. (11). The eigenvector V for A can then be calculated. V_1 and V_2 are directions of the first and second principal components, i.e., directions of x and y, respectively.

$$AV = \lambda V \tag{11}$$

Next, the point cloud M is projected onto principal components using Eq. (12)

$$T = VM \tag{12}$$



Original

Range threshold

Statistical outlier removal

Figure 3.7. Filtered point cloud

Localization and mapping

The RTAB-Map SLAM method [96], a graph-based SLAM technique, is used in this study to locate the robot and produce the occupancy map for navigation, see Figure 3.8. The structure of the map consists of nodes and links. Odometry nodes publish odometry information to estimate robot poses. Visual odometry obtained from ORB-SLAM2 [97] is used as odometry input in this study, because ORB-SLAM2 is fast and accurate. The short-time memory (STM) module is used to create nodes to memorize the odometry and rgb-d images, as well as calculate other information such as visual features and the local occupancy grid. In order to limit the WM size and thus decrease the time to update the graph, a weighting mechanism is used to determine which nodes in working memory (WM) are transferred to long-term memory (LTM). Nodes in the LTM can be brought back to WM when a loop closure is detected. Links are used to store transformation information between two nodes. The neighbor and loop closure links are used as constraints for graph optimization and odometry drift reduction**.** The Bag of Words approach [98] is used for loop closure detection. The visual features extracted from local feature descriptors such as ORB [99] are quantized to a vocabulary for fast compassion. The outputs from RTAB-Map include camera pose and 2D occupancy grid, which are further used for semantic mapping and robot navigation. The rtabmap-ros package is available in ROS, which enables seamless integration with autonomous robots for this application. Because the settings of built environments do not change very frequently, maps of the built environments can be first produced and then used to locate and navigate the robots during the cleaning and disinfection process to improve efficiency and reduce memory use.

Object clustering

Since the camera is constantly moving, an object could be detected from different camera views. Hence, it is necessary to cluster object detection results from different perspectives. 3D coordinates for detected objects are first estimated at different camera views.

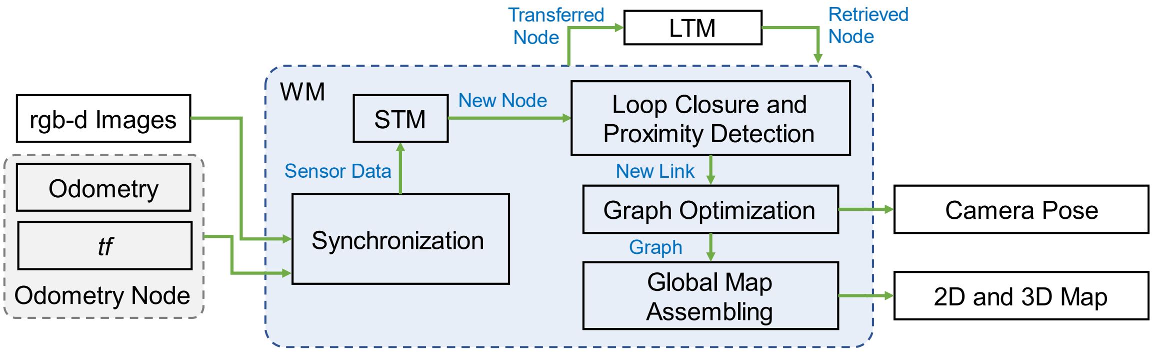


Figure 3.8. RTAB-Map SLAM framework (adapted from [96])

The density-based spatial clustering of applications with noise (DBSCAN) clustering algorithm [100] is used to group points that are closely packed together for each category.

DBSCAN requires two parameters, which are maximum distance eps (ε) and a minimum number of points (minPts) for a cluster. The algorithm starts by selecting an arbitrary point in the point cloud, and points within ε are retrieved. If the number of points is greater than minPts, a cluster is formed. Otherwise, the point is treated as noise. If a point is identified as a part of a cluster, its neighbor points within the distance ε are also added to this cluster. The cluster continues to add new points until no points within ε of the cluster. Next, a new unvisited point is selected and proceeded with the same procedure to identify clusters or noise. The process of DBSCAN can be expressed in pseudocode in Figure 3.9.

Experiment and results

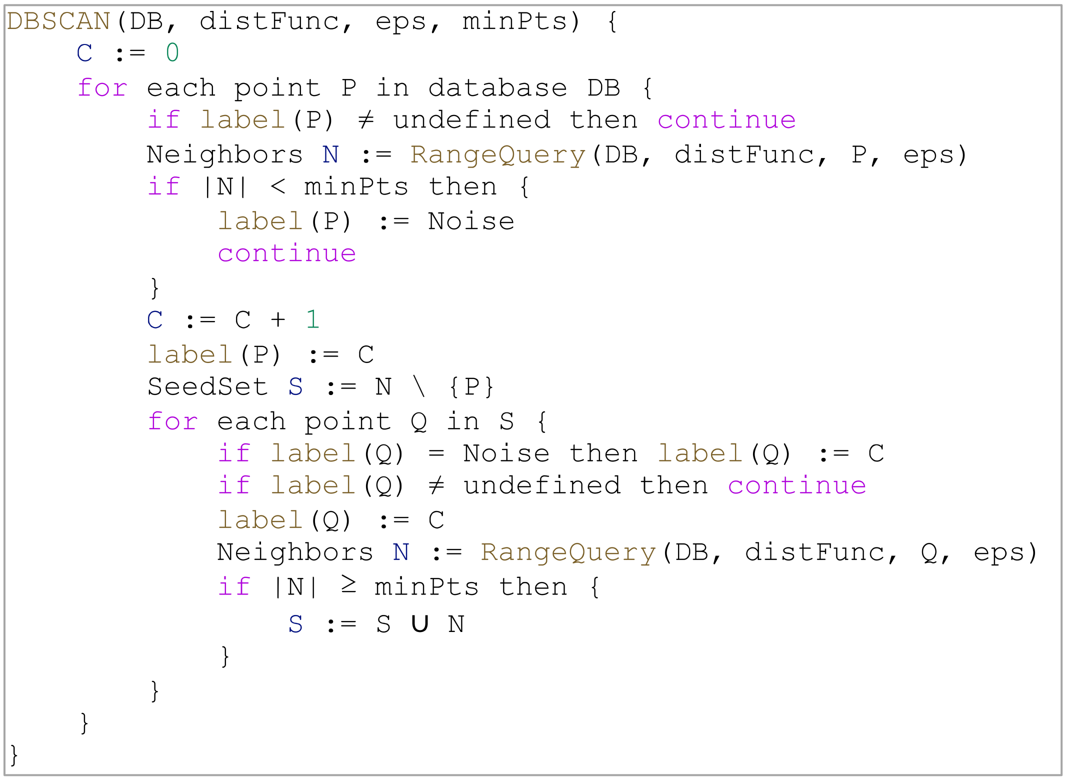
Results on object detection

Implementation details

The network is trained on a workstation running Ubuntu 16.04 with dual Intel Xeon Gold 4114 CPU, 128 GB RAM, and NVIDIA RTX A6000. The Stochastic Gradient Descent (SGD) optimizer is used to train the network. The network is trained for a total of 300 epochs. The EFSBD dataset is randomly split into a training set (80%), and a validation set (20%). The images are resized to 640 × 640. The confidence and intersection over union (IoU) thresholds for non-maximum suppression (NMS) operation are set to 0.1 and 0.4, respectively. The early stopping technique is used to avoid the overfitting problem. Specifically, the network stops training when the loss value does not decrease for 20 epochs. The model with the highest performance on the validation set is saved for performance analysis. The hyperparameters are given in Table 3.1.

Metrics

The average precision at IoU threshold is 0.5 (AP50), and the mean average precision (mAP) over various IoU thresholds are used to quantify the network performance.



```
DBSCAN(DB, distFunc, eps, minPts) {
    C := 0
    for each point P in database DB {
        if label(P) ≠ undefined then continue
        Neighbors N := RangeQuery(DB, distFunc, P, eps)
        if |N| < minPts then {
            label(P) := Noise
            continue
        }
        C := C + 1
        label(P) := C
        SeedSet S := N \ {P}
        for each point Q in S {
            if label(Q) = Noise then label(Q) := C
            if label(Q) ≠ undefined then continue
            label(Q) := C
            Neighbors N := RangeQuery(DB, distFunc, Q, eps)
            if |N| ≥ minPts then {
                S := S ∪ N
            }
        }
    }
}
```

Figure 3.9. Pseudocode of DBSCAN algorithm (adapted from [100])

Table 3.1. Hyperparameters for model training

Parameter	Value	Parameter	Value
Initial learning rate	0.01	IoU training threshold	0.2
Learning rate factor	0.01	Anchor-multiple threshold	4
Momentum	0.937	HSV-hue augmentation	0.015
Weight decay	0.005	HSV-saturation augmentation	0.4
Warmup epochs	3	HSV-value augmentation	0.4
Warmup momentum	0.8	Rotation	0.2
Warmup learning rate	0.1	Translation	0.1
Box loss gain	0.05	Scale	0.5
Classification loss gain	0.5	Flip up-down	0.2
Classification BCELoss positive weight	1	Flip left-right	0.5
Object loss gain	1	Mosaic	1
Object BCELoss positive weight	1	Segment copy-paste	0.2

Average precision (AP) is the area under the precision-recall curve that is defined in Eq. (13). The average of AP for all the classes is defined in Eq. (14) and expressed as AP to be differentiated from mAP, where *nc* represents the number of classes. As different IoU thresholds can produce different predictions, mAP was used to overcome this problem by averaging AP scores on different IoU thresholds. In this study, mAP is calculated as an average of AP over 10 IoUs starting from 0.5 to 0.95 with a step size of 0.05, which has been used as a standard metric to evaluate object detection methods. Therefore, mAP is used as the metric to evaluate the overall performance of the model.

$$\mathrm{AP}' = \int_0^1 \mathrm{precision}(\mathrm{recall})d(\mathrm{recall}) \tag{13}$$

$$\mathrm{mAP} = \frac{1}{nc}\sum_{i=1}^{nc} \mathrm{AP}_i' \tag{14}$$

Network performance

Figure 3.10 shows loss variation on the training and validation dataset, as well as AP_{50} and mAP variation on the validation set during the training stage. On the training dataset, the box, classification, and alpha loss keep decreasing over 300 epochs. The object loss increases at the beginning and then keeps decreasing after around 50 epochs. On the validation set, the classification and alpha loss decreases and becomes stale around 200 epochs. AP_{50} and mAP metrics keep increasing during the training stage. The proposed method achieved an AP_{50} of 67.6% and an mAP of 46.7% on the validation set of the HFIOD dataset.

Figure 3.11 shows the precision and recall variation over confidence and precision-recall curve. Precision is used to measure the correctly classified positive against the total number of classified positives. Recall measures the predictive power of the network in identifying all the positive elements. The precision score increases with increasing confidence while recall score decreases. The results also indicate a large variation of precision and recall scores across different categories. The precision-recall curve indicates the tradeoff between precision and recall under the different threshold.

Figure 3.12 presents the confusion matrix for the proposed method on the validation set.

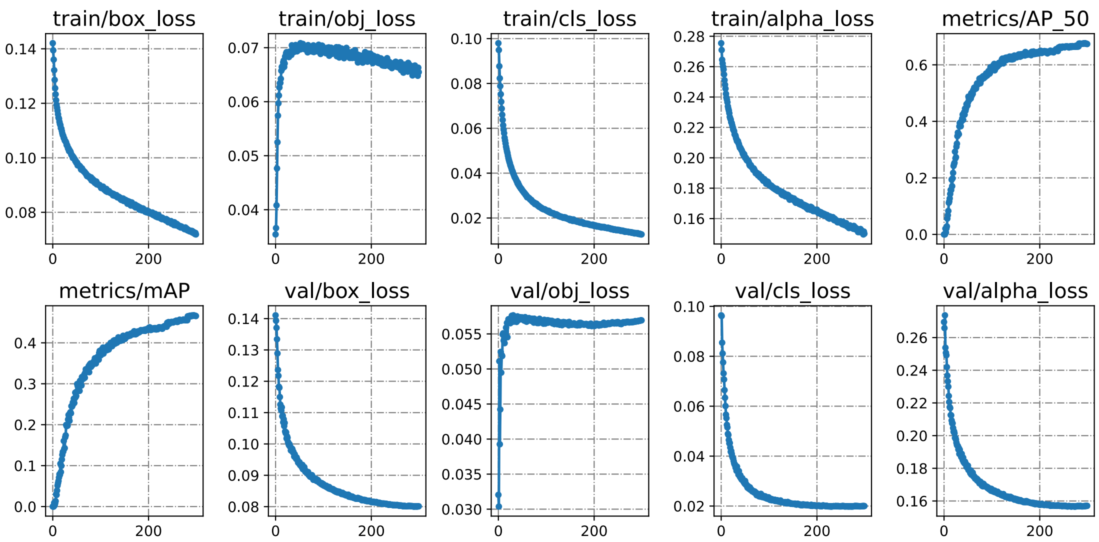


Figure 3.10. Metric variation over epochs

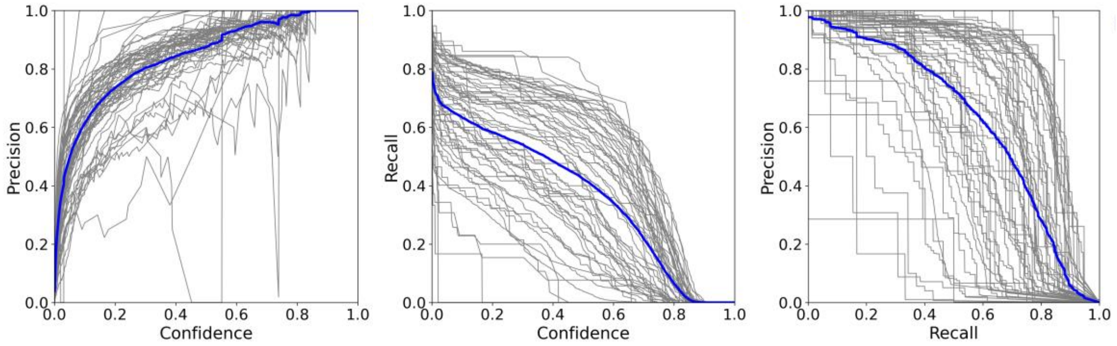


Figure 3.11. Precision and recall variation under different confidence thresholds and precision-recall curves

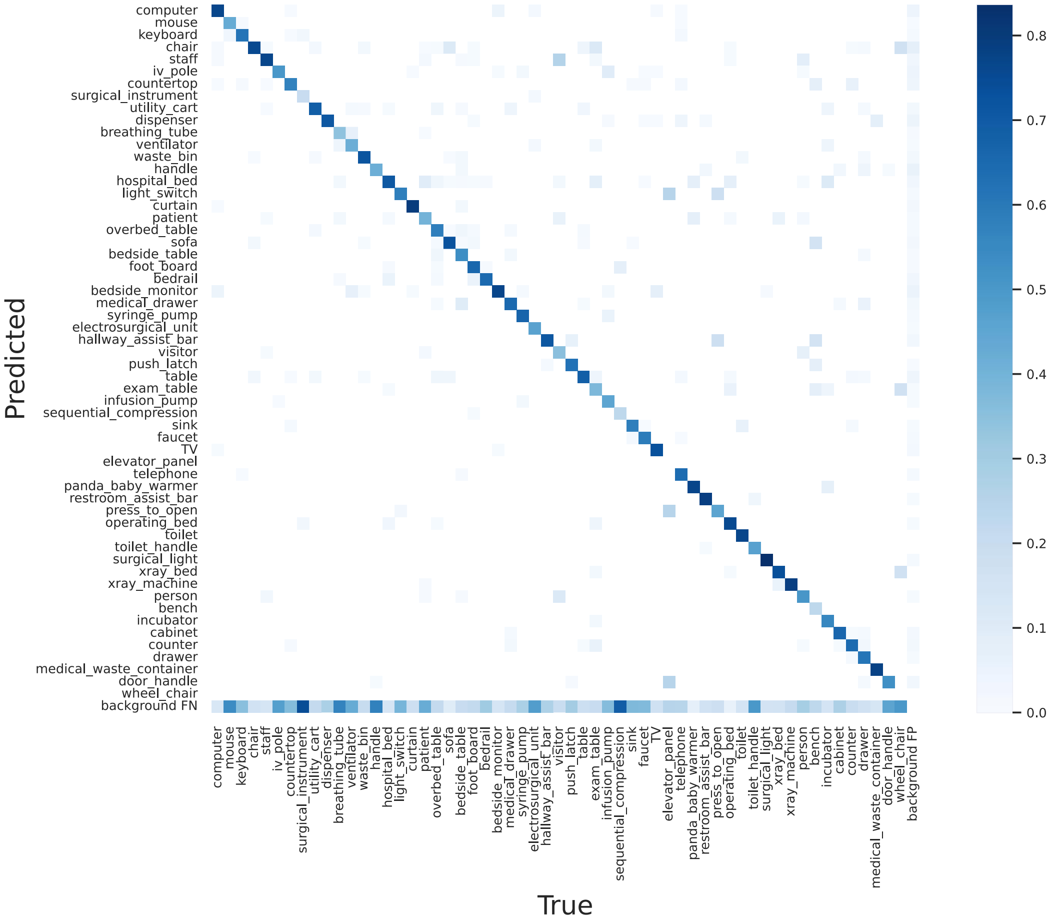


Figure 3.12. Confusion matrix for the proposed method on the validation set of the HFIOD dataset

The matrix is normalized by the column so that diagonal values represent recall for each category. The surgical light category achieves the highest recall score with a recall of 83.6%. The curtain category achieves the second-highest recall score of 79.6%. A high recall score indicates most positive samples for this category can be accurately detected. The elevator panel achieves the lowest recall score because of a very small sample size of this category in both the training and validation datasets. Objects are mostly misclassified as background. For example, The recall score for breathing tubes is 34.3%, and 56.9% is misclassified as background. Misclassifying objects as background could lead to missing disinfection of high-touch objects, resulting in insufficient disinfection of the room. Figure 3.13 illustrates example results of object detection in the validation set of the HFIOD dataset. The results indicate that the proposed method can accurately detect and classify objects in healthcare facilities.

Ablation study

In this section, an ablation study is conducted to assess the effectiveness of the two proposed improvements on the YOLOv5s network. The YOLOv5s is used as the baseline model. The effectiveness of alpha-IoU and coordinate attention are evaluated by individually integrating it into the baseline model. Table 3.2 presents the results of the testing set of HFIOD dataset. The mAP is used to evaluate the performance of the network. The results indicate the baseline network is improved by alpha-IoU with an improvement of 0.6%. The coordinate attention module improves the performance of the baseline by the same margin of 0.6%. A combination of alpha-IoU and coordinate attention achieves the best performance, which has an improvement of 1.7%. The improvement demonstrates the effectiveness of the proposed method in detecting and classifying building damage.

Results on object mapping

To evaluate the accuracy of object mapping, a robot simulation platform was created to replicate indoor environment reconstruction. The platform is built on a laptop running Ubuntu 18.04. The distribution of ROS is Melodic, and the version of Gazebo is 9.

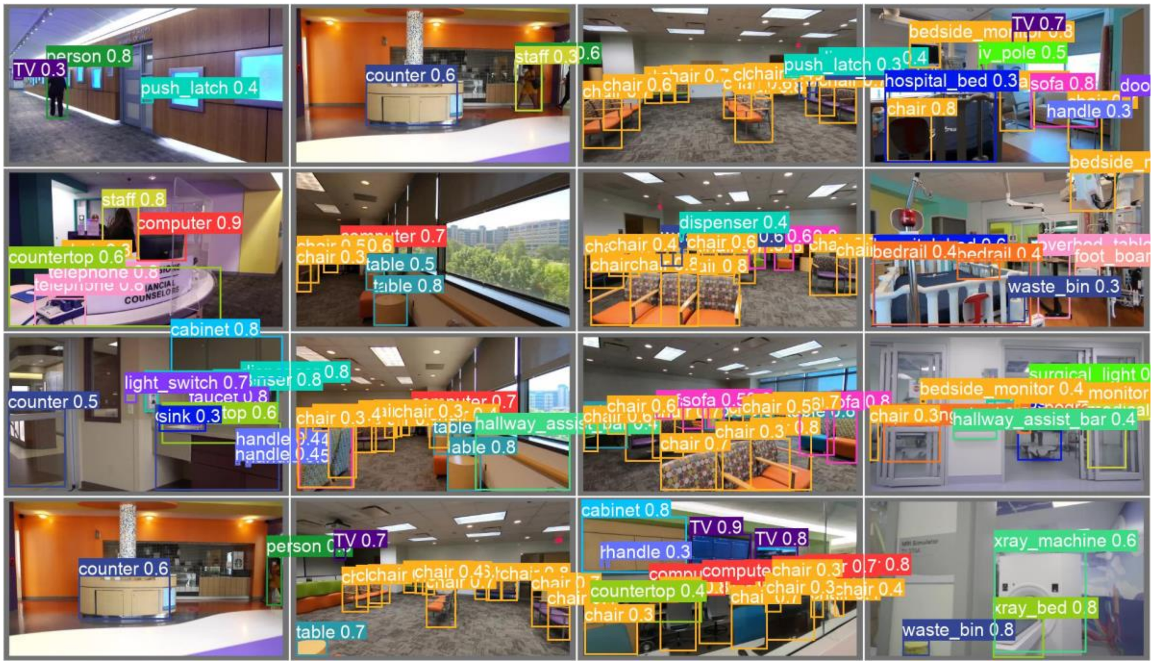


Figure 3.13. Examples of model prediction results

Table 3.2. Ablation study on the testing set of the HFIOD dataset

Model	alpha-IoU	Coordinate attention	mAP (%)
YOLOv5s	-	-	45.0
	✓	-	45.6
	-	✓	45.6
Proposed	✓	✓	**46.7**

In the simulation platform, the robot is equipped with an LMS1xx 2D laser and RealSense RGB-D camera. The onboard camera is used to capture the surrounding environment. Figure 3.14 shows the experiment setup, where six persons are randomly placed in the indoor environment. The human object is selected for evaluation purposes. This is because our deep learning network is trained using real images, compromising its performance in a simulated environment. The human feature in the virtual environment is similar to that in the real world, which can be detected constantly. The estimated positions of the six persons using the proposed method are compared to their ground-truth positions.

Figure 3.15 shows the reconstructed 3D map with object information. As indicated, the indoor environment is properly reconstructed. Six persons are correctly detected and clustered using the proposed method. Note that many objects are not able to detect because of the network's compromising performance in the simulation environment. Table 3.3 shows the performance of the object coordinate estimation. The results indicate that the estimated position is in good agreement with the ground truth. Specifically, the error in the x direction varies from 0.03 m to 0.47 m with an average of 0.24 m. The error in the y direction varies from 0.01 m to 0.37 m with an average of 0.28 m. The promising results demonstrated the potential of the proposed method in object detection and mapping in an indoor environment.

The performance of the proposed method is also evaluated on the real video data collected in a lounge room on UTK campus. The room is crowded with more than 20 chairs, multiple desks, and other furniture, which is very challenging for the proposed method. Figure 3.16 shows the reconstructed 3D map with object information. The results indicate that most chairs are successfully detected with accurate positions. The small door handle on the right side is accurately detected and localized. Note that on the left side, some chairs are also detected as sofas due to their similarities, leading to inconsistent predictions from different camera views.

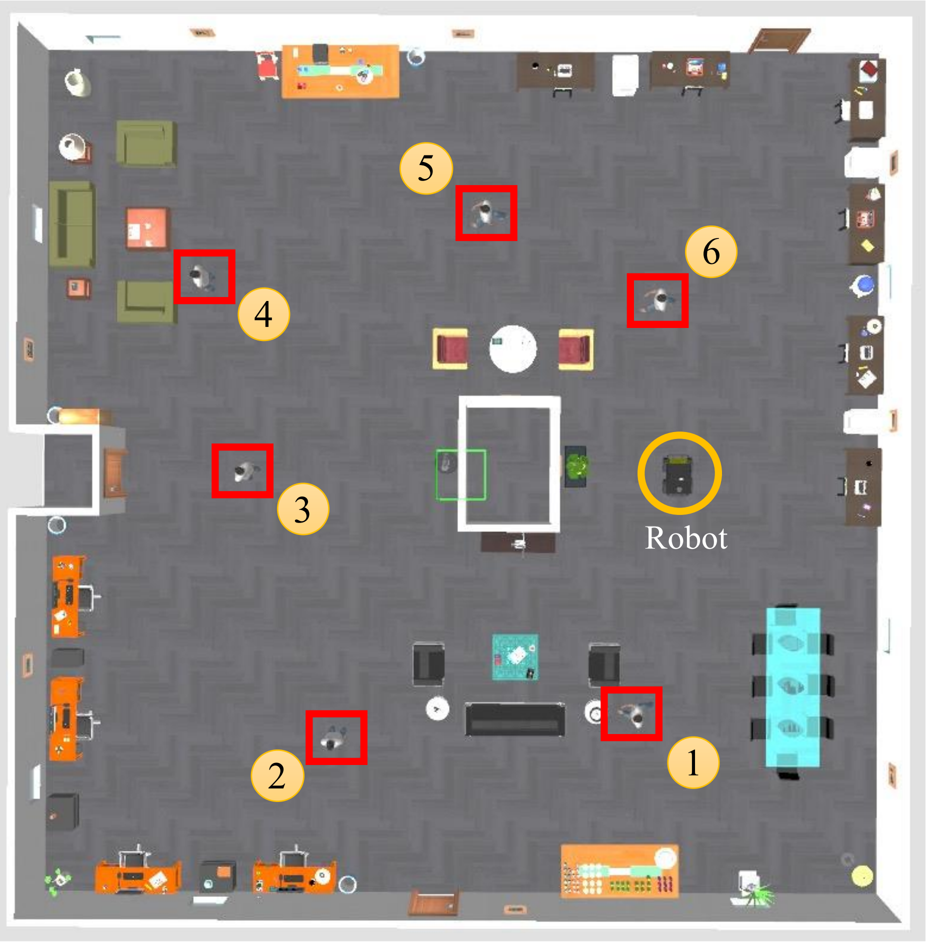


Figure 3.14. Overview of the robot simulation platform

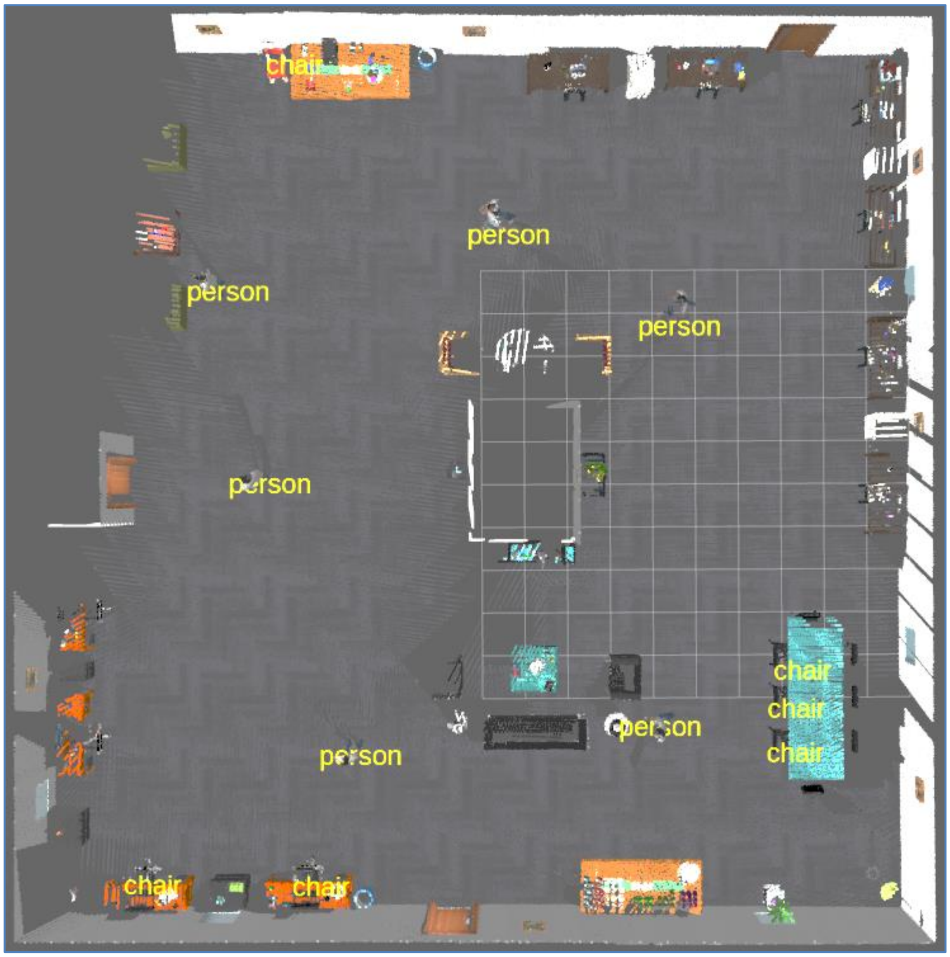


Figure 3.15. Reconstructed 3D map with object information in the simulation platform

Table 3.3. Comparison of estimated and ground-truth object positions

Object id	Estimation	Ground truth	x error (m)	y error (m)
1	(5.45, 4.02)	(5.5, 4)	0.05	0.02
2	(6.08, -2.72)	(6, -3)	0.08	0.28
3	(-0.03, -4.73)	(0, -5)	0.03	0.27
4	(-4.33, -5.63)	(-4.5, -6)	0.17	0.37
5	(-5.60, 0.66)	(-6, 0.5)	0.4	0.16
6	(-3.53, 4.49)	(-4, 4.5)	0.47	0.01

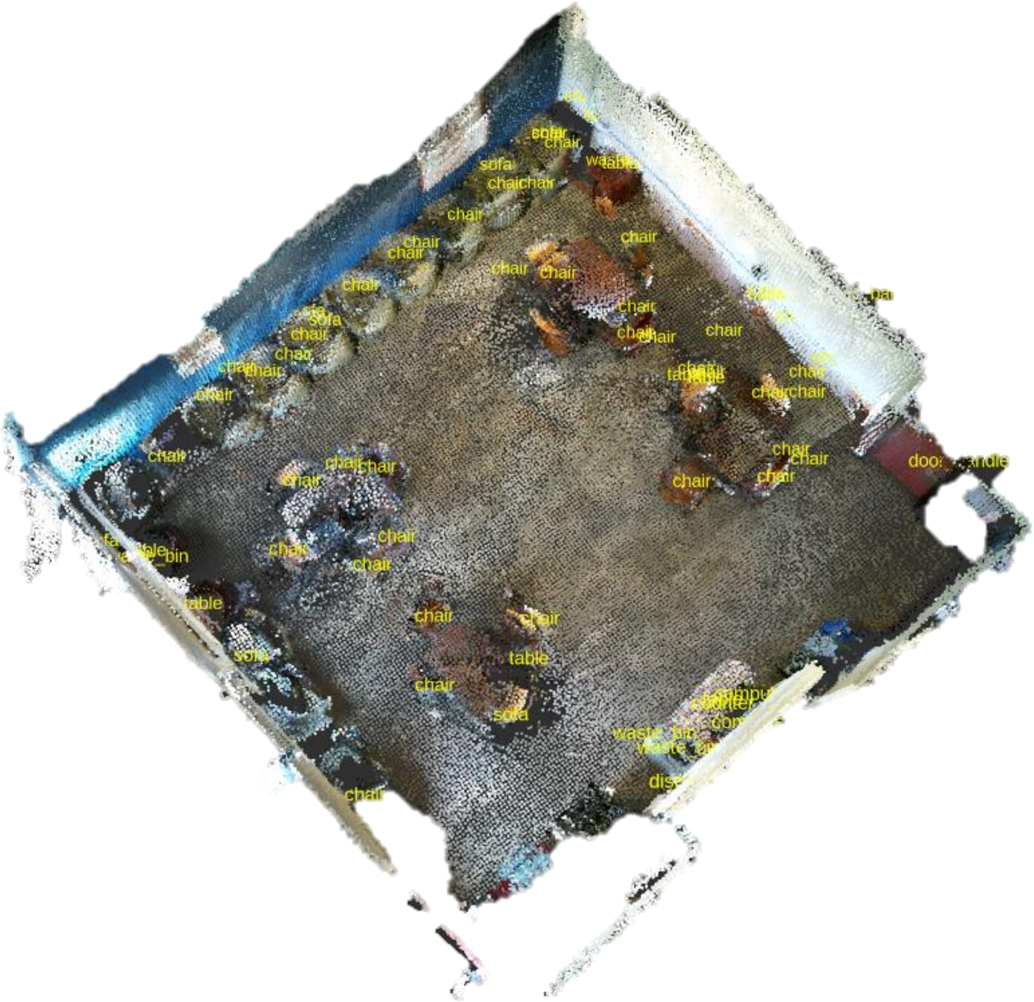


Figure 3.16. Reconstructed 3D map with object information in a building room

Discussion

Influence of image resolution

This section discusses the effect of image resolution and network size in the task of building damage detection. Table 3.4 shows the performance of the proposed method over different image resolutions. The results indicate that the mAP score increases with increasing image resolutions. In particular, the network has the largest performance increase from 416 x 416 to 640 x 640, which has an mAP improvement of 5.2%. The network performance is improved by 2.0% when increasing image resolution from 640 x 640 to 1280 x 1280. On the other hand, the inference time of the network increases with increasing image resolution. The inference speed is evaluated using NVIDIA RTX A6000 GPU. The inference speed of 416 x 416 reaches 69 FPS. When image resolution increases to 640 x 640, inference FPS decreases to 67. The detection can be regarded as real time when the inference speed is greater than 30 FPS. The selection of image resolution is a trade-off between accuracy and speed. In this study, 640 x 640 is chosen for real-time object detection with due accuracy.

Influence of network size

Table 3.5 presents a comparison of the proposed method with other networks in the family of YOLOv5 with larger network sizes. The results indicate the model performance significantly increases with the increasing size of the network from YOLOv5s to YOLOv5x. YOLOv5x has an mAP improvement of 8.6% compared to YOLOv5s. While a network with a large size tends to improve its performance, it requires more storage and computation cost which hinders its deployment to mobile platforms such as robots. The small network has the potential to be integrated into an embedded system for real-time detection.

Limitation and future research directions

Several research directions deserve future studies. First, while the proposed deep learning network achieved promising results on the HFIOD dataset, there remains significant room for improvement.

Table 3.4. Effect of image resolution on network performance

Resolution	AP_{50} (%)	mAP (%)	Time (ms)
416 x 416	61.9	41.5	14.5
640 x 640	67.6	46.7	15.0
832 x 832	68.5	47.5	18.7
1280 x 1280	70.6	48.7	20.4

Table 3.5. Comparison of deep learning networks

Model	AP_{50} (%)	mAP (%)	Parameters (million)	Time (ms)
YOLOv5x	75.5	53.6	86.6	28.8
YOLOv5l	75.0	53.2	46.4	25.5
YOLOv5m	72.4	50.0	21.1	19.7
YOLOv5s	67.7	45.0	7.2	13.9
Ours	67.6	46.7	7.2	15.0

For example, the performance of some object categories such as wheelchairs is unsatisfactory. The low accuracy for these objects stems from a small sample size in the HFIOD dataset. In a future study, more data must be collected, particularly for objects with small sample sizes. Second, 3D object mapping is challenging in a room crowded with furniture. A crowded space will lead to object occlusion, which is a classic problem in image and point cloud processing. In our work, the occlusion will affect the object coordinates estimation. Future study is needed in this direction to address this challenge.

Conclusion

This study developed a new method for indoor object detection and mapping. The practical utility of the proposed methods is sustained by two computational innovations as well as high performance validated using real-world data and scenarios. The developed method is superior to existing solutions as it could accurately detect and classify 57 categories of indoor objects in healthcare facilities in real time. This success was achieved by preparing an unprecedented dataset for robust performance as well as incorporating a new attention mechanism in the deep learning method for detection and classification. The proposed deep learning network achieved an AP50 of 67.6% and an mAP of 46.7% on the validation dataset. The proposed method is lightweight and achieved real-time detection with an FPS of 67. Therefore, the AI method can be developed in an embedded system for real-time detection. A new mechanism was developed to estimate the 3D coordinates of detected objects and project them onto 3D maps. A robot simulation platform was built to test the performance of the 3D coordinate estimation, which resulted in an average error of 0.24 m and 0.28 m in the x and y directions, respectively. The methods and workflow were also validated in a real indoor environment on campus, demonstrating their applicability in real-world applications.

CHAPTER FOUR
OBJECT AFFORDANCE SEGMENTATION

Introduction

Diseases caused by microbial pathogens have long plagued humanity, and are responsible for over 400 million years of potential life lost (a measure of premature mortality) annually across the globe [101]. Mass-gathering built environments such as hospitals, schools, and airports can become hot spots for microbial pathogen colonization, transmission, and exposure, spreading infectious diseases among people in communities, cities, nations, and worldwide. The outbreaks of infectious diseases impose huge burdens on our society. For instance, with more than 259 million people infected and 5.1 million killed [1], the pandemic of the coronavirus disease 2019 (COVID-19) continues to impose a staggering infection and death toll. In addition, the epidemic of flu costs the U.S. healthcare system an average of $11.2 billion each year [102]. During the 2019-2020 flu season, it was estimated that 24,000 to 62,000 people could die because of flu [103]. Each year, there are about 1.7 million hospital-acquired infections in the U.S., resulting in 99,000 related deaths [104]. The disastrous impacts of infections on the society and economy are enormous, highlighting the urgency for developing effective means to mitigate the spread of infectious pathogens in built environments.

Suggested by the World Health Organization (WHO) and the Centers for Disease Control and Prevention (CDC), frequent cleaning and disinfection are critical for preventing pathogen transmission and exposure to slow down the spread of infectious diseases. For instance, during the pandemic of COVID-19, 472 subway stations in New York City were disinfected overnight by workers after a second confirmed COVID-19 case in New York [105]. Deep cleanings are also conducted for school buildings during the closures [106]. Now disinfection is routine and necessary for all mass-gathering facilities, including schools, airports, transit systems, and hospitals. However, the manual process is labor-intensive, time-consuming, and health-undermining, limiting the effectiveness and efficiency of disinfection. First, pathogens can survive on a variety of surfaces for a long period of time. For example, norovirus and influenza A virus were found on objects with frequent human contacts in elementary school classrooms [107]. The coronavirus that causes severe acute respiratory syndrome (SARS) can persist on nonporous surfaces such as plastics for up to 9 days [108]. Second, pathogens spread very quickly within built environments. It was found that contamination of a single doorknob or tabletop can further contaminate commonly touched objects and infect 40-60% of people in the facilities [109]. Hence, the cleaning and disinfection workers are burdened by heavy workloads and subject to high infection risks. Third, workers could be harmed by the chemicals and devices used for disinfection. For instance, nurses who regularly clean surfaces with disinfectants were found to be at a higher risk of chronic obstructive pulmonary disease [110]. Exposure to disinfectants was also found to cause asthma [111]. Therefore, there is a critical need for an automated process for indoor disinfection to replace human workers from such labor-intensive and high-risk work.

To address this critical need, the objective of this study is to create and test a novel framework and new algorithms for a robotic manipulator to conduct automatic disinfection in indoor environments to reduce pathogen transmission and exposure, and thus potentially prevent outbreaks of infectious diseases. The contribution of this study lies in the development of a deep-learning method to detect and segment the areas of potential contamination. Using the visual simultaneous localization and mapping (SLAM) technique, the segmented areas of potential contamination are mapped in a three-dimensional (3D) space to guide the robotic disinfection process. The rest of the paper is organized as follows. Related studies are reviewed in Section 2 to reveal the knowledge gaps and technical barriers to be addressed in this study. Then, the framework and methods are elaborated in Section 3, followed by the experimentation and evaluation in Section 4. Section 5 concludes this study by discussing the applicability of robotic disinfection in built environments, limitations, and future research directions.

Literature review

Cleaning and disinfection robots, such as the ones with ultraviolet (UV) lights, have been used in healthcare facilities [112,113] to prevent hospital-acquired infections. Perceived as a mobile UV light, the robot configuration only allows it to disinfect the room at an aggregate level. Precision disinfection is not considered feasible with this robot. During the COVID-19 pandemic, a new robot has been deployed to use vaporized hydrogen peroxide to clean and disinfect the stations and trains in Hong Kong [114]. Many studies focused on floor-cleaning robots. For example, the hTetro floor cleaning robot was developed [115–117], which can reconfigure its morphology to maximize its productivity based on the perceived environments. A novel method was proposed in [118] to express the situations of floor-cleaning robots to users. There are very few intelligent robot systems that can perform precision disinfection for various objects in different built environments. The absence of such intelligent robots stems from two knowledge gaps. First, there lacks a method to enable the robot to perceive and map the areas of potential contamination in the built environments, hindering the precision disinfection. Second, the robot needs to adapt its trajectories with respect to different areas of potential contamination for effective and safe disinfection. However, this capability has not been achieved by existing robot systems. Therefore, this study aims to address the two knowledge gaps and technical barriers. Next, a number of studies regarding fundamental robotic techniques are also reviewed.

SLAM is a fundamental technique that enables the robots to perceive the environment, localize itself, and build a map for subsequent applications. The SLAM techniques need to be compatible with the robot operating system (ROS) to allow robot navigation in built environments. GMapping [119] is a ROS default SLAM approach that uses a particle filter to create grid maps and estimate robot poses. GMapping and TinySLAM [120] can be used for localization and autonomous navigation. Using 2D light detection and ranging (LiDAR) with low computation resources, Hector SLAM [121] and ethzasl_icp_mapping [122] can provide real-time 2D occupancy mapping. Google Cartographer [123] is an efficient graph-based SLAM approach using portable laser ranger-finders. Maplab [124] and VINS-Mono [125] are graph-based SLAM approaches that fuse the information from an inertial measurement unit and a camera. The RTAB-Map [96] is a complete graph-based SLAM and has been incorporated into a ROS package for various applications. ORB-SLAM2 [97] is a popular feature-based visual SLAM approach that has been adapted to monocular, stereo, and RGB-D cameras. In contrast to feature-based algorithms, dense visual odometry DVO-SLAM [126] uses photometric and depth errors over all pixels of two consecutive RGB-D images to estimate camera motion.

Robot perception is important for deriving intelligent robot decisions and actions. In this application, robots need to perceive the areas of potential contamination on various objects for cleaning and disinfection. Detecting and segmenting the areas of potential contamination from images are related to object detection and semantic segmentation. In addition, the concept of object affordance is also relevant. Many studies have been conducted in object detection and recognition [11–15], scene classification [16,17], indoor scene understanding [18,19]. However, merely detecting the scene elements is not enough to make intelligent decisions. Particularly for this application, detecting a computer on an office desk does not mean that the computer needs to be disinfected. The existing object detection and segmentation techniques lack the capabilities to reason out which areas or under what circumstances, the object or the part of the object needs specific disinfection. Understanding how humans interact with different objects will help determine the potential areas of contamination. For example, high-touch areas are considered to be contagious and should be disinfected based on WHO and CDC suggestions. Human interactions with objects have implications on how and which part of objects may be contaminated and could contaminate different parts of human body. In computer vision, studies have been conducted to predict affordance of whole objects [20,21]. In [22], a region proposal approach was integrated with convolutional neural network (CNN) feature-based recognition method to detect affordance. In [23], a method was developed to learn affordance segmentation using weakly supervised data. In [24], a number of studies of object affordance in computer vision and robotics were reviewed.

Despite the advancements, there still remains a gap in linking semantic segmentation and object affordance with the areas of potential contamination. This study aims to address this limitation.

Methodology

Figure 4.1 presents an overview of the proposed method that enables intelligent robotic disinfection in built environments. The robot is equipped with an RGB-D camera for SLAM and perception in built environments. RTAB-Map is used to provide pose estimation and generate a 2D occupancy map, which has been described in Chapter 3. A deep learning method is developed to segment object affordance from the RGB-D images and map the segments to areas of potential contaminations in a 3D map. The high-touch areas can be automatically detected and segmented, which may be colonized by a variety of pathogens. The 3D semantic occupancy map and the locations of the areas of potential contamination are exploited for robot disinfection planning. The robot, with UV lights attached to end-effectors, will navigate to appropriate positions, and adapt its scanning trajectories to disinfect the objects. The framework and methods are detailed as follows.

3D object affordance segmentation

The areas of potential contamination need to be automatically detected and mapped in 3D space to guide robotic disinfection. Particularly, the object surfaces with frequent human contacts are the areas of potential contamination requiring disinfection. Therefore, those areas need to be automatically detected and segmented from the RGB images, and thereafter projected to a 3D semantic map for robot navigations and actions. To this end, a deep learning method is developed based on the object affordance concept [21] and the approach proposed in [127] to segment the areas of potential contamination. It is necessary to label a surface that has interactions with different parts of human body. For example, the seating surface of a chair has contact with human hip, the backrest has contact with human back, and the armrest has contact with human hand, posing different implications for distinction.

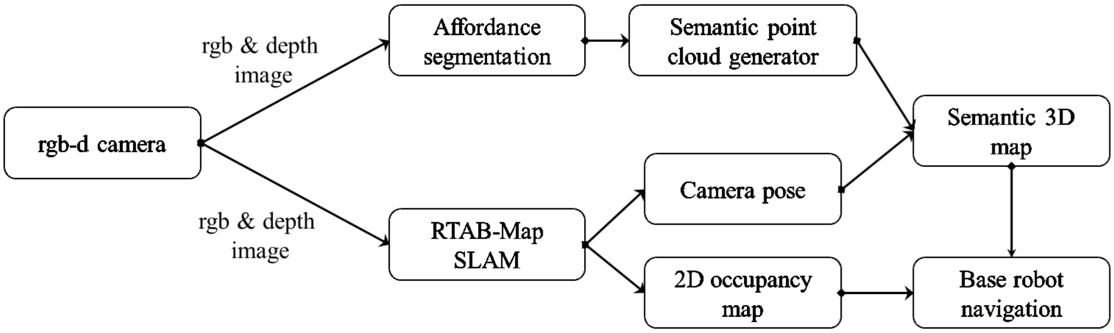


Figure 4.1. Methodology overview

Five object affordance labels are selected, including walk, grasp, pull, place, and sit, as these activities cover the most common interactions with inanimate objects in built environments. For example, the walkable areas indicate the places where the robot can move and conduct floor cleaning. The places where grasping, pulling, and placing occur represent potential high-touch areas that need to be frequently disinfected.

In this study, to train a deep learning method to segment the object surfaces as the areas of potential contamination, the ADE20K datasets [128] and simulated images [129] with appropriate labels are used. Figure 4.2 presents some sample images of the training dataset. The ADE20K training dataset only labels objects and their parts. Similar to [130], a transfer table is defined to map 116 object labels to the corresponding five object affordance labels. Table 4.1 presents several examples. Each object or its part is associated with a five-dimensional vector, representing the five object affordance labels. The value 1 indicates that a specific object affordance is associated with an object or its part, and value 0 indicates that a specific object affordance is not associated with an object or its part. For example, “floor” is associated with “walk” affordance, “*/door/knob” is associated with “pull” affordance. If the correspondence between an object and the five affordance labels cannot be established, then the association will not be performed to ensure the reliability of predicting affordance from the trained network. Figure 4.3 (a) presents an example of the label transformation. Using the transfer table, annotated data from ADE20K can be transferred to affordance ground truth data. For instance, seat base is transferred to sit affordance. Figure 4.3 (b) presents additional labeled simulated images for training. Affordances are directly annotated for the simulated dataset.

UNet. The deep learning method is based on a convolutional neural network (CNN) following the U-Net architecture [131]. The encoder-decoder architecture is efficient for training and implementation when the input and output images are of similar sizes. The ResNet50 network [132] is used as the encoder. The architecture of ResNet50 includes basic block and bottleneck block.

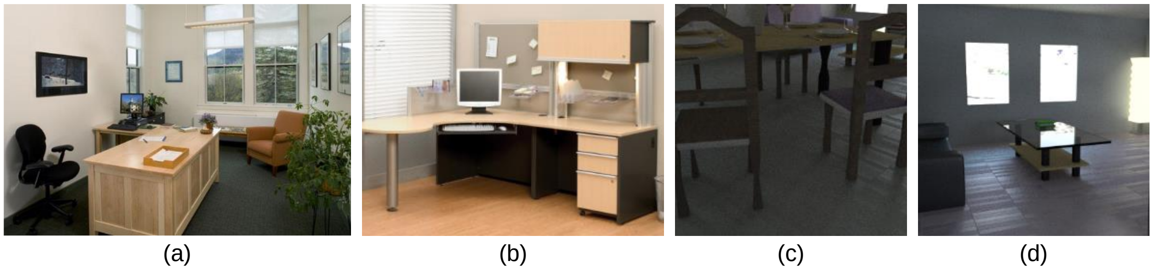


Figure 4.2. Sample images ((a)-(b) from ADE20K dataset [128], (c)-(d) from simulated images [129])

Table 4.1. Examples of the transfer table

Affordance	**Seat**	**Bottle**	**Floor**	***/door/knob**	**Countertop**	**…**
Walk (surfaces a human can walk)	0	0	1	0	0	…
Grasp (objects that can be grasped and moved by hands)	0	1	0	0	0	…
Pull (surfaces that can be pulled by hooking up fingers or by a pinch movement)	0	0	0	1	0	…
Place (elevated surfaces where objects can be placed on)	0	0	0	0	1	…
Sit (surfaces a human can sit)	1	0	0	0	0	…

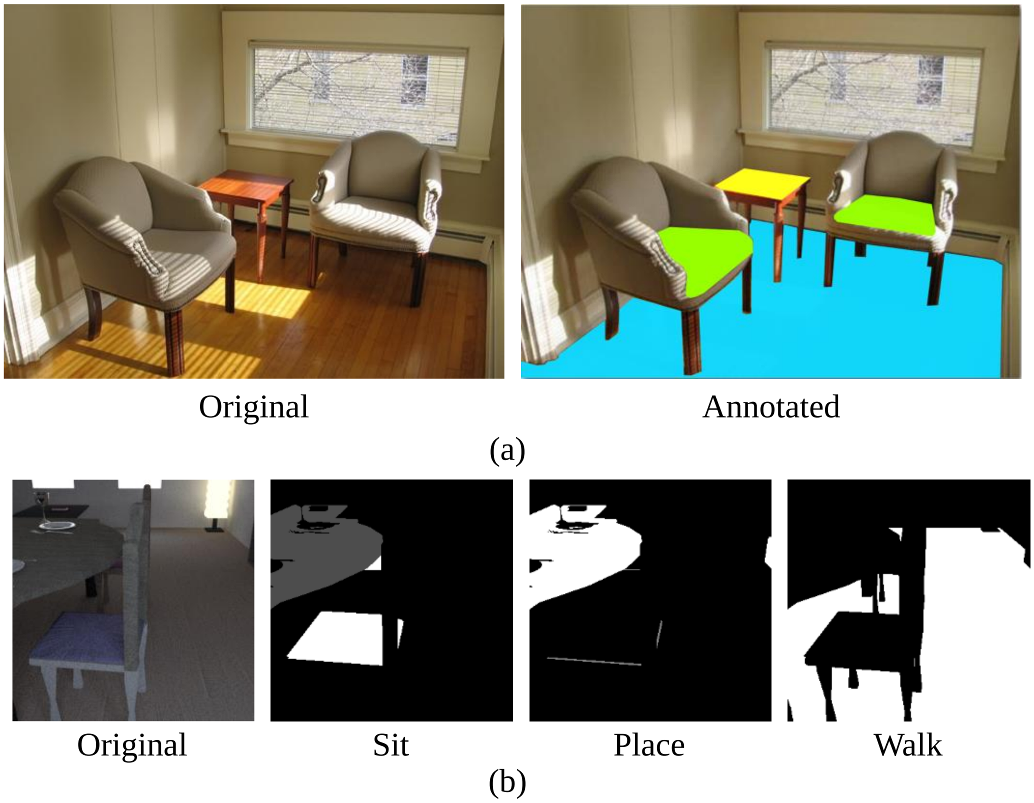


Figure 4.3. Example of annotated dataset. (a) using transfer table to transform original object labels in ADE20K dataset to affordance labels (green (seat base - sit); yellow (tabletop - place); blue (floor - walk)); and (b) annotated simulated image

The basic block consists of convolution, batch normalization, ReLU, and max-pooling layers. An initial 7*7 convolution with a stride of 2 is first applied, followed by the batch normalization and ReLU activation layer. Thereafter, a max-pooling operation is conducted with a kernel size of 3 and a stride of 2. The two steps can reduce the spatial size, and thus reduce the computation cost and the number of parameters in the deep layers. In the bottleneck, the network has four connected blocks. As the network progresses from shallow to deep block, the spatial size of the input image reduces to half, and the channel number doubles.

For the decoder network, a refinement module is used to integrate low-level features and high-level semantic information from the encoder network, thus enhancing mask encoding [133]. First, the refinement module upsamples feature map size to be the same as that of the skip connection from the encoder network. The bilinear interpolation method [134] is used to perform upsampling. Then, skip feature maps are concatenated with the local feature map. Last, convolution and batch normalization with ReLU activation are performed to compute the feature map of the next layer. Figure 4.4 illustrates the U-Net architecture.

DeepLabv3Plus. Figure 4.5 presents the architecture of the proposed network. The proposed deep learning network is adapted from the DeepLabv3Plus architecture [135]. The encoder-decoder structure is integrated with Atrous Spatial Pyramid Pooling (ASPP) module to encode multi-scale contextual information. The backbone encoder is built based on the EfficientNet-B4 network. The obtained feature maps are fed into the ASPP module to extract multi-scale features using multiple parallel filters with different dilated rates. This process can improve inversion accuracy with the ability to account for different object scales. The ASPP module contains a 1×1 convolution layer, three 3×3 convolution layers, and global average pooling. The sampling rate of the four convolution layers is 1, 12, 24, and 36, respectively. The batch normalization and ReLU activation layer are added followed by each convolution layer and pooling layer. The output of a single layer from the ASPP module is 256 channels.

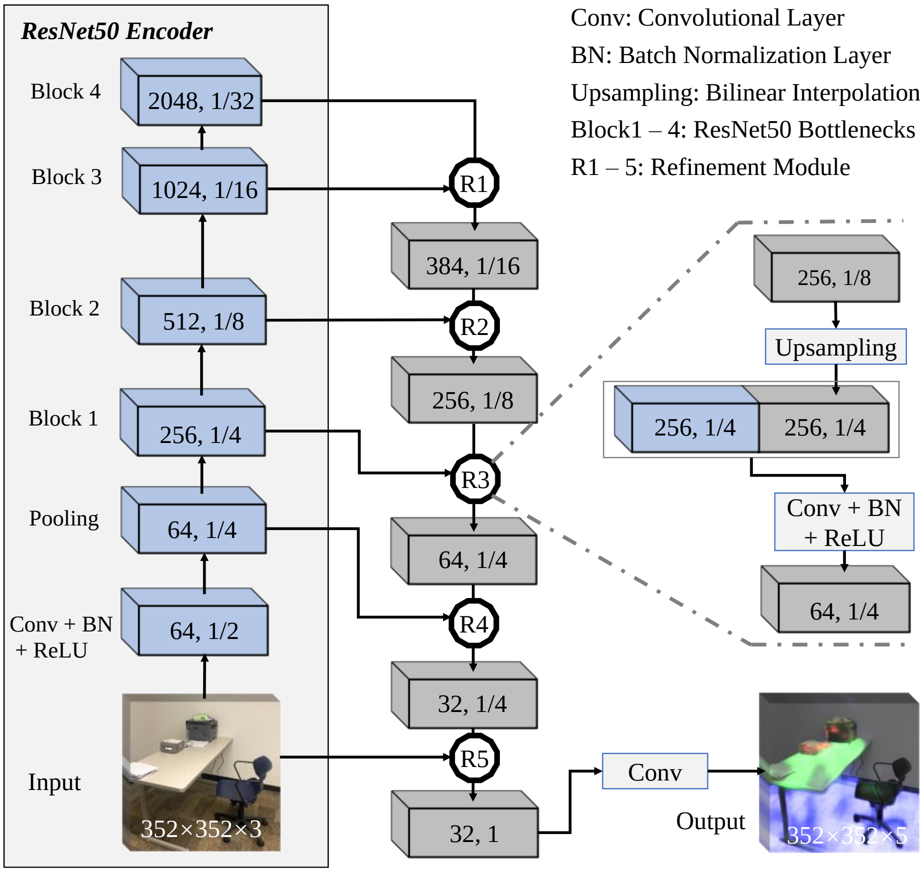


Figure 4.4. Semantic segmentation with the U-Net architecture

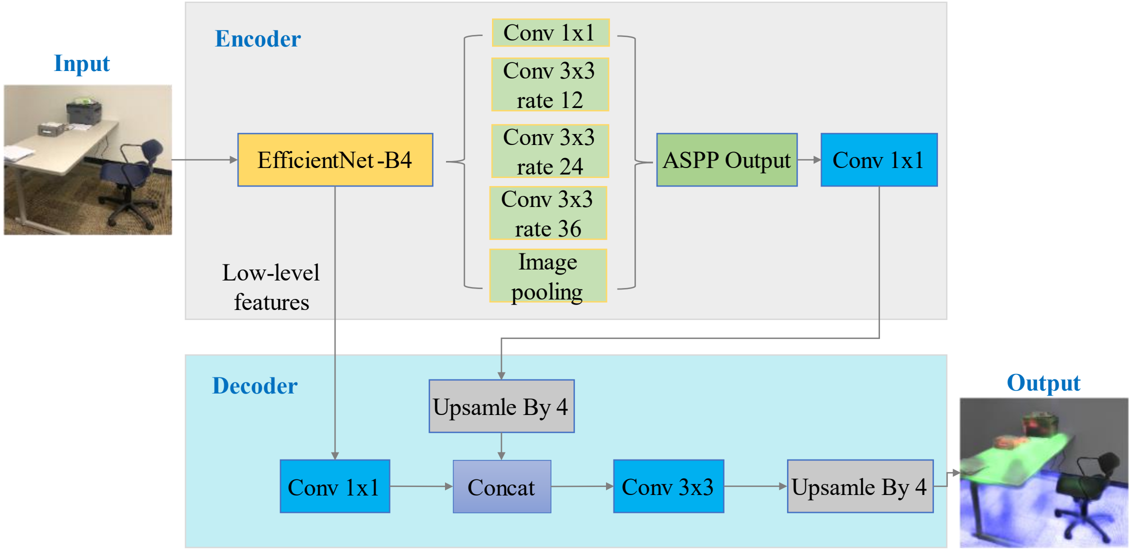


Figure 4.5. Semantic segmentation with the DeepLabV3Plus architecture

The five layers are concatenated together with 1280 channels. Subsequently, a 1×1 convolution with 256 output channels is applied to the concatenated layer to obtain a high-level feature map. At the decoder, upsampling and convolutions are performed to enlarge the feature map and obtain the final prediction. Each of the low-level features extracted from Resnet blocks would be passed to one 1×1 convolutional layer followed by batch normalization layer, ReLU activation layer, and dropout. Then these processed features are integrated into the corresponding high-level features in the decoder. The designs of the Resnet50 backbone and decoder are elaborated below.

Encoder: The backbone network based on the EfficientNet-B4 proposed in Tan and Le [136]. The EfficientNet-B4 network is small and fast on inference. The input is first fed into a 3x3 convolution, batch normalization, and activation layer. The outputs are then fed into 7 inverted residual blocks, also known as MBConv blocks [136], optimized by the squeeze-and-excitation method [92]. MBConv[N] represents an MBConv with an expansion factor of N. MBConv1 is a depth-wise separation block without the expansion operation. The MBConv6 block is the inverted residual block with an expansion factor of 6. The number of sub-blocks for the 7 MBConv blocks are 2, 4, 4, 6, 6, 8, and 2, respectively.

Decoder: The low-level features obtained from the backbone module is passed into one 1×1 convolutional layer followed by one batch normalization layer and one ReLU activation layer. At the same time, the high-level features obtained from the 1×1 convolution with 256 output channels in the encoder are up-sampled through bilinear interpolation with a scale factor is 4. The two feature maps are concatenated together to utilize low-level features. The concatenated feature map is fed into one 3 × 3 convolutional layer followed by one batch normalization layer and one ReLU activation layer. Sequentially, the output is applied to another bilinear interpolation up-sample layer with a scale factor of 4. The output has the same size as the original input images.

After segmenting the object affordance from the 2D RGB images as the areas of potential contamination, it is necessary to project the 2D labels to a 3D grid map for guiding robot navigation and disinfection. As depth images are registered to the reference frame of RGB images, the first step is to use the classical pinhole camera model [95] to obtain the point cloud of the environment. Given a pixel (x, y) and its depth d, its world coordinate (X, Y, Z) is computed by Eq. (1), where f_x, f_y are the camera focal length in pixel units, (c_x, c_y) represents the principal point that is usually at the image center. Figure 4.6 presents an example of the obtained point cloud. Each point stores information of world coordinates, label information, and its highest probability predicted by the network.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} x \\ y \\ d \end{bmatrix} \tag{1}$$

Second, octomap library [137] is applied to generate a 3D occupancy grid map, using the obtained point cloud as input. A voxel filter is used to reduce the size of the point cloud to accelerate the mapping process. In each voxel space, only one point is stored as one point is adequate to update an octree node. The voxel filter resolution is set to the same resolution as that of the occupancy map. The resolution of the occupancy map is set as 4 cm, which can provide adequate details in indoor environments while maintaining processing efficiency. Figure 4.7 presents an example of a 3D point cloud filtering. The image size is 960×540 and 518,400 points are generated for each frame. After using the voxel filter, the number of points reduces to 23,009 for the frame. Note that, the number of filtered points varied from frames due to noise in sensory data.

Since the camera is constantly moving, semantic information may continuously update. For instance, a small object may not be accurately segmented when the camera’s view angle is not at a favorable position. Hence, semantic information at the pixel level from different frames is fused to deal with this situation, see Figure 4.8. If two affordances are the same, the affordance will be kept, and the probability becomes the average of the two affordances. Otherwise, the affordance with higher confidence is kept and the probability is decreased to 0.9 of its original probability.

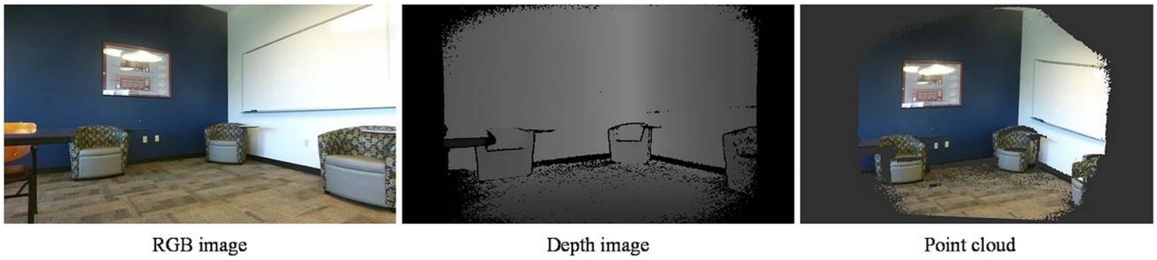


Figure 4.6. An example of point cloud generation

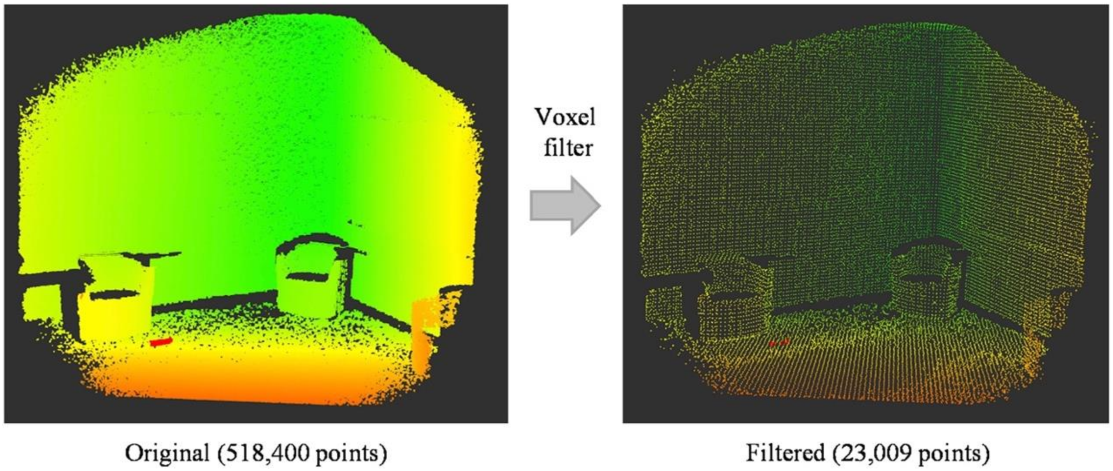


Figure 4.7. Voxel filter for 3D point cloud

Pseudo-code for semantic fusion

```
function fusion ($sem_1$, $sem_2$)
   if $sem_1.color$ is $sem_2.color$ then
      $sem_{fusion}.color = sem_2.color$
      $sem_{fusion}.probability = (sem_1.probability + sem_2.probability)/2$
   else
      if $sem_1.probability < sem_2.probability$ then
         $sem_{fusion} = sem_2$
      else
         $sem_{fusion} = sem_1$
   $sem_{fusion}.probability = 0.9 \times sem_{fusion}.probability$
return $sem_{fusion}$
```

Figure 4.8. Semantic fusion of two different frames

This process can allow the occupancy map to update the semantic information with a new prediction of higher confidence. After the above steps, the areas of potential contamination are predicted and projected to the 3D occupancy map, which can further guide the robotic disinfection.

Robot navigation

After mapping the areas of potential contamination, the next step is to generate robot motions to scan the areas with UV light for disinfection. The robot has a three-degree-of-freedom base and a six-degree-of-freedom manipulator. The robot needs to move to the objects needing disinfection. A hierarchical planning approach is adopted, which consists of global and local path planning. Global path planning provides an optimal path from the start to the goal, and local path planning outputs a series of velocity commands for the robot. The A* algorithm [138] is used to find a globally optimal path for the robot. The heuristic function $h(n)$ is used to guide the trajectory search toward a goal position. The A* algorithm can find the shortest path very efficiently. In this study, the Manhattan distance is used as the heuristic function that is defined in Eq. (2). This equation is used to calculate the Manhattan distance from any node (n (x_n, y_n)), to the goal (g (x_g, y_g)) in the graph.

$$h(x_n, y_n) = |x_n - x_a| + |y_n - y_a| \tag{2}$$

The cost function is given in Eq. (3), where $g(n)$ is the cost from starting point to node n, $f(n)$ is the total cost. The objective is to minimize the total cost.

$$f(n) = \mathrm{g}(n) + h(n) \tag{3}$$

Given a global path to follow, the local planner produces velocity commands for the robot. The Dynamic Window Approach (DWA) algorithm [139] serves as the local planner. The algorithm samples velocities in the robot's control space discretely within a given time window. The samples intersect with obstacles will be recognized and eliminated. An optimal pair of (v, w) for the robot is determined by maximizing the objective function defined in Eq. (4), which is dependent on (1) proximity to the global path, (2) proximity to the goal, and (3) proximity to obstacles.

$$\text{cost} = \alpha f_a(v, w) + \beta f_d(v, w) + \gamma f_c(v, w) \tag{4}$$

where $f_a(v, w)$ represents the distance between global path and the endpoint of the trajectory, $f_d(v, w)$ is the distance to the goal from the endpoint of the trajectory, $f_c(v, w)$ is the grid cell costs along the trajectory, α is the weight for how much the controller should stay close to the global path, β is the weight for how much the robot should attempt to reach the goal, and γ is the weight for how much the robot should attempt to avoid obstacles.

Experiments and results

Segmentation and 3D mapping of potential areas of contamination were validated in indoor environments, including a dining room, a conference room, and a restroom in a university campus building.

Evaluation datasets

The ADE20K dataset [128] and simulated dataset [129] were used to evaluate the performance of the network. The ADE20K dataset contains a total of 22,210 images with 20,210 training images and 2,000 validation images. The simulated dataset contains 2,530 synthetic images with 2,280 training images and 250 testing images. Hence, a total number of 22,490 images including both real and simulated images are used for training. The real and simulated images are first merged and then randomly mixed. Each mini batch for training can have samples from both datasets. In addition, data augmentation technique is used to increase the volume and variability of the training dataset. Training samples were augmented by cropping multiple image patches based on image quality and varying color and contrast of images to improve the capability and generalization of the trained model. The online augmentation method is used due to two reasons. First, as the model observes more samples in the training process, the model trained with online data augmentation can generalize better than the model trained with offline data augmentation [140]. Second, online augmentation does not need to store a large amount of augmented data on local disk. The validation set consists of 1,000 real images and 120 simulated images that are randomly split from the training dataset. The performance of the network was evaluated on 2,000 real images and 250 simulated images.

Metrics

To evaluate the performance of affordance segmentation, the metrics including the intersection over union (IoU), dice coefficient (DSC), and average precision (AP) were used to evaluate the network performance. These metrics have been widely used in evaluating the performance of semantic segmentation [141–143]. The IoU is the area of overlap between the predicted segmentation and the ground truth divided by the area of union between the predicted segmentation and the ground truth. The maximum IoU value is 1 representing perfect segmentation. The IoU metric is defined in Eq. (5), where $\mathbf{Y}_{ai}$ is the ground truth for affordance a at pixel $i \in I$, $\hat{\mathbf{Y}}_{ai}$ represents predicted affordance. The binarized prediction of the network is used to compute the IoU. The threshold values 0.2 and 0.5 are used in the experiment.

$$\mathrm{IoU}_a = \frac{\sum_{i \in I}(\mathbf{Y}_{ai}=\mathbf{1} \cap \hat{\mathbf{Y}}_{ai}=\mathbf{1})}{\sum_{i \in I}(\mathbf{Y}_{ai}=\mathbf{1} \cup \hat{\mathbf{Y}}_{ai}=\mathbf{1})} \tag{5}$$

$$\hat{\mathbf{Y}}_{ai} = \begin{cases} 1 & \text{if } p > \text{threshold} \\ 0 & \text{else} \end{cases} \tag{6}$$

The DSC is similar to the IoU, which is another measure of overlap between prediction and ground truth. This measure ranges from 0 to 1, where a DSC of 1 denotes perfect and complete overlap. The DSC is defined in Eq. (7).

$$DSC_a = \frac{\sum_{i \in I} 2 \times (\mathbf{Y}_{ai}=\mathbf{1} \cap \hat{\mathbf{Y}}_{ai}=\mathbf{1})}{\sum_{i \in I}(\mathbf{Y}_{ai}=\mathbf{1}) + (\hat{\mathbf{Y}}_{ai}=\mathbf{1})} \tag{7}$$

The AP metric summarizes a precision-recall curve as the weighted mean of precisions achieved at each threshold. AP is not dependent on a single threshold value since it averages over multiple levels. The AP is defined in Eq. (8), where P_n and R_n are the precision and recall at the nth threshold, and P_n is defined as precision at cut-off n in the list.

$$\mathrm{AP} = \sum_n (R_n - R_{n-1})\, P_n \tag{8}$$

Implementation details

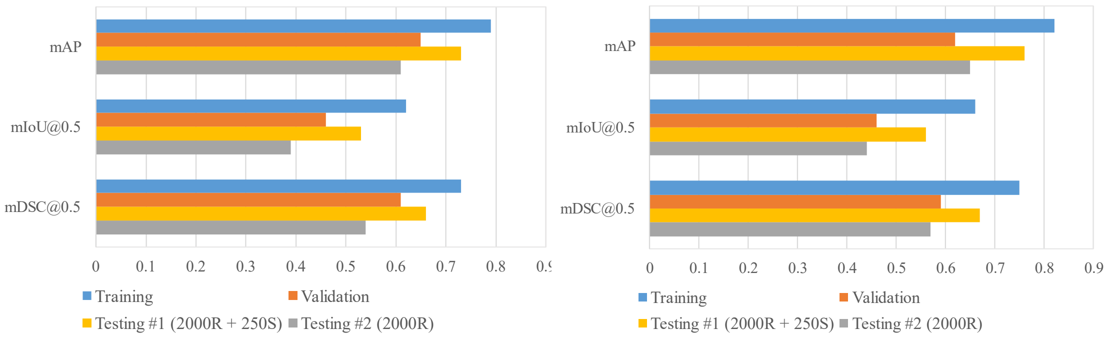
The models were trained on a workstation running Ubuntu 16.04 with dual Intel Xeon Gold 4114 CPU, 128 GB RAM, and NVIDIA RTX A6000 with a PyTorch backend

[144]. The network was trained using RMSProp optimizer [145] with a learning rate of 0.0001 and batch size of 16. The ResNet50 and EfficientNet-B4 were initialized with weights pretrained on ImageNet [146]. The pretrained weight was further trained on the dataset without freezing any weights. The early stopping technique [147] was adopted to prevent overfitting. Specifically, the network is trained on the training set, and if the loss on the evaluation set does not decrease for 20 epochs, the training process will stop, and the best model observed on the evaluation set will be saved. The performance of the network is evaluated on the testing dataset.

Results on semantic mapping

Figure 4.9 presents the results of the affordance segmentation on the training, validation, and testing sets for revised UNet and DeepLabv3Plus. For both the two models, the training set achieved the highest mAP, mIoU, and mDSC since the model is optimized using this set. The testing set #1 achieved the second-highest scores. The difference of all the three metrics between the training set and testing set #1 is not greater than 0.1, suggesting that the network is well-trained to predict the unknown data. However, testing set #2 achieved the smallest scores among the four datasets. This is because the training set contains both real and simulated images, however, testing set #2 only contains real images. The synthetic images have a different distribution from real ones, which leads to a better performance of the network on a combination of both samples. On the training, testing set#1, and testing set#2, DeepLabv3Plus achieves better performance compared to UNet model. Specifically, on testing set #2, DeepLabv3Plus has an improvement of 0.04, 0.05, 0.03 on mAP, mIOU, and mDSC compared to UNet. The performance of the trained model on individual affordance is detailed below.

Table 4.2 presents the network performance of UNet for individual affordance on testing set #2. The results show a strong variation on the performance for different affordances. For instance, affordance walk achieves the highest IoU and AP scores, which is attributed to a relatively large sample size compared to other affordances such as grasp and pull. In addition, the walking surface often covers large areas in the scene.



(a) (b)

Figure 4.9. The performance of the network on the training set, validation set, and two testing sets. (Testing #1 consists of 2000 real images and 250 simulated images and testing #2 only contains 2000 real images. mAP, mIoU, and mDSC are the average of AP, IoU, and DSC over all classes). (a) UNet; and (b) DeepLabv3Plus

Table 4.2. Performance of UNet for individual affordance on testing set #2

Affordance	DSC@0.2	DSC@0.5	IoU@0.2	IoU@0.5	AP
Walk	0.84	0.87	0.72	0.77	0.94
Grasp	0.51	0.51	0.31	0.30	0.50
Pull	0.36	0.17	0.23	0.10	0.32
Place	0.57	0.56	0.38	0.37	0.62
Sit	0.64	0.60	0.47	0.42	0.67

Pull has the lowest prediction accuracy among the five affordances. The pull affordance represents objects that can be pulled such as doorknobs and cabinet handles. These objects are relatively small and have a small sample size in the dataset. The walk, grasp, place, and sit affordances achieved DSC and AP scores higher than 0.5, indicating the usability of the proposed method in built environments.

Table 4.3 presents the network performance of DeepLabv3Plus for individual affordance on testing set #2. The performance on individual affordance is significantly improved compared to UNet. In particular, sit affordance has the largest improvement with an AP improvement of 0.09 followed by grasp affordance 0.06. Pull affordance has an AP improvement of 0.02.

The performance of the proposed method is also compared with two studies that achieved the best performance of affordance segmentation using RGB images captured in built environments. In [148], a Multi-scale CNN was developed to segment affordance in RGB images. Roy and Todorovic [148] achieved IoU scores of 0.67 and 0.34 on the walk and sit affordances, respectively. The proposed network achieved 0.81 and 0.53 for IoU scores for walk and sit affordances at the threshold of 0.5. Lüddecke et al. [130] reported the best AP scores for sit, grasp, pull, place, and walk affordance are 0.54, 0.30, 0.02, 0.45, and 0.96. The proposed network achieved AP scores of 0.76, 0.56, 0.34, 0.67, and 0.96 for sit, grasp, pull, place, and walk affordances on the same test set. Hence, it can be concluded that the proposed semantic segmentation method is at least comparable with the state-of-the-art.

A Kinect sensor is used to perform RTAB-Map SLAM and generate the semantic 3D occupancy map using the network. The frame size provided by the Kinect is 960*540 pixels. Figure 4.10 shows the predicted affordances in images captured in the building. Walk, Grasp, Pull, Place, and Sit affordances are color-coded, and the color intensity represents their corresponding probabilities. The results indicated that walk, place, and sit affordance are accurately segmented in the images.

Table 4.3. Performance of DeepLabv3Plus for individual affordance on testing set #2

Affordance	DSC@0.2	DSC@0.5	IoU@0.2	IoU@0.5	AP
Walk	0.88	0.89	0.79	0.81	0.96
Grasp	0.53	0.55	0.38	0.36	0.56
Pull	0.39	0.14	0.24	0.08	0.34
Place	0.64	0.60	0.47	0.43	0.67
Sit	0.69	0.69	0.53	0.53	0.76

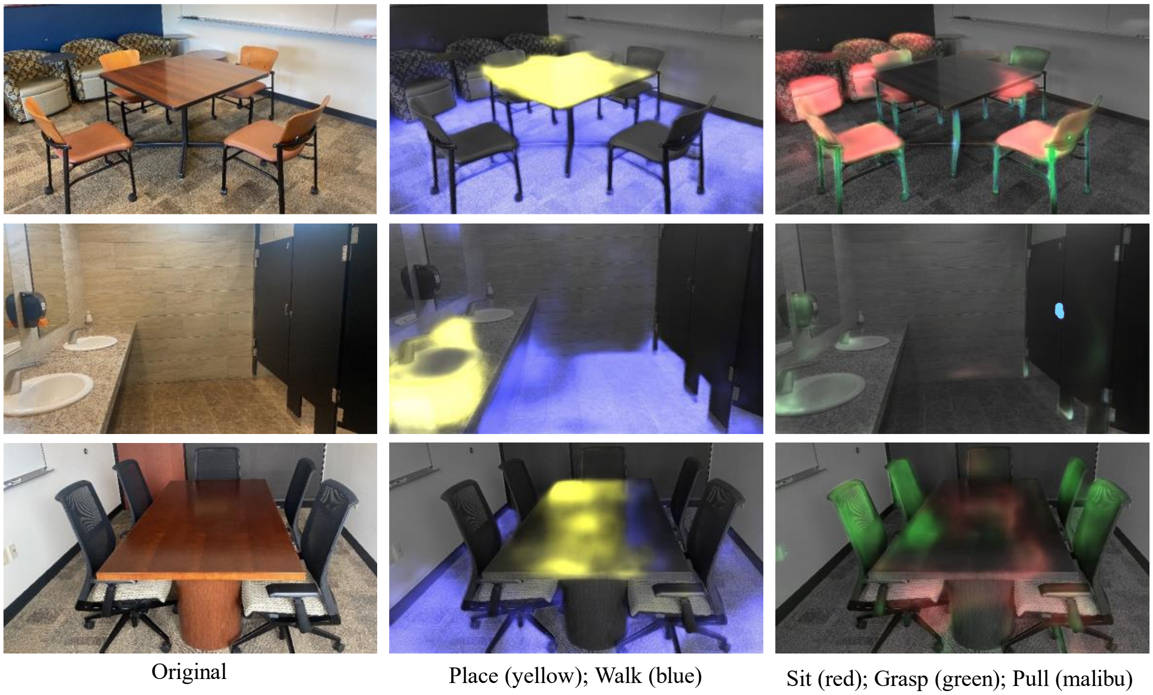

Original
Place (yellow); Walk (blue)
Sit (red); Grasp (green); Pull (malibu)

Figure 4.10. Exemplary results of affordance segmentation using the improved UNet

Figure 4.11 shows the results of affordance segmentation using the revised DeepLabv3Plus. Compared with the results obtained using UNet, the proposed method achieves a much better performance on these images. For example, the countertop and table surface are accurately and completely segmented. However, UNet can only segment a part of these surfaces, demonstrating the efficiency of our method revised from DeepLabv3Plus.

Figure 4.12 presents the results of 3D semantic occupancy mapping using the UNet network. Images were obtained to perform RTAB-Map SLAM to obtain camera poses. Thereafter, semantic reconstruction was conducted using recorded video and camera trajectory. At a resolution of 4 cm, the indoor scene can be properly reconstructed. The results indicate that the proposed method can successfully segment affordances. The walk, place, sit, and grasp affordances are reasonably segmented. In the dining room, small tablet arm of sofa on the left side is correctly segmented as place affordance. However, small objects like doorknobs are not correctly recognized in the semantic map. In addition, a large part of the table surface is not correctly segmented. This is possibly due to the small size of the training data. The occupancy map can be continuously updated during the robot disinfection action to address the incorrect segmentation.

In comparison, the revised DeepLabv3Plus is also used to reconstruct indoor environment with the same settings. Figure 4.13 shows the results of semantic reconstruction. As indicated in the figure, the revised DeepLabv3Plus can better reconstruct indoor environment with accurate semantic information.

The proposed semantic mapping approach can also run offline to generate a 3D semantic map with a high resolution. Figure 4.14 shows the reconstruction results with a resolution of 2 cm with object information. Since the processing time significantly increases with a high resolution, the recorded video is played using a rate of 0.2. As indicated, a high-resolution map has the potential to capture more details.

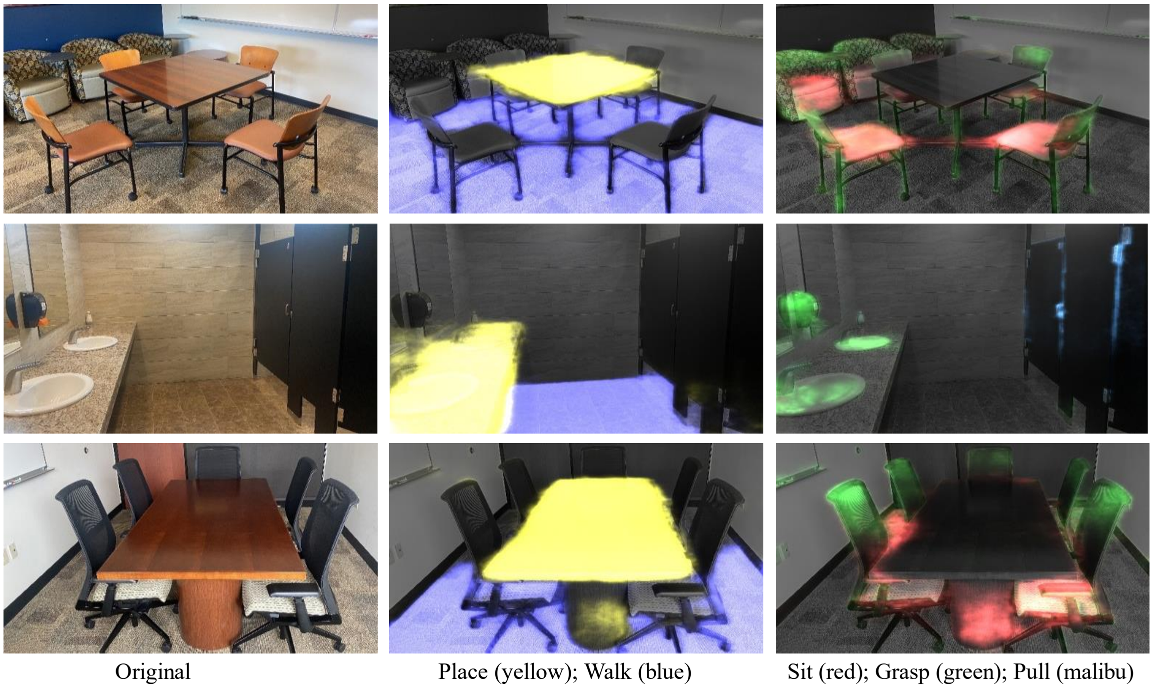

Original
Place (yellow); Walk (blue)
Sit (red); Grasp (green); Pull (malibu)

Figure 4.11. Exemplary results of affordance segmentation using the improved DeepLabv3Plus

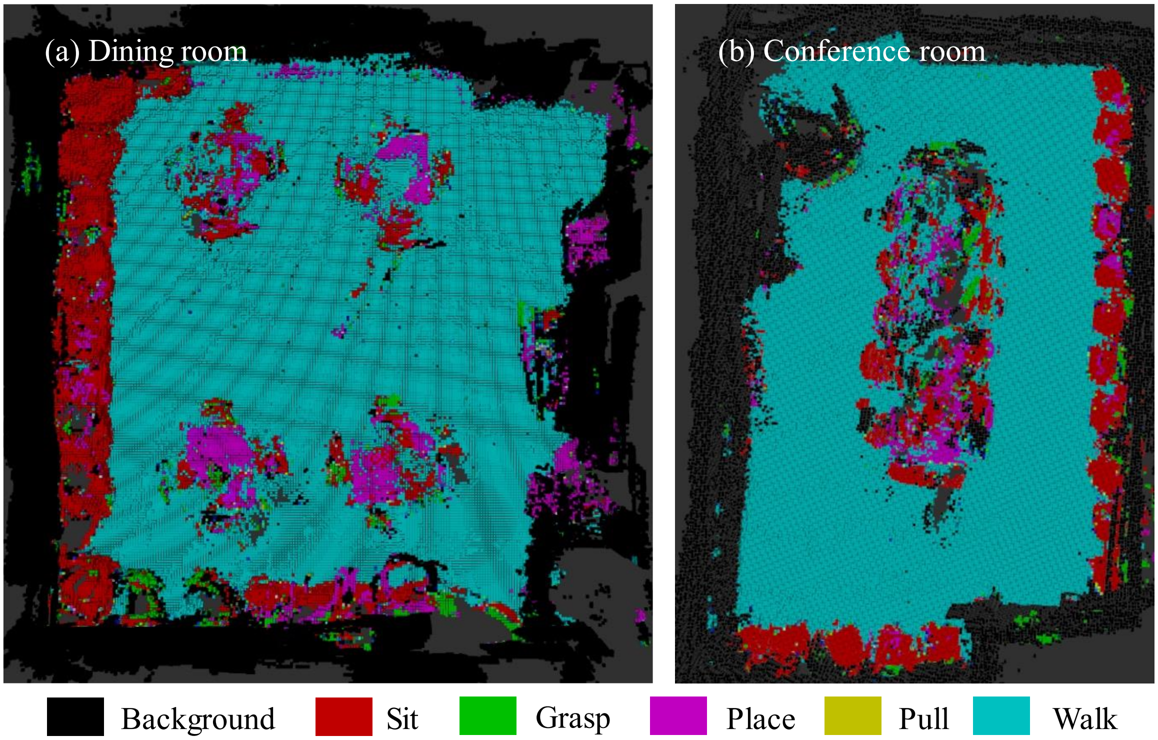


Figure 4.12. Results of 3D semantic reconstruction using the improved UNet

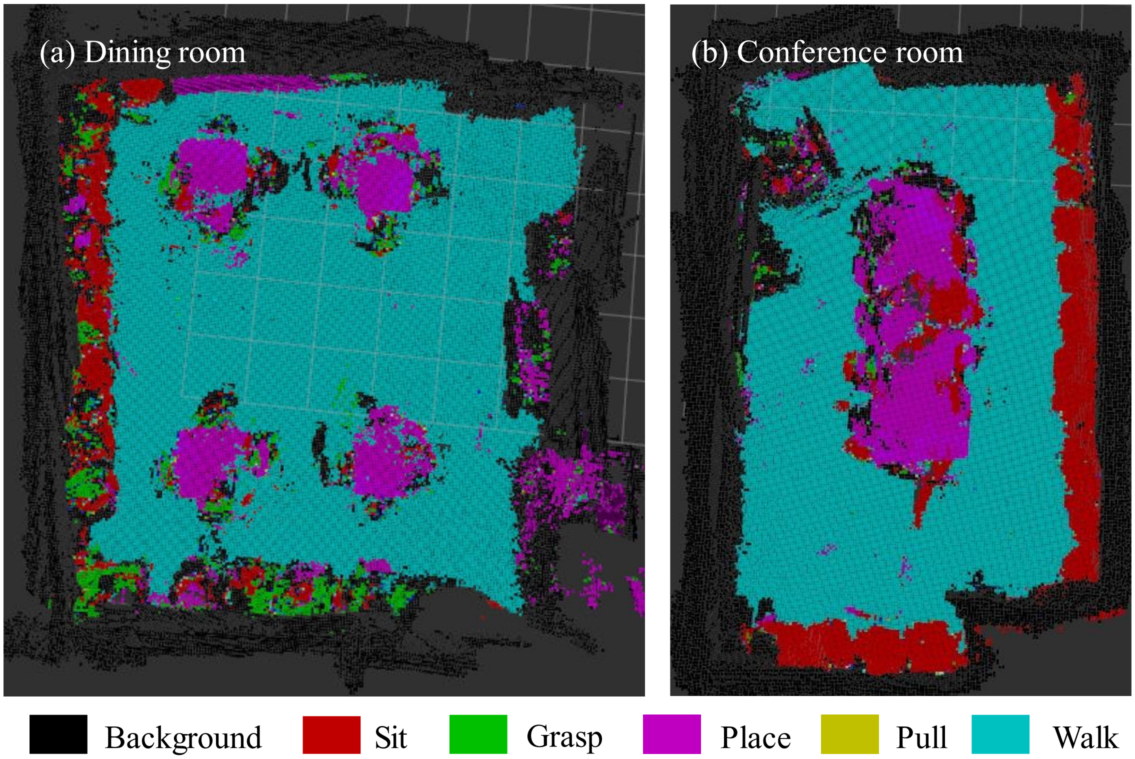


Figure 4.13. Results of 3D semantic reconstruction using the improved DeepLabv3Plus

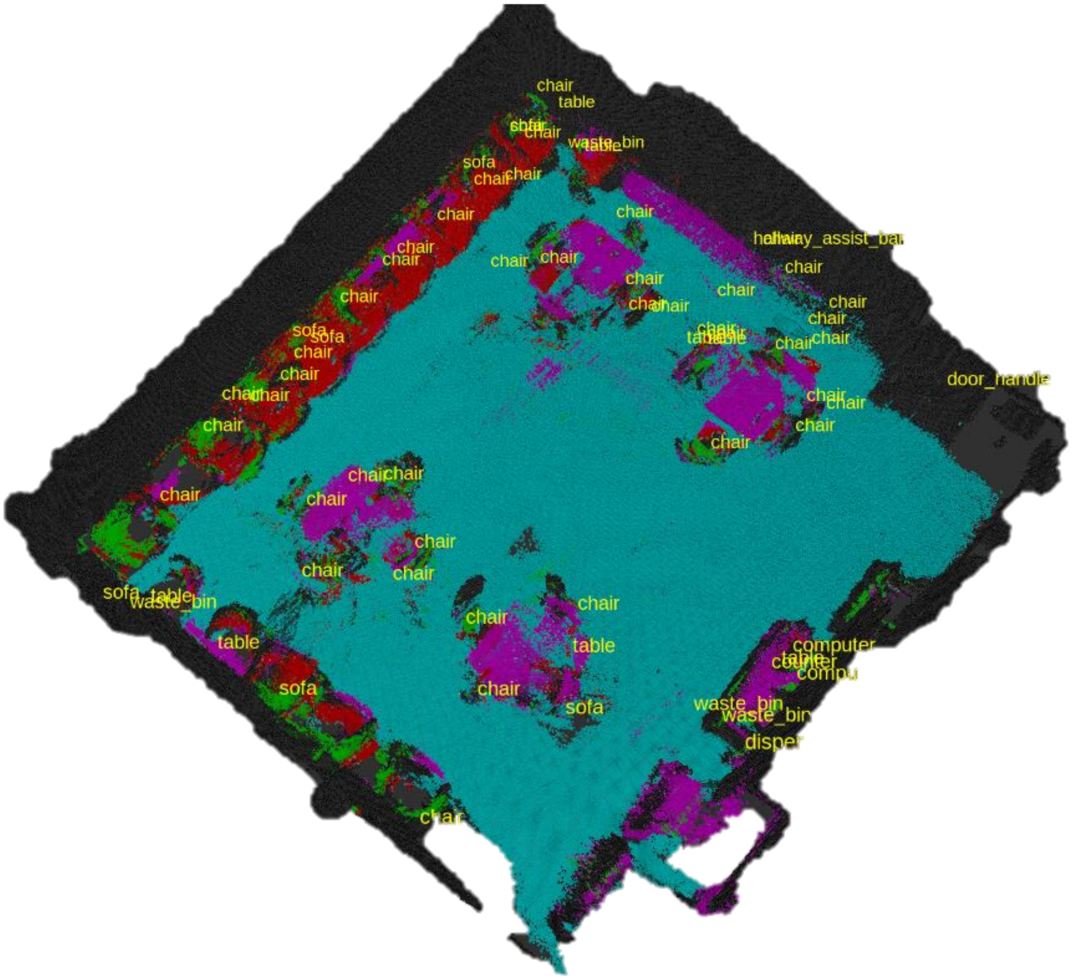


Figure 4.14. 3D object affordance map with object information at a resolution of 2cm

Processing time

The processing time for each step was assessed in this study. Table 4.4 presents the average time spent on each processing stage. The occupancy map resolution is set as 4 cm. As shown in the table, the processing frequency of the entire system is about 3.2 Hz and 4.0 Hz for image size 960×540 and 512×424, respectively. The octomap update is the most time-consuming step in the system, since it requires raycasting to update the occupancy map. The ray casting is used to clear all the voxels along the line between the origin and end point. The SLAM method achieves a high frame rate to track the camera in real time. Semantic segmentation and semantic point cloud generation are also run at a very high frame rate. Our system runs at 3.2 Hz for high-resolution image streaming, which can be adapted to most indoor online applications.

Moreover, the occupancy map resolution significantly impacts the processing time. Figure 4.15 presents the relationship between processing time and occupancy map resolution for image sizes 960×540 and 512×424. The result indicates that the processing time significantly decreases with decreasing map resolution. In addition, processing time increases as the image size increases under different occupancy map resolutions. When the resolution is 6 cm, the processing time can reduce to 206.5 ms and 134.2 ms for image size 960*540 and 512*524, respectively. However, a lower resolution may not capture detailed information, especially for small objects.

Implementation of robotic navigation

The 3D occupancy map collected in the built environment was loaded into the simulation platform to test robot navigation. Table 4.5 shows the performance of path planning of the robot. The average computing time for 20 simulation experiments is low, and generated paths can successfully avoid collision with obstacles. The results demonstrate the efficiency and effectiveness of the robot path planning method. Figure 4.16 presents two representative examples of the path planning of the robot. The robot moves to the proximity of objects needing disinfection.

Table 4.4. Average processing time for each step (Process with * and process with ** processed at the same time)

Step	**Consumed time**	
	960×540	512×424
SLAM *	50.2 ms	35.4 ms
Semantic segmentation **	25.1 ms	20.4 ms
Semantic point cloud generation **	28.3 ms	13.7 ms
Octomap update (resolution 4 cm) **	254.6 ms	215.7 ms
Total	308.0 ms	249.8 ms

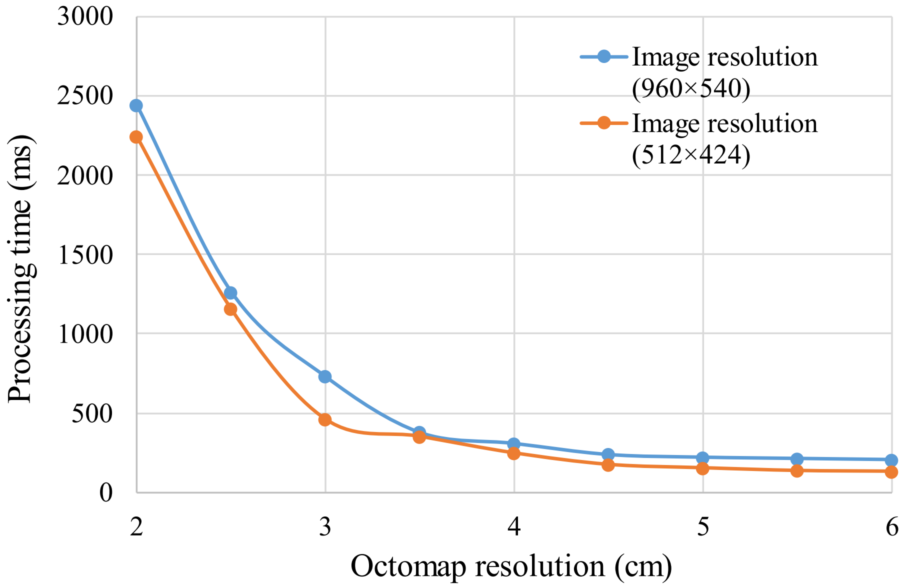


Figure 4.15. Influence of image size and occupancy map resolution on processing time

Table 4.5. Performance of base robot path planning

Simulation	Computing time (second)			Number of cases
case	Average	Minimum	Maximum	without collision
20	0.231	0.2	0.375	20

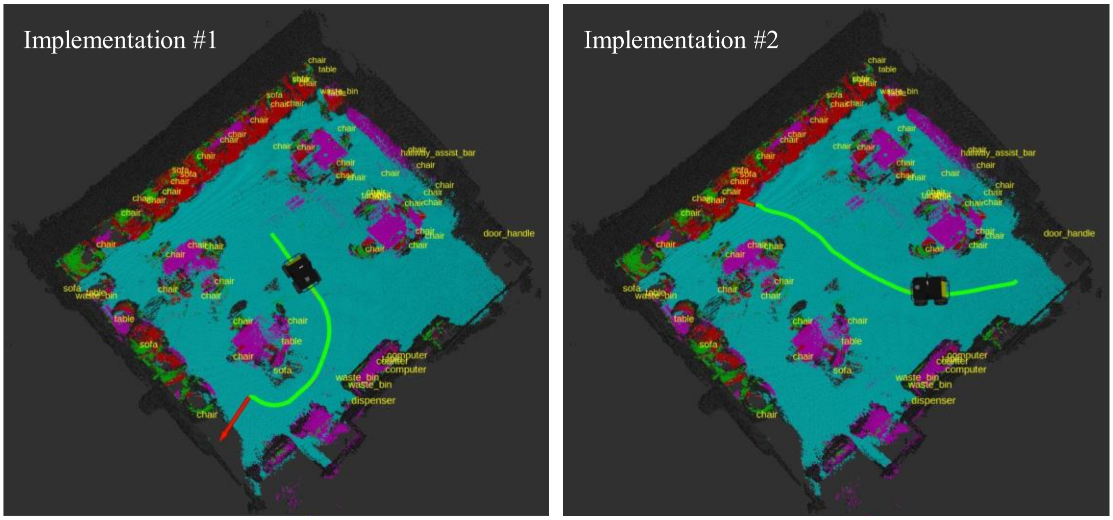


Figure 4.16. Implementation of robot navigation. Red arrow is the pose of a goal point. The green line is the trajectory of the robot

Conclusions

The mass-gathering built environments such as hospitals, schools, airports, and transit systems harbor a variety of pathogens that may cause diseases. Frequent disinfection is critical for preventing the outbreak of infectious diseases in built environments. However, the manual disinfection process is labor-intensive, time-consuming, and health-undermining, highlighting the values of automated and robotic disinfection. To reduce the public health risks and alleviate the extensive labor efforts, this study presents a framework and algorithmic techniques to enable a robot to automatically use UV lights to disinfect the areas of potential contamination. Using SLAM technique, the robot is able to create occupancy map and estimate its pose. The areas of potential contamination are detected and segmented based on object affordance. Affordance information can guide the robot to focus on hot spots and thoroughly disinfect potentially contaminated areas. Thus, the developed methods will help reduce seasonal epidemics, as well as pandemics of new virulent pathogens.

There are some limitations that need to be addressed in future studies. First, human presence has not been incorporated into the framework. To further expand the capabilities of robots and to allow the robots to clean and disinfect the built environment, the robot needs to be able to operate in the presence of humans. In future research, human sensing will be incorporated, and human behaviors will be modeled to enable the robots to operate in a safe manner. Second, in this study, the robot operation is at the room level and only a single robot is considered. For disinfection in large-scale facilities, multiple robots may be needed. The 3D mapping and the coordination of the multiple robots will be an interesting and useful future study. Third, the network reported a low accuracy on pull affordance that represents small structures such as doorknobs and cabinet handle. Finally, in the semantic 3D map, it is difficult to recognize small objects such as bottles and doorknobs. Although map resolution can be increased to capture small objects, the processing time will significantly increase. Future research is needed to develop algorithms to optimize 3D semantic reconstruction at a high frequency.

CHAPTER FIVE
OBJECT SURFACE MATERIAL RECOGNITION

Introduction

Hospitals, nursing homes, airports, and buildings are hotbeds for pathogen colonization and transmission, resulting in a massive number of infections among the people who occupy these facilities [149]. Outbreaks of infectious diseases lead to illness and death, imposing significant burdens on the healthcare systems, reducing productivity, and leading to enormous economic losses. For example, the COVID-19 pandemic has led to over 259 million confirmed cases and 5.1 million deaths [1]. The number of infections and deaths continues to increase with the emergence of more infectious variants of COVID-19, increasing the fear of future surging waves of infections. Healthcare facilities are particularly of concern during the pandemic given the influx of infected patients needing treatment. In healthcare facilities, surfaces can be contaminated through hand touching, respiratory droplets, or bodily secretions. This contamination can cause cross-transmission among patients and between patients and healthcare providers, jeopardizing people's health, and the normal operations of hospitals [2]. In fact, before the COVID-19 pandemic, the United States Centers for Disease Control and Prevention (CDC) estimated that nearly 1.7 million patients are infected during hospitalization, resulting in 98,000 associated deaths [3]. This statistic highlights the urgency and importance of proper surface disinfection to mitigate the transmission of infectious bacteria and viruses and to reduce the possibility and number of healthcare-acquired infections (HAIs) [150].

Healthcare facilities harbor a variety of pathogens that colonize a wide spectrum of surfaces made from different materials. The object surface materials have significant impacts on pathogen colonization and transmission, and thus require different disinfection modes, parameters, and procedures to ensure complete and efficient disinfection. For example, a recent study by Chin et al. [151] suggested that certain pathogens, such as SARS-CoV-2, can stay infectious for as long as 7 days on metal and plastic surfaces, while SARS-CoV-2 may survive for only 2 days on fabric. Furthermore, the transfer efficiency or transmission rate of bacteria or viruses to hands from a surface differs between materials [152]. For example, the transfer efficiency of MS2 can reach 19.3% on glass surfaces but only 0.3% on fabric surfaces under a relative humidity of 15% to 32% [152]. Therefore, the materials of contaminated object surfaces must be considered for appropriate robotic disinfection to occur. However, this research topic has not yet been investigated.

The objective of this study is to develop a new deep-learning-based method to enable the robot to recognize the material types of object surfaces requiring disinfection. To achieve the objective, a new deep-learning network is proposed to classify the materials of object surfaces needing disinfection. Our designed network innovatively integrates multi-level Convolutional Neural Network (CNN) features, multi-scale CNN features, and a texture encoder network in an end-to-end learning fashion, which has not been integrated by existing studies. The multi-level CNN features can capture high-level abstract representations of the material and low-level texture and color information, which can enhance material representation ability of the network. The multi-scale features are captured by the Atrous Spatial Pyramid Pooling (ASPP) with multiple resampling rates, allowing the network to learn spatial repetitive features of material textures. The texture encoder network can capture texture details and local spatial information from different levels. The orderless features and ordered spatial information are then balanced with a bilinear model. The proposed network extracts rich features for accurate material representation, achieving state-of-the-art results on six public material datasets.

Literature review

Material recognition is a long-standing challenging problem and numerous methods have been developed to address it. Traditional material recognition can be simplified into three steps. First, handcrafted features are performed by extracting descriptors from an image; second, texture information is modeled by the global distribution of local descriptors; third, a classification model is trained to classify materials in the image. Thus far, a number of studies have examined the performance of traditional material classification methods. For instance, Caputo et al. [153] adopted an appearance-based method for material classification. Their approach used the Local Binary Pattern (LBP) method to extract feature descriptors, which are concatenated to a texton histogram to represent the image. The Support Vector Machine (SVM) algorithm was then used to classify materials using extracted features. In [154], Scale-Invariant Feature Transform (SIFT) was used to extract features, and the bag-of-words (BOW) approach was used to make frequency histograms of features. A discriminative maximum entropy method was then developed to estimate the posterior distribution of material labels given image features. The main drawback of these methods is that handcrafted features rely heavily on domain expertise, are usually not robust, and are computationally intensive due to high dimensions.

To address these limitations, Convolutional Neural Network (CNN) was applied to learn features from an input image. Compared to handcrafted features, the performance of automatically learned features is found to be more robust in image classification [155]. In [156], the CNN-based feature extraction was first demonstrated to be efficient for material and texture recognition. In their method, CNN was truncated at the level of the convolutional layer to obtain so-called local image descriptors. As the CNN’s output is highly correlated to the spatial order of pixels, traditional orderless pooling encoders (e.g., Fisher Vector (FV); Vector of Locally-Aggregated Descriptors (VLAD); BOW) were used to map local image descriptors to a feature vector, which is suitable for the classification task. The feature vector was then fed into an SVM classifier to predict material labels. The combination of FV and CNN (FV-CNN) achieved superior results in image classification. In a follow-up study by Song et al. [157], learnable locally-connected layers were attached to the output of FV-CNN for feature refinement. However, the FV-CNN architecture learns CNN features, feature encoding, and material classifier separately, which has not fully utilized the labeled dataset.

In recent years, there has been an increasing amount of literature on texture and material recognition using end-to-end deep learning methods with convolutional neural networks

(CNNs) For instance, Bell et al. [158] investigated the performance of three popular CNN architectures that are AlexNet, VGG-16, and GoogLeNet on material recognition. The finetuned AlexNet yielded good results on Flickr Material Database (FMD). In [159], a CNN-based Differential Angular Imaging Network (DAIN) was developed to integrate multi-view images to recognize outdoor materials. Original and differential angular images were fed into the network and combined with their final prediction results. The results indicated that with differential angular images, the prediction accuracy was increased. However, typical CNNs with fully connected (FC) layers are typically not ideal for material recognition due to the need for a spatially-invariant representation describing the feature distributions instead of concatenation. To address this issue, Zhang et al. [25] developed the Deep Texture Encoding Network (Deep-TEN) with an orderless feature pooling encoder network. The encoding layer integrates dictionary learning and residual encoding pipeline as a CNN layer. Building upon the Deep-TEN, a Deep Encoding Pooling Network (DEP) was designed to integrate high-level spatial information and orderless features for the task of material recognition [26]. The DEP added a global average pooling layer based on the Deep-TEN for the outputs of convolutional layers. A bilinear model was then used to merge the outputs from the pooling layer and texture encoding layer. Zhai et al. [160] proposed a Deep Multiple-Attribute-Perceived Network (MAPNet) to perceive multiple visual attributes for texture recognition. The MAPNet was based on a multi-branch architecture that enables visual texture attributes learning synergistically. The CNN features from each branch are fed into a spatially-adaptive global average pooling for feature aggregation.

Methodology

The disinfection robot can navigate in a building and recognize potentially contaminated areas based on our developed method [161]. The robot then moves to the proximity of the contaminated objects needing disinfection and adapts UVC light scanning trajectories. The limitation of the robot is the lack of capability in recognizing surface materials, which impacts its disinfection efficiency. To address this limitation, a deep learning network is proposed to recognize the object surface materials captured by the disinfection robot. The proposed network integrates multi-level CNN features, which leverages both low and high-level information to capture semantic and texture information. High-level features can capture semantic features, which are abstract representations of the material. Low-level features can capture more subtle details such as texture information. The Atrous Spatial Pyramid Pooling (ASPP) module can extract multiscale features by resampling feature maps at multiple rates. The ASPP module increases the size of the receptive field without compromising the feature map resolution. Our network further integrates an encoder component, which combines both orderless and local spatial feature pooling. The encoder can preserve texture and ordered spatial information from different layers, which can better capture spatially invariant features of materials. Figure 5.1 presents an illustration of the proposed network, which is composed of four components: the backbone, a multi-level feature fusion, Atrous Spatial Pyramid Pooling (ASPP), and an encoder. Each component is detailed below.

Backbone. The classification model is designed based on the EfficientNet-B4 network proposed in Tan and Le [136]. The EfficientNet-B4 network is small and fast on inference. The input is first fed into a 3x3 convolution, batch normalization, and activation layer. The outputs are then fed into 7 inverted residual blocks, also known as MBConv blocks [136], optimized by the squeeze-and-excitation method [92]. MBConv[N] represents an MBConv with an expansion factor of N. MBConv1 is a depth-wise separation block without the expansion operation. The MBConv6 block is the inverted residual block with an expansion factor of 6. The number of sub-blocks for the 7 MBConv blocks are 2, 4, 4, 6, 6, 8, and 2, respectively.

Multi-feature integration. The multi-level features are innovatively extracted in this study to capture the low-level texture and color information and the high-level semantic information. Specifically, the outputs from the last three MBConv blocks are extracted and separately fed into the ASPP component. The multi-level CNN features are extracted to utilize features from different layers of the EfficientNet-B4 network.

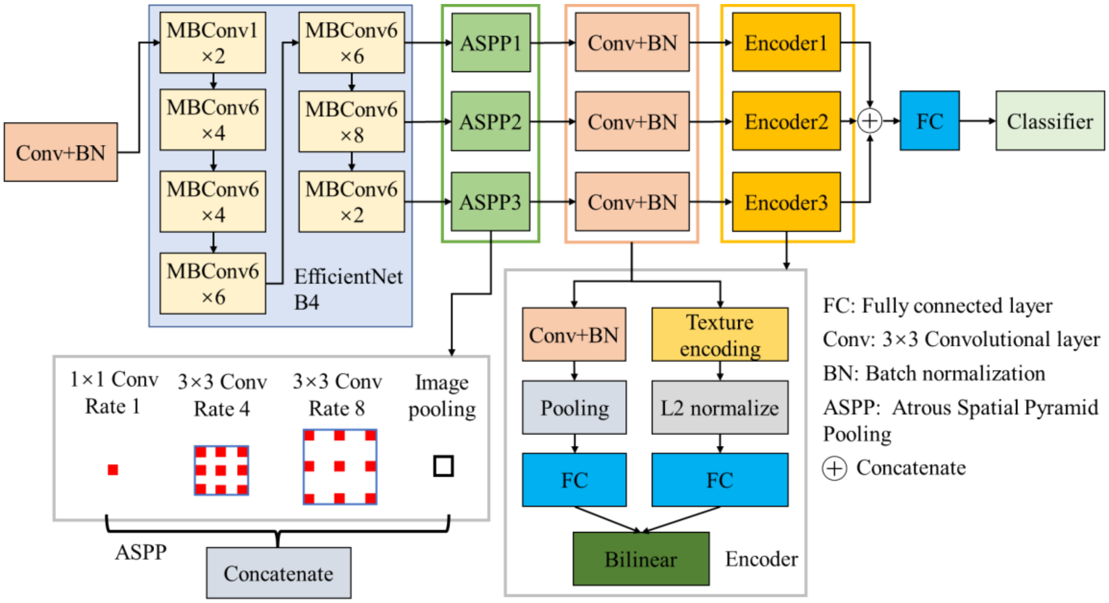


Figure 5.1. Flowchart of the proposed network

The reason for multi-layer feature fusion is that texture details learned from the shallow layers tend to vanish with going deeper into the layer. The texture details learned from the low-level features are important for material recognition. The features from multiple layers can capture complementary information and a combination of these features can improve performance.

ASPP. The ASPP is used to obtain multi-scale context information [162]. The outputs from the last three MBConv blocks are separately fed into the three ASPP layers. The ASPP layer consists of three Atrous convolutions with rates of 1, 4, and 8 and one global average pooling layer. Different rates of Atrous convolution have different sizes for their receptive fields. Since material textures are typically translationally invariant, a larger size for the receptive field can better capture spatial repetition features. The ASPP layer can extract multi-scale features while preserving the resolution of the features. The features extracted from multiple rates and the pooling layer are fused as the global features. The kernel sizes for Atrous convolutions are 1×1, 3×3, and 3×3. Atrous convolution is a generalized standard convolution and expands the window size to capture large features without adding computational cost by inserting zero-values into the convolution kernels. The outputs of the ASPP layer are then fed into a 3×3 convolutional layer with batch normalization.

Encoder. Since material properties are usually translationally invariant, material recognition methods need to capture an orderless measure encompassing some spatial repetition. Previous studies have shown that orderless pooling, like the Fisher Vector (FV), works better than order-sensitive pooling in material recognition [156]. The CNN combined with orderless pooling encoders, such as BOW, FV, and Vector of Locally Aggregated Descriptors (VLAD), has been demonstrated to be effective in material classification [156]. The proposed encoder module consists of a texture encoding network (TEN) [25] and local spatial pooling (LSP).

The TEN is used to build dictionary learning and feature encoding into a single CNN layer. The TEN can encode CNN features in an orderless manner such as FV and VLAD using a residual layer. The input of the TEN component is the output from the ASPP module with the shape of C×H×W, where C is the number of channels and H×W is the size of the feature map. The feature map is formed as a C-dimensional input features $X = \{x_1, \dots x_m\}$, where m is the total number of features given by H×W. The TEN layer learns an inherent codebook $C = \{c_1, \dots c_k\}$ and smoothing factors $S = \{s_1, \dots s_k\}$. The definition of residual encoding vector for codeword c_j is given in Eq. (1), where r_{ij} is the residual vector calculated as $r_{ij} = x_i - c_j$. The TEN layer aggregates CNN features into residual encoding vectors $E = \{e_1, \dots e_k\}$. Note that, increasing the number of codewords has the potential to capture more detailed texture information.

$$e_j = \sum_i^k \frac{e^{-s_j\left\|r_{ij}\right\|^2}}{\sum_{n=1}^m -s_n\left\|r_{ij}\right\|^2} r_{ij} \qquad (1)$$

To capture local spatial information, the outputs from the ASPP layer are also fed into a 3×3 convolutional layer with stride 2, followed by a batch normalization operation to standardize the outputs. A two-dimensional adaptive average pooling is then applied over outputs from the batch normalization. A fully connected layer is finally applied to reduce the feature dimension. The bilinear model is used to fuse outputs from the TEN and pooling layer by multiplying their feature maps using the outer product. The outer product captures pairwise correlations between the material texture encodings and spatial observation structures. The bilinear function is given by Eq. (2), where m^d is the output from the TEN network, g^d is the output from the LSP layer, $f^{d\times d}$ is the output of the bilinear model.

$$f^{d\times d} = \sum_{i=1}^d \sum_{j=1}^d m^d g^d \qquad (2)$$

Table 5.1 displays the detailed architecture of the proposed network. The input image size is 224x224x3. The outputs dimension from the Efficientnet_B4 module is 448x7x7, which is fed into the ASPP modules. The dimension of the outputs from the ASPP1, ASPP2, and ASPP3 are 512x14x14, 1024x7x7, and 1024x7x7, respectively. The channels of the outputs from the ASPP modules are reduced to 348 using a convolutional layer.

Table 5.1. Architecture of the proposed network

Module	Operator	Kernel size	Output size	#Channel	#Layer
EfficientNet-B4	Conv+BN+Swish	3X3	224X224	48	1
	MBConv1	3X3	112X112	24	2
	MBConv6	3X3	112X112	32	4
	MBConv6	5X5	56X56	56	4
	MBConv6	3X3	28X28	112	6
	MBConv6	5X5	14 X14	160	6
	MBConv6	5X5	7X7	272	8
	MBConv6	3X3	7X7	448	2
ASPP1	Atrous Conv1	1X1	14 X14	128	1
	Atrous Conv2	3X3	14 X14	128	1
	Atrous Conv3	3X3	14 X14	128	1
	Pooling+Conv+BN +ReLU	1X1	14 X14	128	1
	Concatenate	-	14 X14	512	1
	Conv+BN	1X1	14 X14	348	1
ASPP2 & ASPP3	Atrous Conv1	1X1	7X7	256	1
	Atrous Conv2	3X3	7X7	256	1
	Atrous Conv3	3X3	7X7	256	1
	Pooling+Conv+BN +ReLU	1X1	7X7	256	1
	Concatenate	-	7X7	1024	1
	Conv+BN	1X1	7X7	348	1
Encoder1	TEN	-	1	3072	1
	FC	-	1	64	1
	Conv+BN	3X3	6X6	192	1
	Pooling	6X6	1X1	1	
	FC	-	1	64	1
	Bilinear mapping	-	1	4096	1

Table 5.1. Continued

Encoder2 & Encoder3	TEN	-	1	3072	1
	FC	-	1	64	1
	Conv+BN	3X3	3X3	384	1
	Pooling	3X3	1X1	192	1
	FC	-	1	64	1
	Bilinear mapping	-	1	4096	1
	Concatenate	-	1	12288	1
Classifier	FC	-	1	256	1
	Classification	-	1	9	1

The outputs are then passed to the Encoder modules, which consist of the TEN and LEP layers. There are 8 codewords for the TEN layers.

The feature dimension from the TEN is 1x3072, which is then fed into an FC layer to reduce the feature dimension to 1x64. The LSP first applies a convolutional and batch normalization operation, and then the average pooling operation is conducted. An FC layer is applied as a dimension reduction step for outputs from the pooling layer. The dimensions of outputs from the TEN and the LSP are both 1x64. The bilinear model is then used to fuse the outputs from these two modules together, with an output of 1x4096. The outputs from Encoder1, Encoder2, and Encoder3 are concatenated together as a 1x12288 feature vector. Note that L2 normalization is used for the outputs from the TEN layer and the bilinear model. Finally, a fully connected classifier is used to classify the image.

Experiment and results

Dataset

The training dataset is prepared using the Materials in Context Database (MINC), which is a large dataset collected from a variety of contexts [158]. The MINC contains 2,996,674 single-point clicks across 436,749 images, and each click is associated with one of 23 material classes. This study aims to recognize the materials needing disinfection in infrastructure facilities. Some materials like skin and sky in the MINC are not applicable and are discarded for this study. In total, we select 9 types of materials that are commonly seen in infrastructure facilities. These material classes are fabric, leather, paper, ceramic, glass, metal, plastic, polished stone, and wood. To train the CNN model, square image patch data was extracted from the original images. The patch center is defined as the click point, and the size of the patch is 362x362. In many cases, patch areas may go beyond the border of images. Out-of-image pixels were filled with RGB (0,0,0). Note that if the out-of-image pixels number greater than 262, the patch will be removed from the dataset. The patch counts for each material class are shown in Table 5.2.

Table 5.2. Sample counts for each material in train, validate and test sets

Material	**Train**	**Validate**	**test**
Fabric	299,929	21,270	36,254
Leather	62,372	4,480	7,313
Paper	17,797	1,242	2,173
Ceramic	21,644	1,571	2,747
Glass	153,492	10,958	17,910
Metal	137,998	9,897	15,850
Plastic	31,282	2,146	3,661
Polished stone	85,196	6,054	9,855
Wood	39,8575	28,610	47,094

The training, validation, and testing datasets are obtained from the provided train/validation/test splits, which include 1,208,285, 86,228, and 142,857 patches, respectively. Figure 5.2 presents example patches for each material.

Implementation details

The models were trained using the PyTorch backend [62] with Dual NVIDIA Quadro P5000. The Stochastic Gradient Descent (SGD) optimizer was used with a learning rate of 0.002. The learning rate is divided by 10 for every epoch. The batch size is 128, the weight decay is 0.0001, and the momentum is 0.9. The pretrained weights on ImageNet of the EfficientNet-B4 backbone were used. The model that achieved the best score on the validation dataset is saved and used to further evaluate on the test dataset. Following the precedent set by existing literature [25,26], the patches are first resized to 256x256. Training samples were augmented by taking random crops measuring 224x224 out of the total 256x256. Horizontal and vertical mirror flips are applied to improve the generalization capability of the network.

Results of material recognition

Figure 5.3 presents the confusion matrix of the material classifications on the validation and test datasets. The trained model achieves high overall accuracy on the validation and test sets, measuring 92.24% and 91.84%, respectively. However, since the validation and test datasets are both imbalanced, recall is a better metric to evaluate the model. Therefore, the confusion matrix is normalized so that diagonal values represent recall for each class. Recall is calculated as the ratio of correctly predicted positives to true positive elements. This is used to measure the model’s predictive accuracy for the positive class. The material wood and fabric achieve a high recall score of 95% on both validation and test sets. The glass and polished stone achieve the second-highest score at 92%. A high recall score indicates the predictive power of the trained model on these classes. Plastic has the lowest recall score, 71%, and 10% of plastics are falsely classified as metal. The Barnes-Hut t-SNE algorithm [60] is adopted to visualize CNN feature maps in 2D, which is a tool to visualize high-dimensional features.

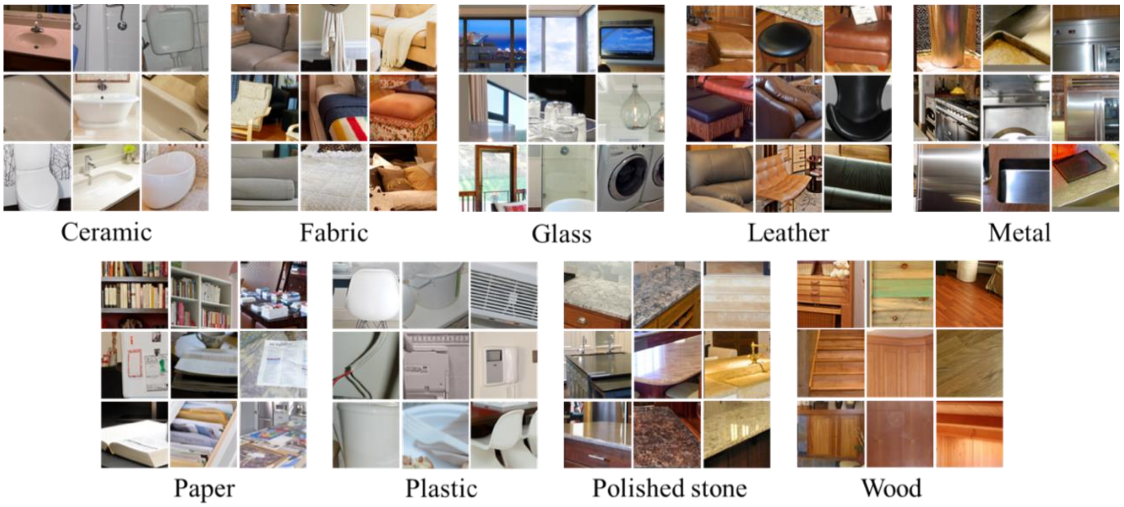
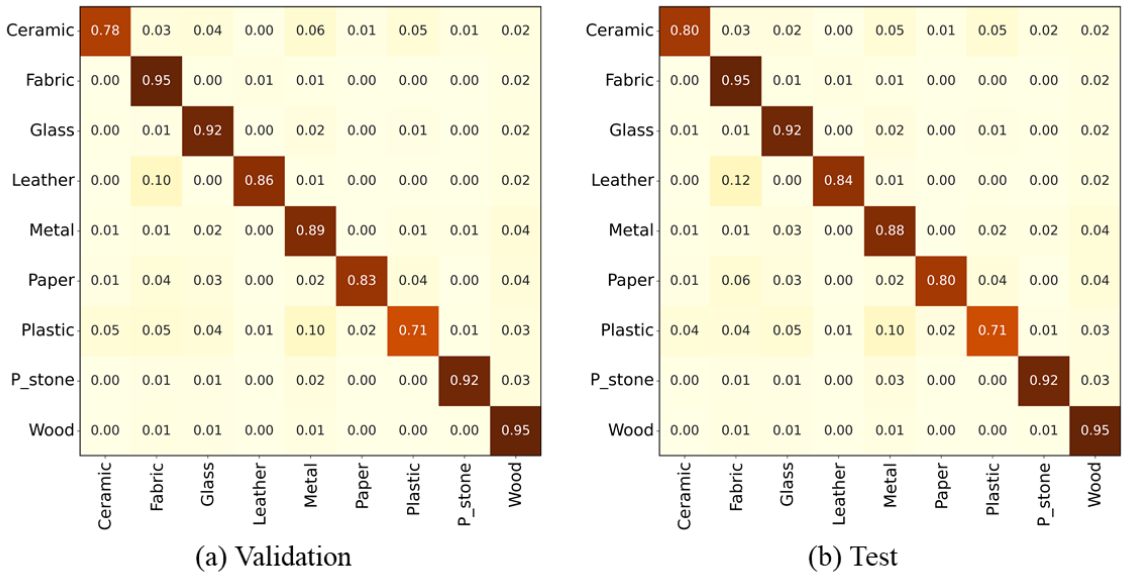


Figure 5.2. Example patches from all 9 material classes with context. Note that the patch center is the associated material (not necessarily the entire patch)



	Ceramic	Fabric	Glass	Leather	Metal	Paper	Plastic	P_stone	Wood
Ceramic	0.78	0.03	0.04	0.00	0.06	0.01	0.05	0.01	0.02
Fabric	0.00	0.95	0.00	0.01	0.01	0.00	0.00	0.00	0.02
Glass	0.00	0.01	0.92	0.00	0.02	0.00	0.01	0.00	0.02
Leather	0.00	0.10	0.00	0.86	0.01	0.00	0.00	0.00	0.02
Metal	0.01	0.01	0.02	0.00	0.89	0.00	0.01	0.01	0.04
Paper	0.01	0.04	0.03	0.00	0.02	0.83	0.04	0.00	0.04
Plastic	0.05	0.05	0.04	0.01	0.10	0.02	0.71	0.01	0.03
P_stone	0.00	0.01	0.01	0.00	0.02	0.00	0.00	0.92	0.03
Wood	0.00	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.95

(a) Validation

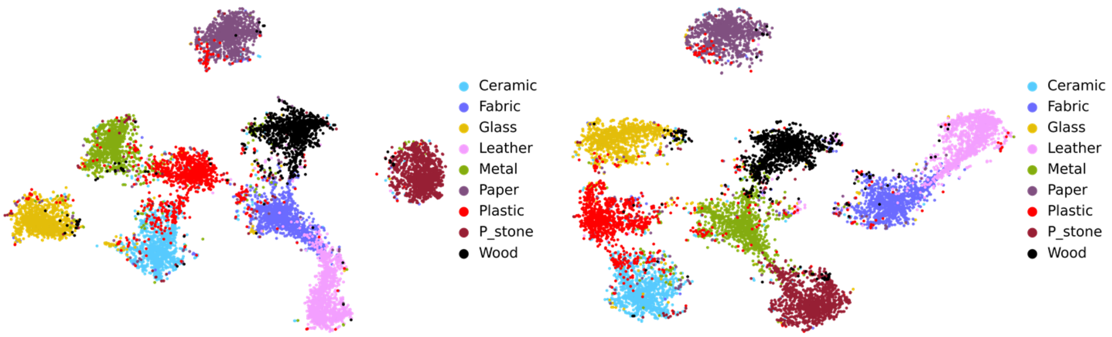
	Ceramic	Fabric	Glass	Leather	Metal	Paper	Plastic	P_stone	Wood
Ceramic	0.80	0.03	0.02	0.00	0.05	0.01	0.05	0.02	0.02
Fabric	0.00	0.95	0.01	0.01	0.01	0.00	0.00	0.00	0.02
Glass	0.01	0.01	0.92	0.00	0.02	0.00	0.01	0.00	0.02
Leather	0.00	0.12	0.00	0.84	0.01	0.00	0.00	0.00	0.02
Metal	0.01	0.01	0.03	0.00	0.88	0.00	0.02	0.02	0.04
Paper	0.01	0.06	0.03	0.00	0.02	0.80	0.04	0.00	0.04
Plastic	0.04	0.04	0.05	0.01	0.10	0.02	0.71	0.01	0.03
P_stone	0.00	0.01	0.01	0.00	0.03	0.00	0.00	0.92	0.03
Wood	0.00	0.01	0.01	0.00	0.01	0.00	0.00	0.01	0.95

(b) Test

Figure 5.3. The confusion matrix of model performance on (a) validate and (b) test dataset. Rows are actual classes and columns are predictions

The t-SNE algorithm assigns a high probability to similar objects and a low probability for dissimilar objects to construct a probability distribution over pairs of high-dimensional data. The t-SNE also constructs a probability distribution over pairs in the low-dimensional map. The location of the points in the map is determined by minimizing the Kullback-Leibler divergence between two distributions. In this study, Principal Component Analysis (PCA) [163] is first applied to the feature map to reduce its dimension to 50. The outputs of PCA are then fed into the t-SNE algorithm. The t-SNE algorithm converts the data into a 2D matrix. A relatively large perplexity value of 150 is used in the t-SNE to capture the global structure of the data [61].

Figure 5.4 shows a two-dimensional (2D) representation of the CNN feature map extracted from the layer before the last FC layer. The inputs for the t-SNE visualization are the material class and extracted CNN features. The CNN feature is 1×1 pixel size only, but with 256 channels. These 256 numbers are essentially all the features that the network extracted from the input image. For both validation and test sets, 1110 images are randomly selected from each class for visualization. Each point is associated with an image, and the distance between points approximates the original Euclidean distance in the high-dimensional features. If two image features are similar to each other, they will stay close in the resulting projection in the 2D map. The point color represents its related material classes. The t-SNE plots indicate that points from the same class are organized into clusters, which is an indicator of good differentiation between images of different classes using features before the last fully connected layer. The separated clusters indicate that the proposed network can understand the material data and its classes and is able to differentiate them. The results also highlight the relation between the clusters (i.e., connected clusters indicate there are some semantic relations between materials). For example, the material clusters of leather and fabric are connected to each other. These two materials are similar to each other compared to other materials in real life. This is because leather and fabric are both soft materials, and they are popular materials for furniture upholstery. Therefore, it is more challenging for the network to differentiate between these two materials.



(a) Validation

(b) Test

Figure 5.4. The Barnes-Hut t-SNE [33] visualization of CNN features on (a) validate; and (b) test dataset. Note: for Barnes-Hut t-SNE, 1110 images for each material class were randomly selected. The feature map before the classifier was extracted and used for t-SNE visualization

To further evaluate the model performance in the context of healthcare facilities, we manually labeled 1,173 patches consisting of 9 material classes. Hospital images were first downloaded from Google Images. These images contained different rooms at hospitals, such as the operating room, the consulting room, the intensive care unit, hallways, day rooms, the ward, and restrooms. We then collected clicks (i.e., single points) in images and assigned material labels to each point. To increase the accuracy of our labels, we only collected clicks with explicit materials. Table 5.3 presents sample counts for each material. Figure 5.5 presents example labeled patches for each material in the context of healthcare facilities.

Figure 5.6 presents the results of the model evaluation on the hospital material dataset. The overall accuracy of the trained model is 89.09%. The confusion matrix indicates that plastic has the smallest recall score at 60%. The recall of plastic is also the smallest on the MINC validation and test sets. This may be attributed to a relatively small number of plastic samples in the training set, and the fact that plastic features are similar to other materials like metal and ceramic. The top three false negatives for plastic are leather, ceramic, and metal. The recall score for leather is 87%, of which 13% of the leather samples were falsely classified as fabric. Other materials achieve high recall scores, demonstrating the efficiency of the trained model. The t-SNE plot shows the same materials are clustered together. The fabric and leather clusters are found to be close to each other. Figure 5.7 presents correct and incorrect predictions on the hospital material dataset with high confidence.

Comparison to state-of-the-art methods

We further evaluated the proposed network by comparing it to state-of-the-art methods on six material/texture datasets. These datasets are described as follows. (a) A subset of Material in Context Database (MINC) [158] datasets called MINC-2500 contains 23 material classes and 2,500 images per class. (b) Flicker Material Dataset (FMD) [164] contains a total of 1000 images that are equally distributed across 10 material categories, which has been used as an evaluation benchmark.

Table 5.3. Sample counts for each material in the hospital material dataset

Material	**Count**	**Material**	**Count**
Fabric	176	Metal	127
Leather	100	Plastic	100
Paper	100	Polished stone	100
Ceramic	135	Wood	142
Glass	193		

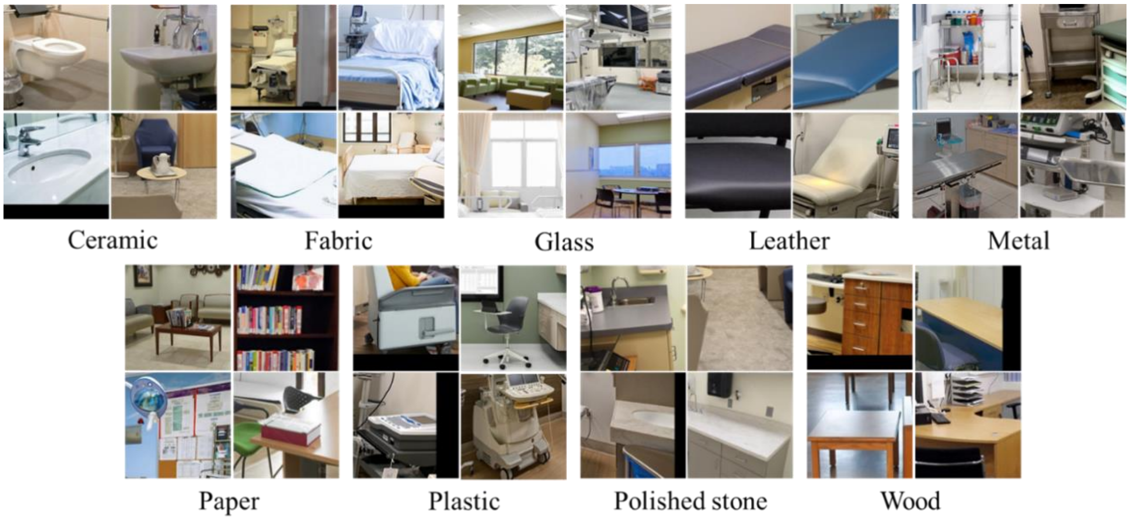
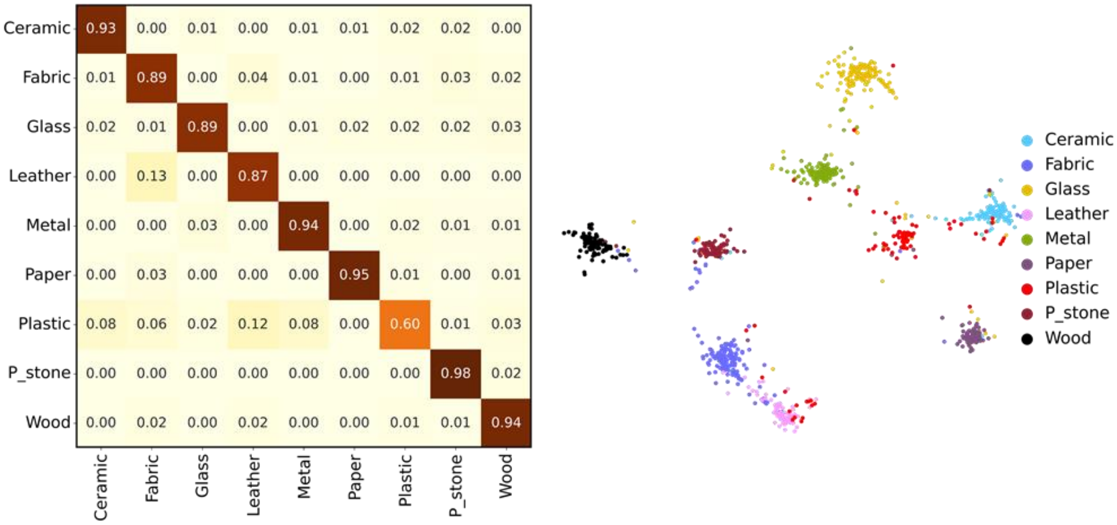


Figure 5.5. Example patches from all 9 material classes at hospitals. Note that the patch center is the associated material (not necessarily the entire patch)



	Ceramic	Fabric	Glass	Leather	Metal	Paper	Plastic	P_stone	Wood
Ceramic	0.93	0.00	0.01	0.00	0.01	0.01	0.02	0.02	0.00
Fabric	0.01	0.89	0.00	0.04	0.01	0.00	0.01	0.03	0.02
Glass	0.02	0.01	0.89	0.00	0.01	0.02	0.02	0.02	0.03
Leather	0.00	0.13	0.00	0.87	0.00	0.00	0.00	0.00	0.00
Metal	0.00	0.00	0.03	0.00	0.94	0.00	0.02	0.01	0.01
Paper	0.00	0.03	0.00	0.00	0.00	0.95	0.01	0.00	0.01
Plastic	0.08	0.06	0.02	0.12	0.08	0.00	0.60	0.01	0.03
P_stone	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.02
Wood	0.00	0.02	0.00	0.02	0.00	0.00	0.01	0.01	0.94

(a) Confusion matrix

(b) t-SNE plot

Figure 5.6. The confusion matrix of the model performance on the hospital material dataset. Rows are actual class and columns are predictions

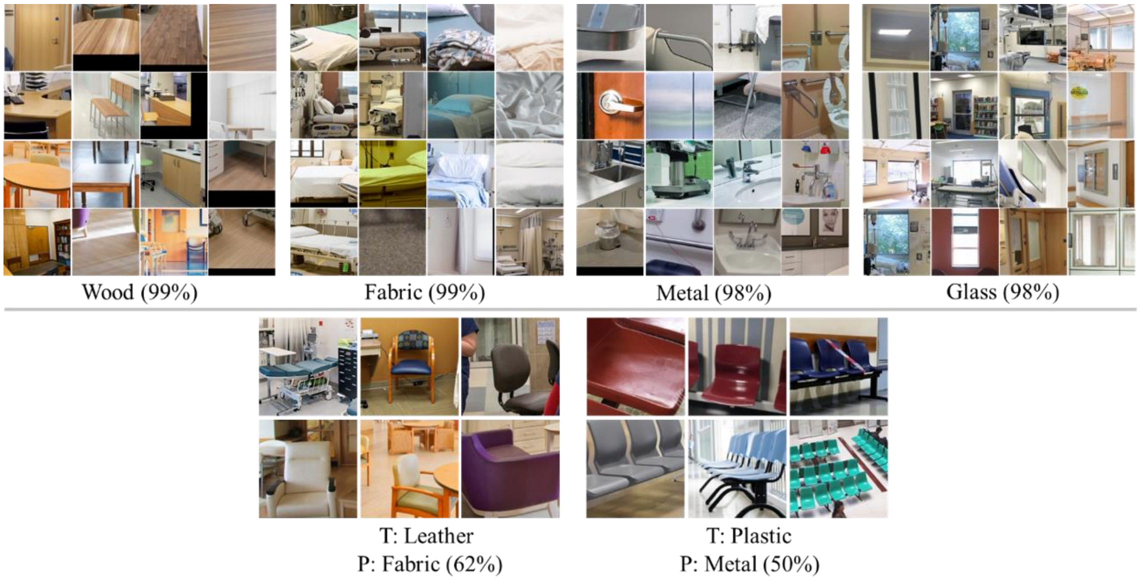


Figure 5.7. Samples with high confidence predictions in hospital material dataset. The first row is correct predictions, and the second row is incorrect predictions (T: actual material, P: predicted). The percentages shown are at least this confident

(c) Ground Terrain in Outdoor Scenes Dataset (GTOS) [26] contains more than 30,000 images with 40 material classes. (d) GTOS-Mobile [26] is a ground terrain dataset captured by mobile phones that consists of 93,945 training images and 6,066 testing images. (e) Describable Textures Database (DTD) [165] is a texture database consisting of 5640 images covering 47 classes. Each class consists of 120 images. (f) KTH-TIPS-2b (KTH) [166] is a material dataset, which is composed of 11 material classes with four samples per class. Each sample contains 108 images.

For a fair comparison with other methods [27,28,160], the evaluation is based on the provided train-test random splits for MINC-2500, DTD, GTOS-Mobile, and GTOS datasets. As for FMD, the dataset is randomly split into a train-test split in each run with 90 images per class used for training and 10 images used for testing. For KTH, three samples are randomly picked for training and the rest for testing in each run. The results are based on the 5-time run statistics for all the datasets. The learning rate is 0.01 and decays by a factor of 0.1 for every 10 epochs on MINC-2500. For FMD, GTOS, GTOS-Mobile, DTD, and KTH, the learning rate is set to 0.01. The network is trained using a momentum of 0.9, a weight decay of 0.0001. The training is finished in 30 epochs.

The performance of the proposed method is compared with the Fisher Vector CNN (FV-CN) with a VGG-VD backbone [156], Bilinear-CNN (B-CNN) [167], Locally-Transferred Fisher Vectors (LFV) [157], First and Second-Order information fusion Network (FASON) [168], Deep Texture Encoding Network (Deep-TEN) [25], Deep Encoding Pooling Network (DEP) [26], Deep Multiple-Attribute-Perceived Network (MAPNet) [160], Multi-level Texture Encoding and Representation Network (MuLTER) [169], Deep Structure-Revealed Network (DSRNet) [27], and Cross-Layer Aggregation of Statistical Self-similarity (CLASSNet) [28]. Table 5.4 presents the comparison results with these state-of-the-art methods. It should be noted that data distribution is different across the six datasets, which leads to varied performance. In addition, the dataset size and the material categories are also different among different datasets, which could also influence the model performance.

Table 5.4. Comparison to state-of-the-art methods on six material/textures dataset

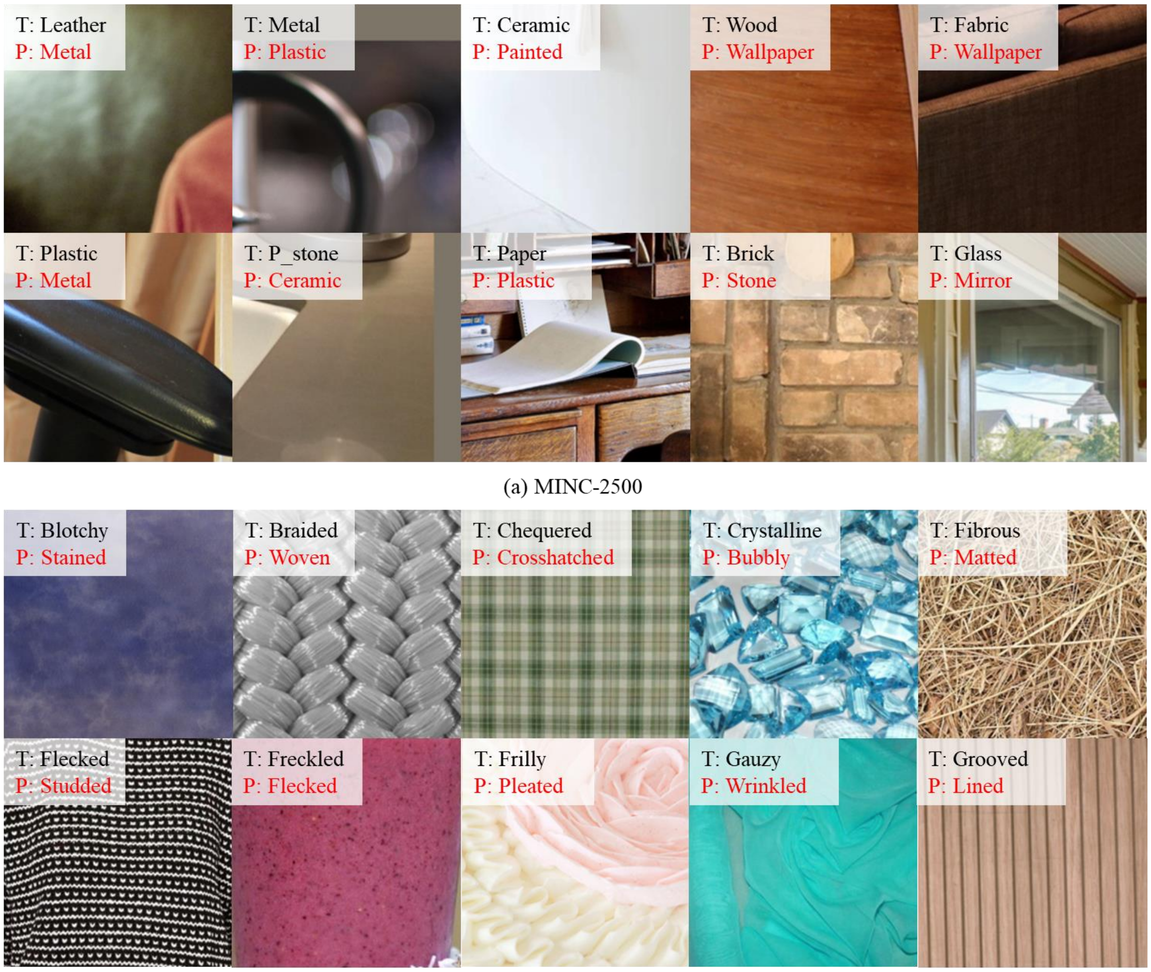
Method	MINC-2500		FMD		GTOS		GTOS-Mobile		DTD		KTH	
	mean	std	mean	std	mean	std	mean	std	mean	std	mean	std
FV-CNN (VGG-VD) (2015)	-	-	79.8	1.8	77.1	-	-	-	72.3	1.0	75.4	1.5
B-CNN (2016)	-	-	77.8	1.9	-	-	75.43	-	69.6	0.7	75.1	2.8
LFV (2017)	-	-	82.1	1.9	-	-	-	-	73.8	1.0	82.6	2.6
FASON (2017)	-	-	-	-	-	-	-	-	72.3	0.6	76.5	2.3
Deep-TEN (2017)	81.3	-	80.2	0.9	84.5	2.9	76.12	-	69.6	-	-	-
DEP (2018)	82.0	-	-	-	-	-	-	-	73.2	-	-	-
MAPNet (2019)	-	-	85.2	0.7	84.7	2.2	86.6	1.5	76.1	0.6	84.5	1.3
MuLTER (2019)	82.2	-	-	-	-	-	78.2	-	-	-	-	-
DSRNet (2020)	-	-	86.0	0.8	85.3	2.0	87.0	1.5	77.6	0.6	85.9	1.3
CLASSNet (2021)	84.0	0.6	86.2	0.9	85.6	2.2	85.7	1.4	74.0	0.5	87.7	1.3
Ours	**85.9**	0.4	**88.4**	1.8	**86.1**	0.5	**88.4**	0.6	**77.9**	0.4	**87.9**	2.4

Our method has the highest accuracy on the six datasets compared to the other texture/material recognition methods. Specifically, our method showed an improvement of 1.9%/2.2%/0.6%/2.7%/0.3%/0.2% in mean accuracy on MINC-2500, FMD, GTOS, GTOS-Mobile, DTD, and KTH compared to state-of-the-art methods, respectively. The comparison with the state-of-the-art methods demonstrates the robustness and accuracy of our methods for the task of material recognition.

The method performance is further compared to the Deep Encoding Pooling Network (DEP), which is the newest model available for our testing. This comparison aims to understand what kind of image features lead to misclassification by DEP, while our method can correctly recognize. The MINC-2500 and DTD are selected for our comparison. Figure 5.8 shows some example images that are correctly classified by our method and misclassified by the DEP. There are two major challenging in the material/texture classification for these example images. First, some of these images are smooth and featureless. Second, some images show spatially invariant features. Our proposed method can capture both the low-level texture and color information and the high-level semantic information, thus benefitting images with insignificant features. Furthermore, ASPP and the orderless encoder allow the network to learn the spatial repetitive features of material and textures.

Ablation study

To evaluate the proposed network, we study three components - the multi-level feature integration (ML), ASPP, and encoder (EC) - and summarize their effects on the model performance. The baseline model is generated by removing the integration of the three components, which becomes an EfficientNet-B4 network. The experiment is designed as follows. The effectiveness of each ML, ASPP, and EC component are evaluated by individually integrating them into the baseline model. A combination of either of the two components is also evaluated based on the accuracy metric. The performance is evaluated on the DTD and FMD. The results are listed in Table 5.5 for comparison. The results reported in the table represent the accuracy in the form of “mean ± st.d%”.



(a) MINC-2500

(b) DTD

Figure 5.8. Example images in MINC-2500 and DTD that are misclassified by DEP while correctly classified by our method. T is the true class; P is the predicted class by DEP

Table 5.5. Ablation study on DTD and FMD. ‘ML’ is multi-level feature integration. ‘EC’ is the encoder

Model	ML	ASPP	EC	DTD	FMD
Baseline				72.7±0.5	80.3±1.5
	√			75.9±0.3	86.3±1.5
		√		73.8±0.3	83.3±2.1
			√	74.3±0.4	85.3±1.5
	√	√		76.5±0.3	87.0±1.0
	√		√	77.6±0.7	87.6±1.5
		√	√	75.1±0.9	87.3±2.3
Proposed	√	√	√	**77.9**±0.4	**88.4**±1.8

The experiment results are detailed below.

Multi-level feature integration. In this part, we study the effects of multi-level feature integration, which is proposed to capture the low-level texture and color information and the high-level semantic information. The multi-level feature significantly improves the model performance on DTD and FMD with an improvement of 3.2%/6% (baseline→baseline+ML). We also conduct experiments on a combination of ML with either of the ASPP and EC components. In detail, ML+ASPP and ML+EC combinations are evaluated, and the results indicate an improvement compared with only ML integration on DTD and FMD. The ML has the highest improvement compared to ASPP and EC, which highlights the effectiveness of multi-level features in material representation.

ASPP. The ASPP component is used to capture multi-scale information, which can learn spatial repetitive features in material textures. Compared to the baseline, the integration of ASPP component improves the performance of the network by 1.1%/3% (baseline→baseline+ASPP). ASPP+EC is found to be better than baseline+ASPP, which has an improvement of 1.3%/4%.

Encoder. In this section, we evaluate the effectiveness of the encoder component, which is designed to capture both texture and local spatial information. The performance is improved by 1.6%/5% (baseline→baseline+EC) on DTD and FMD. This comparison indicates that the EC component can improve model performance. As mentioned above, a combination of EC with ML or ASPP can further improve performance.

Implementation

The material-aware disinfection robot is tested in a virtual environment built based on a patient room at a healthcare facility. The room is used to hospitalize patients with COVID-19. The disinfection robot equipped with UV light is used as an illustration. The disinfection robot first moves to potentially contaminated objects requiring disinfection.

The images captured by the camera are then fed into the CNN network to classify the surface materials. Figure 5.9 shows some example results on the disinfection of the overbed table, a door handle, a book, the seat of a chair, a sofa, and a vase in the patient room. The results indicate the proposed material recognition network can recognize the materials of the object surfaces needing disinfection.

Discussions

Robustness of material recognition method

Our proposed robotic disinfection system was successfully implemented in a fully modeled hospital room. The proposed material recognition method achieved an overall accuracy of 89.09% on the dataset collected in the context of healthcare facilities. The processing time for material recognition is around 0.04 seconds for a single image. The processing speed can be enhanced by increasing the batch size in the inference. For example, by setting the inference batch size to 64, a processing time of 0.12 seconds is sufficient to predict all the images. The promising results of our method demonstrated its accuracy and efficiency to provide the material information for the disinfection robot. The potential influence of illumination conditions and the robustness of the proposed approach to illumination is discussed below. Illumination is related to lighting and weather conditions. Illumination variation is a significant influencing factor for the image classification task. In this experiment, the gamma correction method is adapted to change the illuminance of the image based on the Power-Law Transform function [170]. The image is darker when the gamma values are smaller than 1, and the image is lighter when the gamma values are greater than 1. Gamma values of 0.5, 1.5, 2, and 2.5 are investigated in the experiment. Figure 5.10 presents the material prediction results for wood, fabric, and leather surfaces under varied illumination conditions. The wood surface can be recognized under different illuminations, while the prediction confidence decreases with increasing gamma values. The fabric surface is recognized with high confidence under all investigated illumination conditions. The leather surface is misclassified as fabric when the gamma values are 2 and 2.5. The incorrect predictions stem from the following reasons.

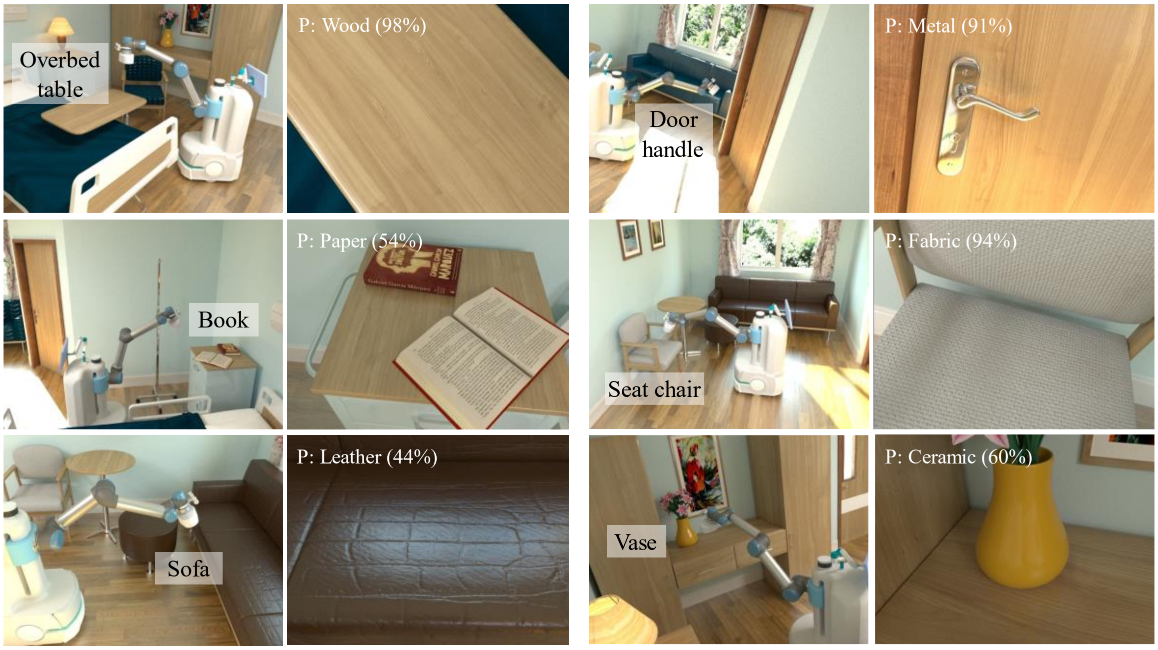


Figure 5.9. Example results of material classification on images captured by the robot. P: predicted material (confidence value in parentheses)

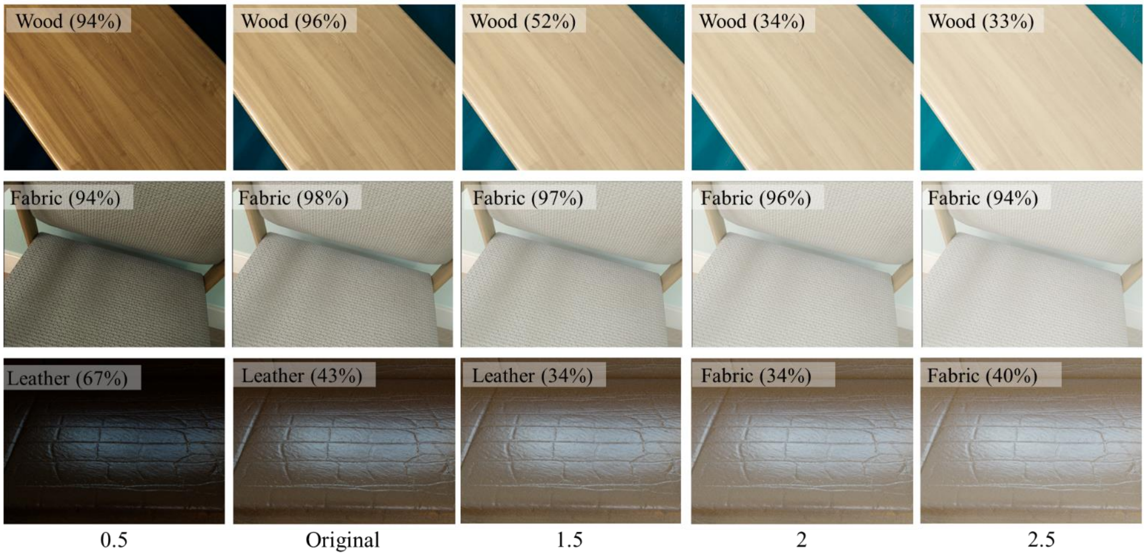


Figure 5.10. Material recognition performance under varied illumination conditions. Predicted material (confidence value in parentheses)

First, the prediction confidence is 43% for the original image, which is relatively low compared to wood and fabric. Second, leather surface features are close to fabric surface features, as indicated in Figure 5.10. The prediction can be improved with more leather material data in various illumination conditions. The illumination conditions have an impact on the performance of our network, particularly for bright images. The network can accurately predict materials in the image under different illumination conditions when the prediction confidence is high for original images. For prediction with low confidence (e.g., leather), the network also works for slightly brighter and darker conditions. Therefore, our material recognition network can be viewed as robust and reliable regarding illumination variation.

Limitation and future studies

This study suffers from several limitations that deserve future studies. First, despite the overall high performance of the material recognition network, the performance on plastic surfaces achieved a lower accuracy compared to other material categories. This underperformance was caused by a relatively small number of plastic samples in the training dataset when compared to other materials. In addition, the overall accuracy of the material classification model is smaller for the hospital material dataset than the accuracy for the MINC validation and test sets. The relatively lower accuracy stems from a lack of surface materials collected at hospitals in the training dataset. In the future, more surface material data needs to be collected in the healthcare facilities to fine-tune the network, especially for materials with fewer samples. In addition, other sensory data, such as thermal and time-of-flight depth cameras, could be integrated into the deep learning network to create a more robust model. Second, our work primarily focuses on recognizing materials in context, which does not differentiate the interface between different materials. A semantic segmentation approach is needed to classify materials at a pixel level. However, the segmentation task requires pixel-level annotations, which is expensive and time-consuming. In future research, material segmentation would be an interesting area to explore when more data becomes available.

Conclusions

This study proposed a new computational process and deep learning-based material recognition network to classify object surface materials and to adapt disinfection modes and parameters to disinfect surfaces thoroughly and efficiently. The deep learning network integrated multi-level and multi-scale CNN features, as well as a texture encoder to achieve material recognition with high accuracy. The trained network was evaluated on MINC validation and test dataset, and the results achieved an accuracy of 92.24% and 91.84%, respectively. The network achieved an accuracy of 89.09% on a small material dataset containing 1,173 samples collected in the context of healthcare facilities. Furthermore, the proposed material recognition network achieved state-of-the-art results compared to other texture/material recognition methods.

CHAPTER SIX
ROBOTIC ARM DISINFECTION MOTION PLANNING

Introduction

Disinfection and cleaning are the most important practice to mitigate the spread of infectious pathogens in mass-gathering built environments such as healthcare facilities, commercial buildings, airports, and schools. Many facilities still rely on physical labor to carry out disinfection processes, such as using hydrogen peroxide and ultraviolet disinfection, which is time-consuming, labor-intensive, and poses an infection risk to the cleaning staff [5]. Furthermore, manual disinfection is influenced by human behavioral factors, and real-world practices are highly variable [6]. For instance, Rutala and Weber [8] found that 10-50% of surfaces are contaminated in the rooms of patients infected with C. difficile, MRSA, and VRE. However, 51% of surfaces in patient rooms are found to not be thoroughly cleaned or disinfected, which could lead to a 120% increase in infection probability for future occupants of the room. There is a critical need for intelligent robotic disinfection to reduce viral bioburdens on contaminated surfaces, and thus prevent fomite-mediated transmission of infectious pathogens.

The elevated concerns due to the COVID-19 pandemic have increased the adoption of robotic technology for infection control and environmental hygiene. The disinfection robot market was valued at $0.49 billion in 2020 and is expected to reach $3.31 billion by 2026, representing a compound annual growth rate (CAGR) of 36.4% [171]. Existing disinfection robots are implemented as roaming bases with cleaning sources, such as UV lights applied to coarse sanitation scenarios. This automated approach requires the absence of people within the rooms or buildings to be disinfected [152]. Given the high volume of patients in healthcare facilities needing treatments, for example, disinfection robots are challenging to deploy. Furthermore, current practice is focused on solving coverage planning for the disinfection robot with a UV light column mounted on a mobile robot, which cannot be deployed in the presence of humans due to the risks of dangerous UV exposure. Therefore, developing an area coverage path planning method for a disinfection robot to adapt to multiple environments is critical to ensuring efficient and comprehensive disinfection applications.

The category, affordance, and material of an object significantly impact pathogen colonization and transmission. For example, the seating surface of a chair has direct contact with a person's hip, the backrest with a person's back, and the armrests with human hands, each posing different implications for disinfection. An object with frequent human contact is referred to as "high-touch" and may be a reservoir for nosocomial pathogens transmitted directly or indirectly by the hands of healthcare workers. Furthermore, object surface materials could also significantly impact pathogen colonization and transmission, requiring alternate disinfection modes, parameters, and procedures to ensure complete and efficient disinfection. However, generating efficient disinfection plans based on an object's surface information remains a significant challenge for mobile manipulator robots.

This research computationally links the recognition of surface characteristics (i.e., affordance, type, and material) to standard robotic disinfection actions, completing the operational connection between robotic perception and its actions. The innovation presented here is the computational modeling of the interactions between the surfaces, pathogens, and disinfection modes and the parameters assigned to adapt appropriate robotic disinfection actions, which has not been achieved by previous studies and currently deployed systems. These methods could lead to an intelligent robotic disinfection paradigm that extends existing systems limited to roaming UV lights for coarse disinfection. Intelligent and precise robotic disinfection could also be implemented in critical infrastructure facilities, such as hospitals, airports, school buildings, and food processing plants, to improve environmental and public health.

Literature review

Robotic disinfection has long been treated as a solution to mitigating the spread of infectious diseases in infrastructure facilities, which has been an active research area in recent years. The disinfection mode of these disinfection robots can be characterized by “UVC light”, “wipe,” and “spray”.

A UV-disinfection robot offers a non-touch method, disinfecting surfaces from a distance using a UVC light. UVC light is an environmentally friendly disinfection method, as it does not leave any residues on surfaces. UV-disinfection robots are essentially mobile robots with UVC light columns mounted to their top. These robots are commonly integrated with a variety of sensors for navigation and object detection, such as cameras, LiDAR, and ultrasound. Gibson et al. [113] deployed Xenon UV-disinfection robots in a hematopoietic stem cell transplant unit for three months and found the rate of HAI decreased to less than one per quarter. In [172], UV-disinfection robots were deployed facility-wide for terminal disinfection of the rooms where hospitalized patients were infected with Clostridium difficile. Since the beginning of the COVID-19 pandemic, many UV-disinfection robots have been developed and tested. For example, the MIT UV robot consists of an Ava robotics’ mobile base and a customized UVC light fixture, and the robot’s disinfection capabilities were tested in a food bank [173]. The robot took around 30 minutes to cover 4,000-square-foot spaces with a speed of 0.22 mph. The power and the number of light columns could be customized based on the size of rooms and their disinfection requirements. However, since UVC lamps are powerful enough to cause harm to the skin and eyes, the rooms must be evacuated during disinfection. To overcome this limitation, McGinn et al. [174] developed a prototype UV-disinfection robot called the “Violet robot platform”. A UVC reflectance shield was added to enclose the UVC lamp, thus reflecting the radiation emitted behind the robot. The authors claim that the Violet robot has the potential to work safely alongside human cleaners. In Hu, et al. [161], a UVC light wand was mounted onto a robotic arm with a mobile base to navigate in the built environment and to disinfect potentially contaminated surfaces. The major drawback of the existing UV-disinfection robots is that they cannot perceive object surface materials and adapt their disinfection parameters, which may lead to the incomplete disinfection of high-risk surfaces.

Wiping with chemical disinfectants is the main disinfection method for decontamination of high-touch surfaces in infrastructure facilities. “Wipe” mode has also been integrated into robot systems to disinfect contaminated surfaces. Toyota developed a ceiling-mounted home robot to wipe surfaces with soft rubble mounted on the gripper [175]. The robot can travel on the ceiling to avoid the problems associated with navigating a cluttered floor. Ramalingam et al. [176] proposed a disinfection robot prototype to automate the disinfection of door handles in infrastructure facilities. The authors designed a deep learning model to detect door handles in the image and to calculate the doorhandle location. More recently, the Fraunhofer Institute for Manufacturing Engineering and Automation proposed a prototype robot, called DeKonBot, to disinfect contaminated surfaces such as bedrails, light switches, and elevator buttons [177]. The DeKonBot consists of a mobile base and a robotic arm that carries out a wipe disinfection mode. Material information is critical for the wipe-disinfection robot because wipe mode is not suitable for certain types of surfaces, such as fabric and paper. However, existing wipe-disinfection robots have not been developed with the capability to recognize the material of the surfaces that require disinfection.

The disinfectant spray is another important disinfection method that has been widely used in the COVID-19 pandemic. For example, a smart prefabricated sanitizing chamber was designed for COVID-19 to enable a uniform spraying of sanitizing fluid onto healthcare workers [178]. The spray mode has also been integrated with the robotic system. Zhao et al. [179] developed a smart disinfection robot system that sprays disinfectants in the operating theaters or the patients’ rooms in healthcare facilities. The developed system primarily focused on the integration of multiple technologies, such as the Internet of Things (IoT), SLAM, hand gesture recognition, and navigation. Thakar et al. [180] developed an area-coverage planning algorithm for a spray-based disinfection robot to compute a path for the nozzle to follow to completely disinfect surfaces. The remote operator needs to select the area to be disinfected and extract the corresponding point cloud for path calculation. These spray-based robots achieved good results in controlled experiments. However, the disinfection dosage is not adaptable based on the surface materials, which can lead to incomplete disinfection. Furthermore, spray-based disinfection could damage paper surfaces. Therefore, it is important for these spray-based robots to recognize surface materials and to adapt disinfection modes and parameters.

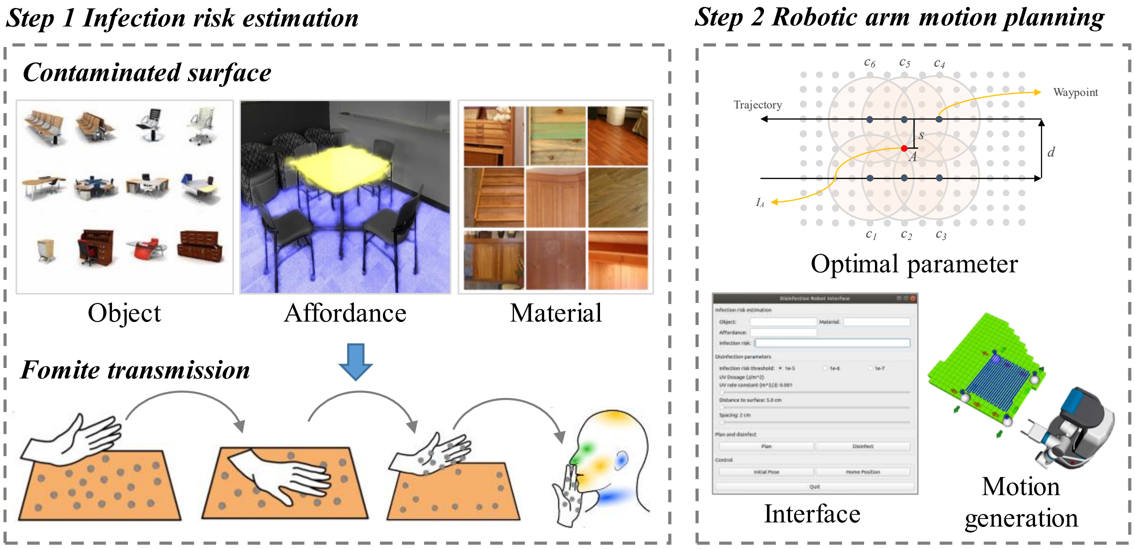
Methodology

Figure 6.1 overviews the proposed two-step approach. First, a fomite transmission model is designed to compute the infection risk from an object surface considering its characteristics, i.e., object type, affordance, and material. Second, the estimated infection risk is used to determine the $\log_{10}$ reduction needed to reduce the viral bioburden of the object surface below established target safety levels. This required $\log_{10}$ reduction in the bioburden next determines the required disinfection dosage for complete disinfection. The disinfection parameters are then optimized to ensure sufficient disinfection dosage is applied to the targeted surfaces. A semi-automatic robotic arm motion planning approach is proposed coupled with an interface to enable the operator to customize disinfection parameters.

Fomite transmission risk

The fomite transmission risk for surfaces is the core model for determining the disinfection practice needed to prevent the spread of pathogens. For high-risk contaminated surfaces, a high disinfection level is necessary for comprehensive disinfection. The fomite transmission risk for a contaminated surface is related to its object type, materials, and affordance. How these three factors affect the transmission of infectious disease is discussed below.

The frequency of human contact with objects is an important factor impacting the transmission of infectious diseases. In [74], a quantitative approach was conducted to identify high-touch surfaces in healthcare facilities.



Disinfection Robot Interface

Infection risk estimation

Object:
Material:
Affordance:
Infection risk:

Disinfection parameters

Infection risk threshold: ● 1e-5 ○ 1e-6 ○ 1e-7
UV Dosage (J/m^2)
UV rate constant (m^2/J): 0.001
Distance to surface: 5.0 cm
Spacing: 2 cm

Plan and disinfect

Plan | Disinfect

Control

Initial Pose | Home Position

Quit

Figure 6.1. Methodology overview

Human contacts with surfaces were observed and recorded during an 18-month period to estimate the mean frequency of contact for 28 surfaces located within intensive care units (ICUs). Figure 6.2 lists the statistics of this observed data. Three surfaces were considered high-touch, including bed rails, bed surfaces, and supply carts. Medium-touch surfaces included 11 surfaces with a mean of 1.75 contacts, and the remaining observed were considered low-touch surfaces. The risk of infections from touching high-touch objects is relatively higher compared to medium- and low-touch surfaces due to higher concentrations of bacteria and viruses, thus suggesting a requirement for complete disinfection. In this dissertation, the effect of the frequency of human contact is modeled as a weighting factor in the infection risk estimation. Specifically, the weighting factors applied for high-touch, medium-touch, and low-touch are 2, 1.5, and 1, respectively.

Object affordance infers how and with which parts of the human body people interact with object surfaces. Five object affordance labels are considered in this research, including walk, grasp, pull, place, and sit, which cover the most common interactions and actions with inanimate objects within constructed environments. For example, walkable areas of a building indicate where a robot can move to perform floor cleaning. Grasp and pull affordance represent object surfaces with direct interactions with human hands. The places where sitting occurs represent interactions with the human lower torso. Place affordance represents elevated surfaces where objects can be located. Object surfaces with human hand contact are assumed to pose higher transmission risks, which is considered reasonable because a person’s hand can likely have direct contact with human mucous, an essential route of fomite transmission. In particular, an object surface that includes direct contact with human hands has a higher bacteria and virus concentration under same touch frequency. Therefore, object affordance is modeled as a virus transfer probability from a human hand to a contaminated surface. The transfer probabilities for grasp and pull are defined as 1, 0.8 for place affordance, 0.6 for sit affordance, and 0.5 for walk affordance.

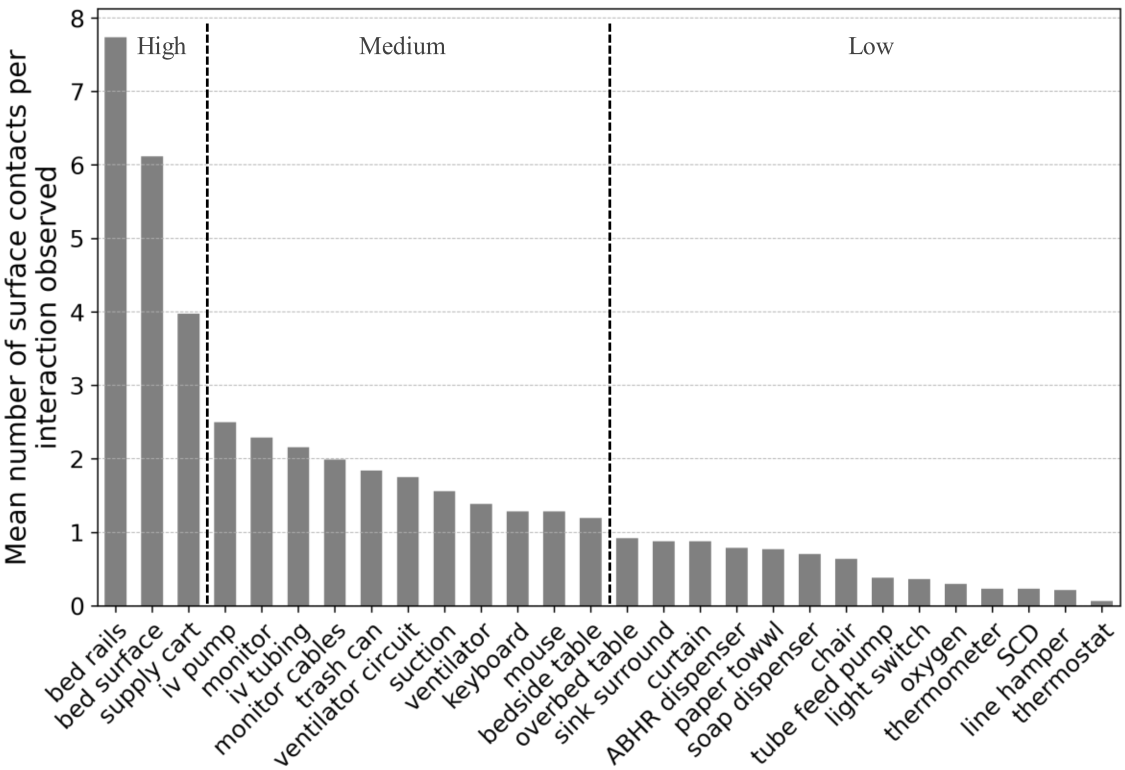


Figure 6.2. Mean frequency of healthcare worker contact for environmental surfaces in the intensive care unit

The object surface materials have significant impacts on pathogen colonization and transmission, and thus require different disinfection modes, parameters, and procedures to ensure complete and efficient disinfection. For example, a recent study by Chin et al. [151] suggested that certain pathogens, such as SARS-CoV-2, can stay infectious for as long as 7 days on metal and plastic surfaces, while SARS-CoV-2 may survive for only 2 days on fabric. Furthermore, the transfer efficiency or transmission rate of bacteria or viruses to hands from a surface differs between materials [152]. For example, the transfer efficiency of MS2 can reach 19.3% on glass surfaces but only 0.3% on fabric surfaces under a relative humidity of 15% to 32% [152]. Therefore, the effect of object surface materials has two significant impacts on bacteria or viruses colonized on a contaminated surface that are transmission efficiency and survival rate.

The concentration of bacteria and viruses undergoes an exponential decay in the survival period on different surfaces [181]. The concentration of bacteria and viruses at different times is given in Eq. (3), where C_0 is the initial concentration, t is the time in hours, w_o is the weight related to high-touch, medium-touch, and low-touch objects, and λ is the inactivation rate, P_A is the affordance-specific probability.

$$C_h = w_o \times P_A \times C_0 \times e^{-\lambda t} \tag{1}$$

The parameter λ can be estimated based on pathogen survivability on surfaces. In Chin, et al. [151], SARS-CoV-2 was found to not be detectable when the concentration was smaller than 100 TCID50/mL, which is around 0.02% of the initial concentration on wood surfaces. Therefore, 0.02% is used as the survival fraction for pathogens at the end of the survival periods. The parameter λ can be calculated as ln5000/st, where st is the survival time of the pathogen.

A stochastic-mechanistic model developed by Pitol and Julian [182] is adopted in this study to estimate the infection risk from contaminated surfaces with different materials. The model is built based on surface-to-hand-to-mucous contact. The transfer of a pathogen from surface-to-hand is first calculated in Eq. (2), where C_h is the bacteria and virus concentration on the surface at time h, *c_PFU* is the conversion factor from

Genome copies and Colony-forming unit to the infectious virus in PFU, *eff* is the pathogen recovery efficiency from surfaces, TE_{sh} is the transfer efficiency of the pathogen from surface to hand, β is $\log_{10}$ reduction in the number of bacteria and virus, and C_{hand} is the concentration on the hand.

$$C_{hand} = \frac{C_h}{eff} \times c\_PFU \times TE_{sh} \times\times 10^{-\beta} \tag{2}$$

The transfer of the pathogen from hand to mucous can be approximated by the concentration of pathogens on the hand and the transfer efficiency, which is defined in Eq. (3), where TE_{hm} is the transfer efficiency from the hand to mucous, and FSA represents the fractional surface area in contact with the mucous membranes, and D is the infectious dosage.

$$D = C_{hand} \times TE_{hm} \times FSA \tag{3}$$

The infectious dosage is then used to estimate the risk of infection using Eq. (4), where k represents the dose-response parameter.

$$P = 1 - e^{-kD} \tag{4}$$

Robotic arm motion generation

The $\log_{10}$ reduction of viral bioburden on surfaces represents the required level of disinfection to reduce transmission risks, with a higher value relating to a higher disinfection level. The robot must interpret defined disinfection parameters to apply appropriate actions for lowering an infection risk below a certain threshold. The required $\log_{10}$ reduction is an input parameter to calculate the disinfection parameters for the UV-based disinfection robot [161]. Bacteria and viruses decay when exposed to UVC light, which can be estimated as a first-order decay rate model [183] defined in Eq. (5), where i_t is the infection risk threshold, ρ is the UVC inactivation rate (m^2/J), and E_D is the UVC exposure dosage (J/m^2). The inactivation rate ρ for different microbial groups is found in [184]. For example, the inactivation rate for bacteria is 0.0864 m^2/J and 0.0580 m^2/J for viruses.

$$10^{\beta - i_t} = e^{-\rho \times E_D} \tag{5}$$

The UV light distribution shape from its source is assumed to be conical with a cone angle α, and the UV radiation intensity decreases with distance from the light source. A distance *D_n* is defined as the cut-off distance from the source, after which the effectiveness of the UV light is minimal. Another distance D_0 is specified to represent the minimum distance from an environmental surface that avoids collision with the UV light source. While a smaller distance could result in higher light intensity, generating a precise disinfection trajectory is difficult for the robotic arm. In addition, the measured depth from the onboard sensor likely incurs some error in real-world applications. Therefore, D_0 is essential to ensure safe disinfection processing and robotic maneuvers. The UV light is considered effective when the surface being disinfected is within the frustum of the cone, illustrated in Figure 6.3, and D_0 and D_n from the light source. The intensity of UV radiation on a surface perpendicular to the direction of the UV source at D_0 is modeled as a Gaussian distribution with a standard deviation of *σ*, with an intensity at the center of the surface represented as I_0.

Determining the intensity of UV radiation at any point within the volume of the frustum of the cone between D_0 and D_n is necessary. The intensity at D_0 shown in Figure 6.4 is defined in Eq. (6), where *r* is the radial distance from the center of the surface perpendicular to the direction of UV light emission.

$$I_R(D_0, r) = I_0 e^{-\frac{r^2}{2\sigma^2}} \tag{6}$$

The intensity of UV radiation at any distance D_z is given in Eq. (7).

$$I_R(D_z, r) = I_z e^{-\frac{r^2}{2\sigma_z^2}} \tag{7}$$

The light intensity at any cross-section perpendicular to the UV light emission direction is assumed to remain constant. Therefore, the total light intensity at D_0 is equal to the intensity at D_z, and is defined in Eq. (8).

$$\int_0^{R_0} I_0 2\pi r e^{-\frac{r^2}{2\sigma^2}} dr = \int_0^{R_z} I_z 2\pi r e^{-\frac{r^2}{2\sigma_z^2}} dr \tag{8}$$

Then, Eq. (8) can be reformulated as Eq. (9)

$$2\pi I_0 \sigma^2 \left(1 - e^{-\frac{R_0^2}{2\sigma^2}}\right) = 2\pi I_z \sigma_z^2 \left(1 - e^{-\frac{R_z^2}{2\sigma_z^2}}\right) \tag{9}$$

The radius R_z can be calculated in Eq. (10)

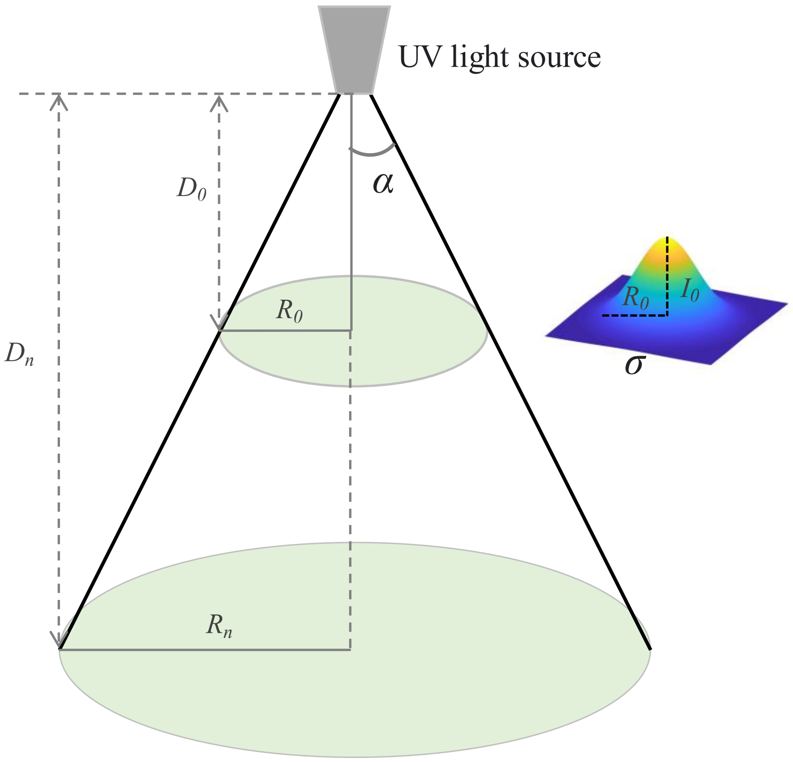


Figure 6.3. UV light distribution in a conical shape with angle α within the effective distance range D_0 and D_n. The intensity is modeled as a Gaussian distribution for the circular cross-section at the distance D_0

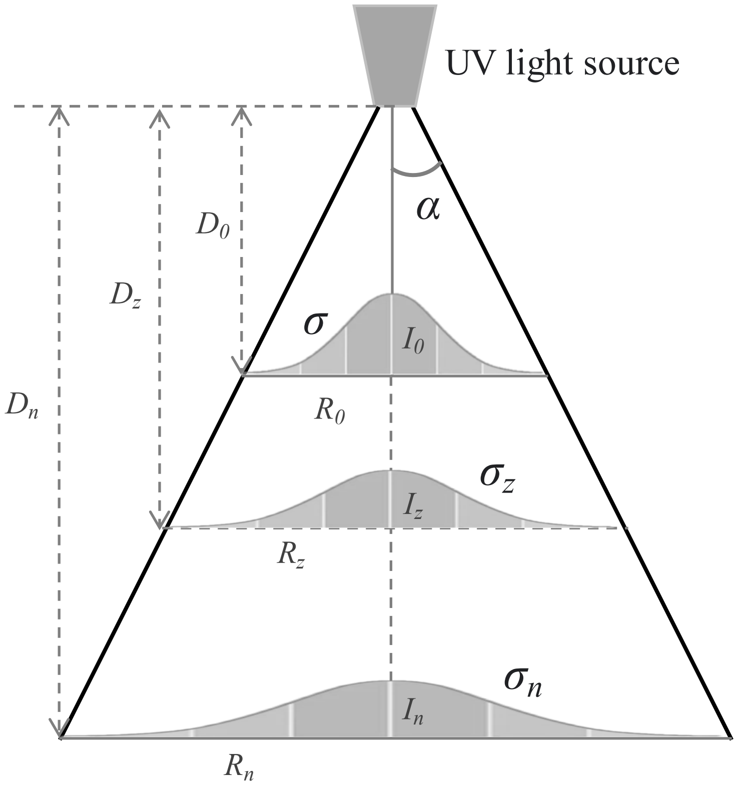


Figure 6.4. The intensity of UV radiation is modeled as a Gaussian distribution as the distance from the UV light source

$$R_z = R_0 \frac{D_z}{D_0} \tag{10}$$

The standard deviation is assumed to vary linearly with distance, so σ_z *and* I_z can then be estimated in Eq. (11).

$$\sigma_z = \sigma \frac{D_z}{D_0}; I_z = I_0 \frac{D_0^2}{D_z^2} \tag{11}$$

Therefore, Eq. (11) can be reformulated as Eq. (12)

$$I_R(D_z, r) = I_0 \frac{D_0^2}{D_z^2} e^{-\frac{r^2 D_0^2}{2\sigma^2 D_z^2}} \tag{12}$$

The UV dosage E_d received at any target surface point can now be estimated in Eq. (13), where E_t is the exposure time.

$$E_D = I_R(D_z, r) \times E_t \tag{13}$$

The exposure time at a surface point is related to the velocity of the robotic arm manipulator *V*. Specifically, smaller velocity results in a longer exposure time, i.e., a high UV dosage. The maximum velocity of the manipulator is set to V_{max}. The disinfection trajectory follows a lawn mower pattern to ensure a full coverage on the surface to be disinfected. The velocity of the robotic arm is configured to be the same during each execution. The minimum UV dosage E_{min} within the surface must be at least as high as the required UV dosage. Figure 6.5 shows two adjacent disinfection paths with waypoints. Point *A* is located between the waypoints for c_2 and c_5, which is assumed to be the point that receives the minimum UV dosage. The UV dosage I_A for this point at a given time is defined in Eq. (14), where *d* is the spacing of the trajectory, which must be smaller than R_z to ensure complete coverage of the selected area.

$$I_A = I_0 \frac{D_0^2}{D_z^2} e^{-\frac{s^2 D_0^2}{2\sigma^2 D_z^2}} + I_0 \frac{D_0^2}{D_z^2} e^{-\frac{(d-s)^2 D_0^2}{2\sigma^2 D_z^2}} \tag{14}$$

Based on $d(I_A)/ds = 0$, *s* is equal to *d*/2, where the minimum of I_A is obtained.

As indicated in Figure 6.5, from c_1 to c_2, point *A* is initially located outside and later inside the circle. Therefore, the intensity is zero initially and gradually increases as the manipulator moves forward. From c_2 to c_3, the intensity decreases as the manipulator moves forward.

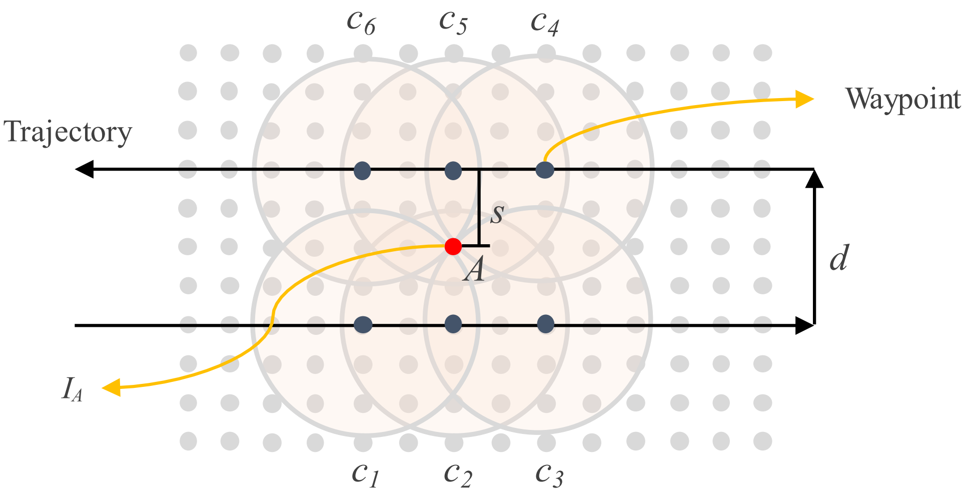


Figure 6.5. Disinfection trajectory with waypoints. The cycle means the effective range of UV light on the surface

The minimum UV dosage received by point *A* is estimated in Eq. (15).

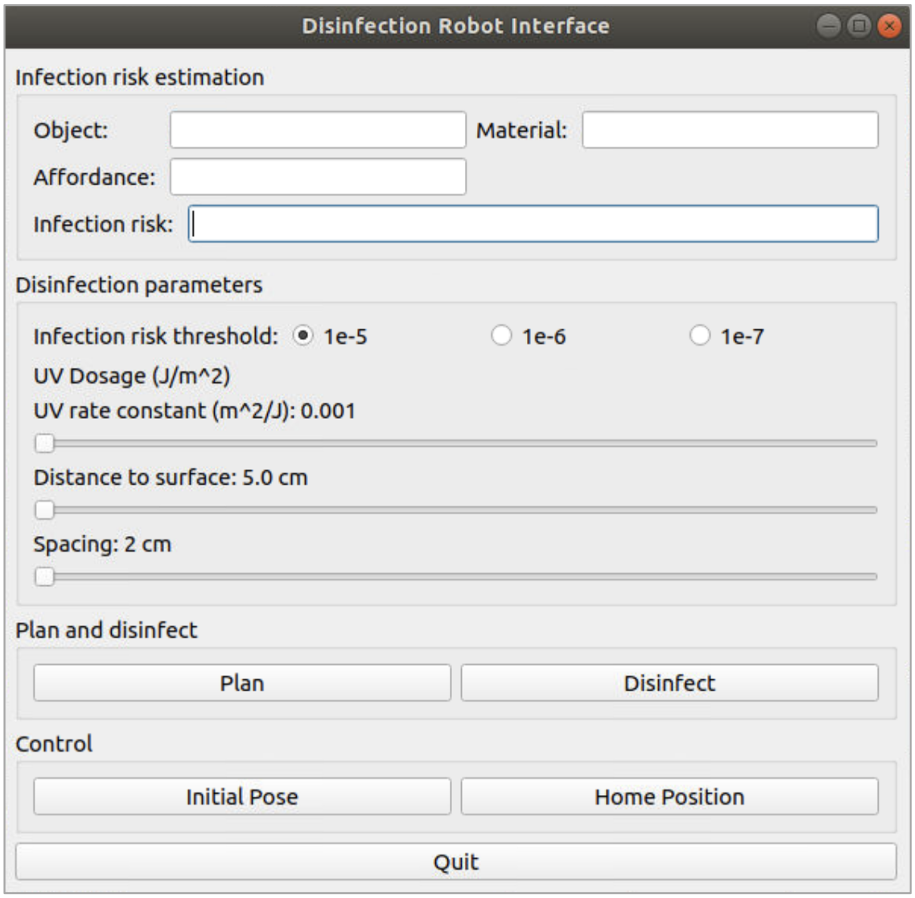
$$sum(I_A) = 4 \times \frac{\sqrt{R_z^2-(d/2)^2}}{V_{max}} \int_{d/2}^{R_z} I_0 \frac{D_0^2}{D_z^2} e^{-\frac{r^2 D_0^2}{2\sigma^2 D_z^2}} dr \tag{15}$$

The minimum UV dosage $sum(I_A)$ needs to be at least greater than the required UV dosage E_d. Dividing this minimum dosage by the required dosage results in the scaling velocity factor implemented by the robotic arm.

The parameters I_0, D_0, R_0, σ, α, and D_n used in the above model are UV light-specific parameters that could vary between UV light types and can be determined experimentally. An operator must specify other parameters through the interface shown in Figure 6.6, including the threshold of infection risk, UV rate constant, distance from the UV light source to the target surface, and spacing. The object surface details (i.e., object type, affordance, material) are observed using the onboard camera and used for the infection risk estimation. With these additionally specified parameters, the robotic arm computes its velocity to ensure the contaminated surfaces receive sufficient dosage.

The occupancy map of the surrounding environment is presented to the human operator in a Rviz interface shown in Figure 6.7. In this interface, the operator places markers onto the target object surfaces requiring disinfection. The disinfection trajectory is automatically generated based on the lawn mower pattern when these markers form a polygon. The spacings between the rows are also specified using this interface. The distance between waypoints is configured to be the same as the spacing between the rows because the velocity is expected to be constant in every implementation, and the distance between two waypoints is not an influential factor in effective disinfection.

Waypoints are defined as individual points along the path followed by the end effector. The disinfection trajectory can be generated using these waypoints follows four steps, as outlined in Figure 6.8. First, sampling points are linearly interpolated between the first and second waypoints. The number of these sampling points is the distance divided by the resolution.



Disinfection Robot Interface

Infection risk estimation

Object: Material:

Affordance:

Infection risk:

Disinfection parameters

Infection risk threshold: ● 1e-5 ○ 1e-6 ○ 1e-7

UV Dosage (J/m^2)

UV rate constant (m^2/J): 0.001

Distance to surface: 5.0 cm

Spacing: 2 cm

Plan and disinfect

Control

Figure 6.6. Disinfection robot control interface

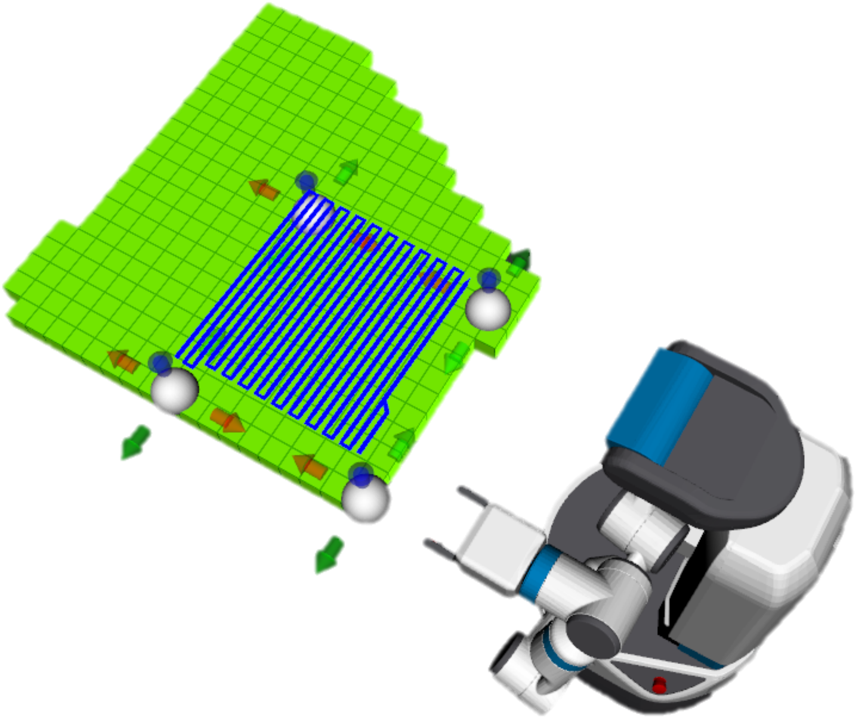


Figure 6.7. RViz interface with markers placed on the 3D map. The blue lines represent the disinfection trajectory with a certain distance to the surface

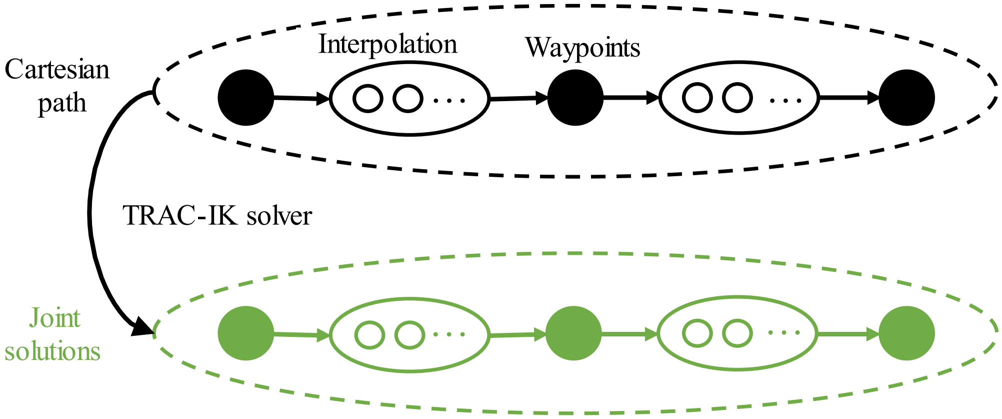


Figure 6.8. Flowchart to generate a trajectory from waypoints

Second, the TRAC-IK algorithm [185] is applied as inverse kinematics (IK) solver to calculate joint solutions. This algorithm is a numerical IK solver that combines both pseudoinverse Jacobian and Sequential Quadratic Programming-based nonlinear optimization solvers. The solver will stop and return the best solution once either of the two solvers finishes with a solution. Third, taking the current position as the starting position and the next waypoint as the goal, the first and second steps are repeated until all the waypoints are traversed. Finally, all the joint solutions are connected to generate a trajectory including velocity, acceleration, and duration.

Experiment and results

Results on fomite transmission risk

Experiment settings

Healthcare facilities contain different types of units that serve patients battling different illnesses, providing habitats and transmission pathways for various infectious pathogens. In this study, SARS-CoV-2 and Escherichia coli (E. coli) are selected as representative examples to illustrate how to transfer material information to disinfection practices. SARS-CoV-2 continues to lead to outbreaks of COVID-19 in healthcare facilities. E. coli has been identified as the major cause of urinary tract infections in healthcare facilities [186]. Patients infected with SARS-CoV-2 and E. coli are typically diagnosed in the pulmonology and urology departments and are hospitalized in different patient rooms. Therefore, the disinfection practices need to be adapted according to the prevalent infectious pathogens present in different types of hospital rooms.

Table 6.1 presents the survival time and transfer efficiency for SARS-CoV-2 and E. coli on different surfaces. As indicated, SARS-CoV-2 and E Coli. can generally survive longer on hard surfaces than on soft surfaces. For instance, SARS-CoV-2 can stay active for 7 days on metal and plastic but only 3 hours on paper [151]. Note that the survival time of SARS-CoV-2 is used for each material except for ceramic. The survivability of HCOV-229E, which is also a species of coronavirus, on ceramic is used instead.

Table 6.1. The survival time and transfer efficiency for SARS-CoV-2 and E. coli

Material	**SARS-CoV-2**		**E. coli**	
	Survival time	Transfer efficiency (%)	Survival time	Transfer efficiency (%)
Fabric	2 days [187]	0.73 [152]	4 – 56 days [188]	5.32 [152]
Leather	1 day [189]	7.00 [190]	-	-
Paper	3 hours [187]	0.55 [152]	1 – 96 h [191]	0.08 [152]
Ceramic	5 days [192]	24.15 [152]	14 days [193]	36.15 [152]
Glass	4 days [187]	43.30 [152]	1 – 14 days [188]	41.85 [152]
Metal	7 days [187]	21.95 [152]	14 – 60 days [188]	28.95 [152]
Plastic	7 days [187]	50.60 [152]	24 h – 300 days [188]	47.00 [152]
Polished stone	5 days [194]	20.10[152]	-	21.9 [152]
Wood	2 days [187]	31.50 [195]	2h – 28 days [188]	-

For fomite-to-hand transfer efficiency, data from MS2 coliphage are used due to the unavailability of SARS-CoV-2 transmission data. MS2 and SARS-CoV-2 are both single-stranded RNA viruses, which have similar transfer mechanisms from fomite to humans. Furthermore, MS2 has been used as a surrogate to facilitate the investigation of transmission and disinfection of SARS-CoV-2 [190]. Note that for the survival time of E. coli on paper and ceramic, Francisella tularensis and Klebsiella pneumoniae are used, as they are both gram-negative bacteria like E. coli.

Data collected at different surfaces indicated that the concentration of SARS-CoV-2 varied from 0.1 to 102.4 gc/cm^2 [196,197]. For E. coli, the concentration varied from 0.1 to 15.8 CFU/cm^2 [197,198] on contaminated surfaces. In this study, the initial concentration of SARS-CoV-2 and E. coli are assumed to be 100 gc/cm^2 and 10 CFU/cm^2, respectively. The virus is assumed to be colonized on a low-touch surface that has interactions with human hand. Table 6.2 gives the input parameters and their distributions used to estimate the required $\log_{10}$ reduction.

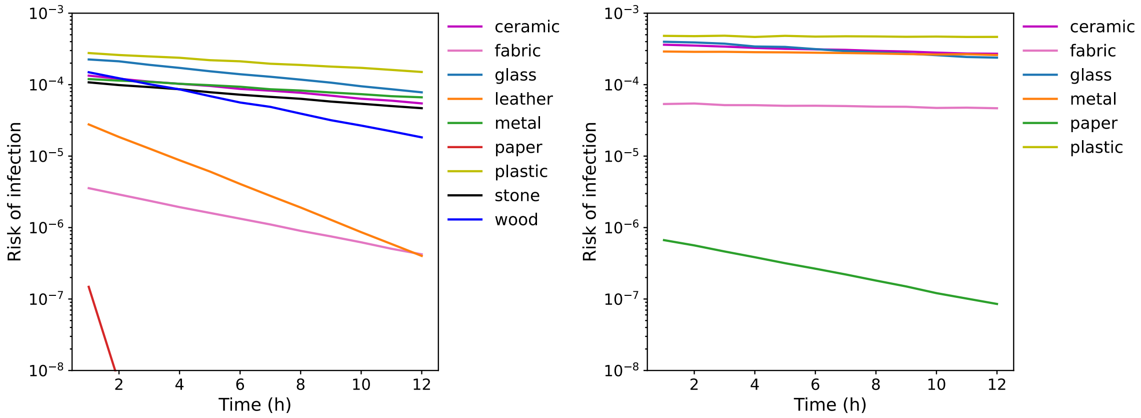
Analysis of results

The Monte Carlo simulation is used to estimate the infection risk by incorporating the input parameters' distributions. The model is simulated 50,000 times and the median risk values are reported. The survival time of E. coli is assumed to be a uniform distribution within the range. Figure 6.9 shows the estimated infection risk for different surfaces. The results indicate that infection risks of SARS-CoV-2 and E. coli for plastic, glass, metal, ceramic, and polished stone are higher than 10^{-4}. Soft surfaces, such as leather and plastic, have a lower infection risk compared to hard surfaces, such as plastic, glass, and metal. In addition, paper surfaces show a low infection risk, which is smaller than 10^{-6}.

In cleaning and disinfection practices, the disinfection dosage should be higher than needed to meet the disinfection requirements [199]. Therefore, the $\log_{10}$ reduction is rounded up to an integer to ensure the object surfaces are completely disinfected. The infection risk threshold is set to 10^{-6} for illustration.

Table 6.2. Input parameters

Parameter	**Unit**	**SARS-CoV-2**	**E. coli**
TE_{hm}	unitless	Normal (0.20, 0.06) [200]	
SA	cm^2	Uniform (4, 6) [201,202]	
k	PFU^{-1}	Triangle (0.00107, 0.00246, 0.00680) [203]	
eff	unitless	Normal (0.6, 0.266) [204]	
c_PFU	unitless	Uniform (0.01,0.001) [203]	Uniform (0.01,0.05) [205]



(a) (b)

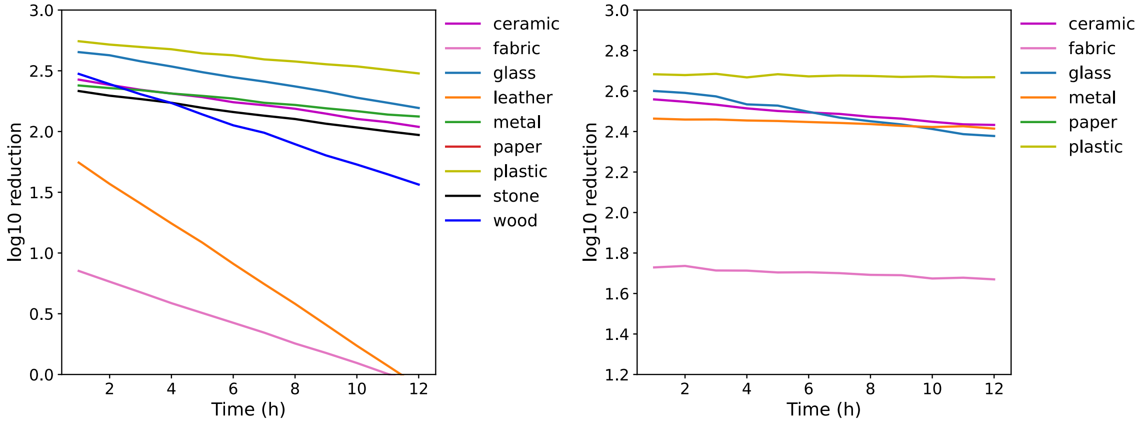
Figure 6.9. Risk of infection variation over time. (a) SARS-CoV-2; and (b) E. coli

Figure 6.10 shows the required $\log_{10}$ reduction to lower risk to below 10^{-6}. As indicated, the required $\log_{10}$ reduction is decreasing over time for both SARS-CoV-2 and E. coli. However, the decreasing trend for E. coli is much smaller than that of SARS-CoV-2 due to its long persistence period. A 2 $\log_{10}$ reduction would result in a risk of less than 10^{-6} for leather when the infection risk is relatively high for the first several hours. For fabric, $\log_{10}$ reductions of 1 and 2 are needed to achieve an infection risk of less than 10^{-6} for SARS-CoV-2 and E. coli, respectively. For ceramic, glass, metal, plastic, and polished stone, a 3 $\log_{10}$ reduction is needed to lower the infection risk below 10^{-6} for SARS-CoV-2 and E. coli. Note that paper doesn't require disinfection because of its infection risk of below 10^{-6} for both SARS-CoV-2 and E. coli. To be conservative, a 0.5 $\log_{10}$ reduction of bioburden is used for the paper surfaces.

The disinfection methods used in infrastructure facilities typically consist of UVC light, spray mode, and wipe mode. Each mode has its advantages and disadvantages and selecting the suitable mode depends upon the type and condition of the contaminated surfaces. Table 6.3 provides the disinfection mode and required $\log_{10}$ reduction to lower the risk below 10^{-6} for different surfaces. Note that $\log_{10}$ reductions for SARS-CoV-2 and E. coli in Table 6.3 were obtained within the first several hours when the risk was high. The wipe disinfection mode is not suitable for fabric and paper. For paper materials, the spray mode is also not applicable. For leather and hard surfaces, all disinfection methods are considered to be applicable modes. The disinfection level has different implementation methods for each disinfection mode. For UVC light, the disinfection level can be achieved by changing the distance, irradiance level, and exposure time. For wipe mode, the variables include wiping force, contact time, and disinfectant concentration. The spray mode can change the disinfectant concentration and amount to achieve different disinfection levels.

Uncertainty and sensitivity analysis

In this part, the sensitivity and the uncertainty of the model used to estimate the required $\log_{10}$ reduction are analyzed.



(a) (b)

Figure 6.10. $\log_{10}$ reduction needed to reduce infection risk below 10^{-6}. (a) SARS-CoV-2; and (b) E. coli

Table 6.3. Disinfection mode and level for different materials

Surface type	**Material**	**Disinfection mode**			**Log_{10} reduction**	
		Wipe	Spray	UVC light	SARS-CoV-2	E. coli
Soft	Fabric	-	√	√	1	2
	Leather	√	√	√	2	-
	Paper	-	-	√	0.5	0.5
Hard	Ceramic	√	√	√	3	3
	Glass	√	√	√	3	3
	Metal	√	√	√	3	3
	Plastic	√	√	√	3	3
	Polished stone	√	√	√	3	-
	Wood	√	√	√	3	-

Monte Carlo simulations are used to incorporate uncertainty and variability of the input parameters in the risk characterization. A ceramic surface contaminated with SARS-CoV-2 is selected as an illustration. Convergence is tested for the model by running 1000, 5,000, 10,000, 20,000, 50,000, and 100,000 simulations five times. The model estimation becomes stable after 50,000 runs, as indicated in Figure 6.11. Therefore, our study simulated a total of 50,000 runs for all the models.

The distribution of the disinfection dosage for different surfaces is further evaluated with SARS-CoV-2 persisting on the surface after 6 hours. The required $\log_{10}$ reduction distribution is shown in Figure 6.12. Note that the infection risk for paper is lower than 10^{-6}, which does not require disinfection to control its risk. For fabric and leather, a 2 $\log_{10}$ reduction is found to be sufficient to lower the infection risk. For other materials, the median $\log_{10}$ reduction is between 2 and 3. In some scenarios, the required $\log_{10}$ reduction could go beyond 3.

Spearman correlation coefficients are used to examine the relationship between the model input parameters and the disinfection dosage. The SARS-CoV-2 is selected for the sensitivity analysis. The transfer efficiency and survival time of SARS-CoV-2 are assumed to be uniformly distributed in the range given in Table 6.2. In addition, SARS-CoV-2 concentration after 6 hours is used for analysis. A total of 50,000 simulations are conducted. Figure 6.13 presents Spearman's correlation coefficients for the input parameters of the model. According to the sensitivity analysis, the model input parameters that mostly influence the required $\log_{10}$ reduction are the transfer efficiency between the surface and the hand and the survival time of the pathogen, which are both positively related to the disinfection dosage. These two parameters are material-specific parameters, which further confirms the importance of the material information for disinfection. The pathogen recovery efficiency is negatively correlated with the disinfection dosage. The correlation was positive for all other modeled parameters.

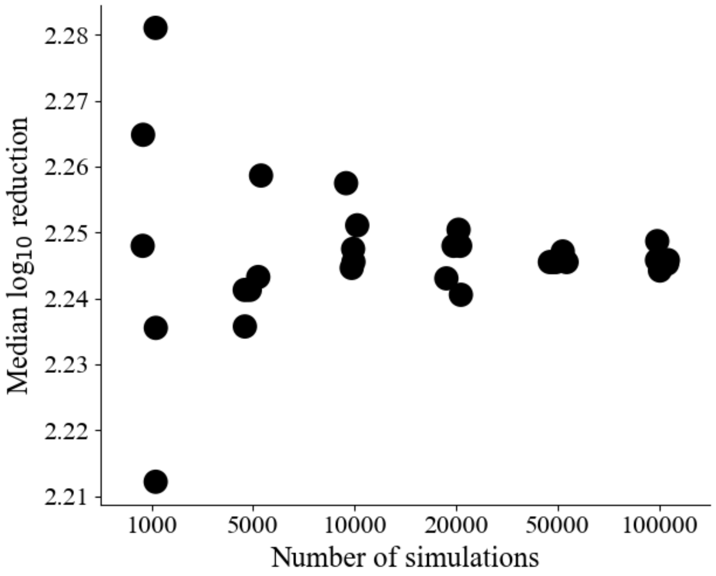


Figure 6.11. Median $\log_{10}$ reduction vs. number of Monte Carlo simulations. The results are based on five runs

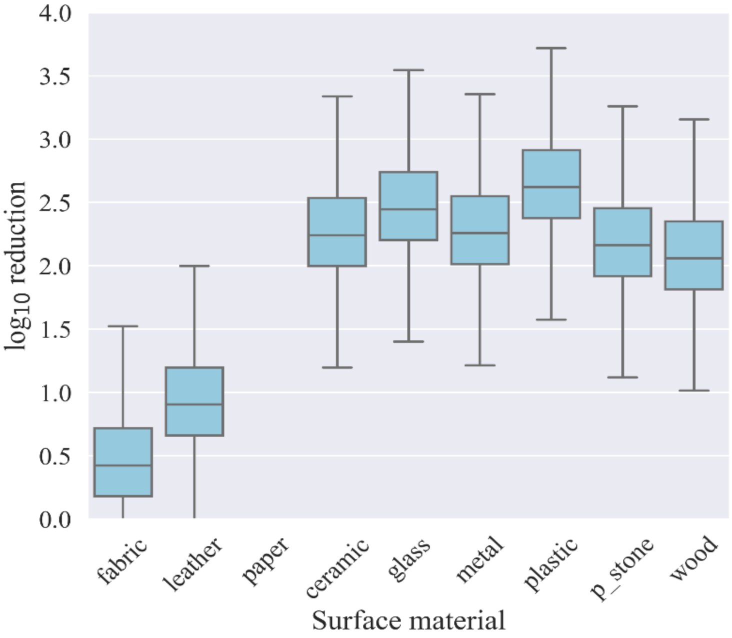


Figure 6.12. Boxplot of $\log_{10}$ reduction for different surfaces contaminated by SARS-CoV-2

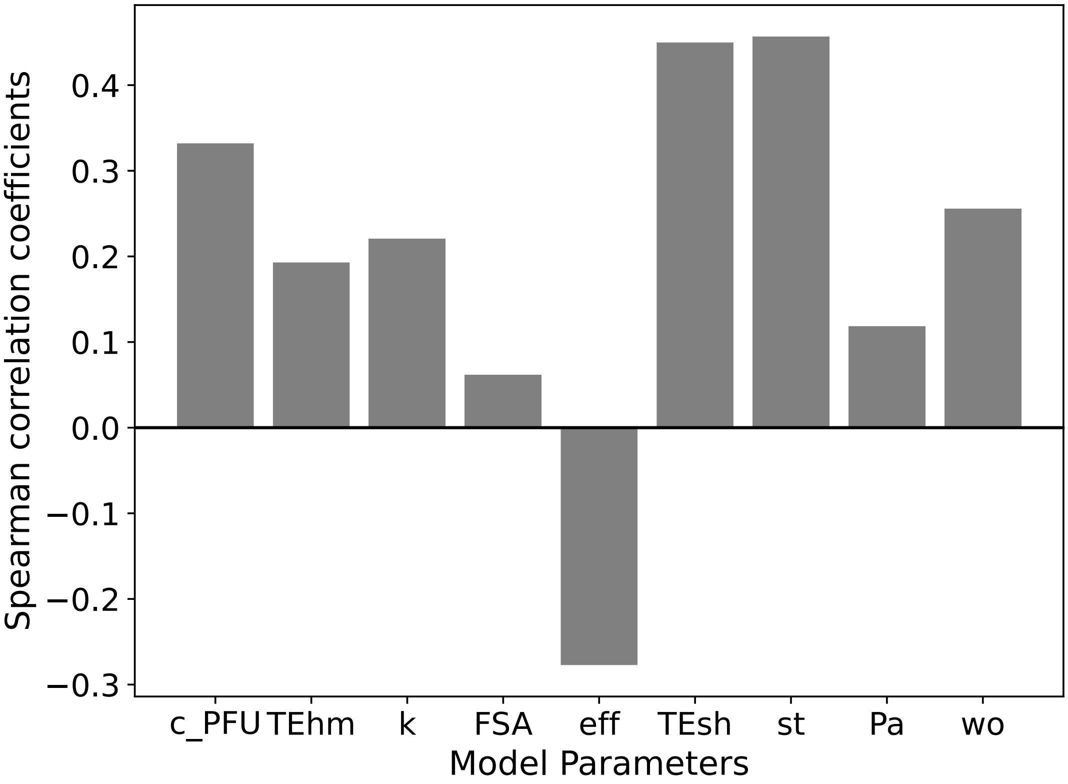


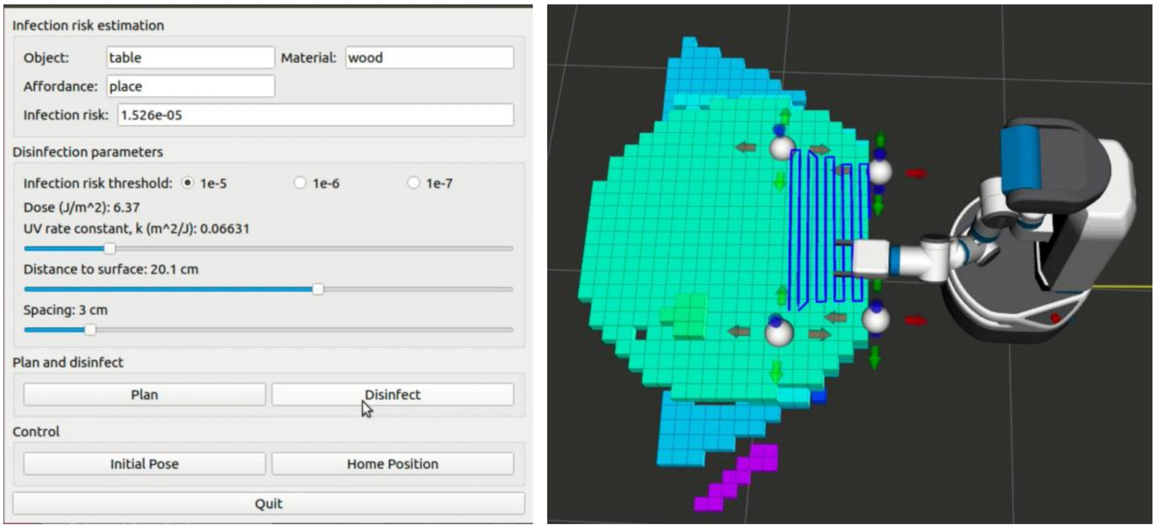
Figure 6.13. Spearman's correlation coefficients for the parameters used in estimating required disinfection dosage. Parameters are abbreviated as follows: c_PFU = conversion factor from Genome copies to the infectious virus in PFU; TEhm = = transfer efficiency of viruses from hand to mucous; k = dose-response parameter; FSA = fractional surface area; eff = pathogen recovery efficiency; TEsh = transfer efficiency of viruses from surface to hand; st = survival time of pathogen; Pa = affordance-specific transfer probability; wo = object touch frequency weight

Results on robotic arm motion planning

After navigating to the areas of potential contamination, a trajectory will be generated to perform disinfection. Figure 6.14 presents an example of a robotic disinfection process on a wood tabletop using the proposed method. The target areas for disinfection are first selected by manually adding four interactive markers through the interface, and the operator sets the disinfection parameters. For this illustration, the infection risk threshold is set to 10^{-5}, the UV rate constant is 0.06631 m^2/J, the distance to the target surface is 20.1 cm, and the spacing is 3 cm. The disinfection trajectory is then calculated based on these disinfection parameters. The manipulator moving speed is estimated as 0.50 m/s to ensure a sufficient dosage received by the contaminated surfaces. Through the same interface, the operator can customize disinfection parameters considering the immediate surrounding environment. The waypoint generator tool is interactive and enables the operator to select the areas needing disinfection.

Figure 6.15 shows the effect of the disinfection parameters on the scanning velocity of the robotic arm. These results indicate that the scanning velocity increases with an increasing distance to the surface. On the contrary, the velocity decreases with increasing robotic arm scanning spacing. While the velocity decreases with increased spacing, the total scanning distance also reduces. As such, the required time to disinfect the area could be reduced. The infection risk threshold is determined to be a significant influential factor on the velocity. In particular, the velocity decreases from 0.5 m/s to 0.08 m/s when the infection risk threshold decreases from 10^{-5} to 10^{-6}. A lower infection risk threshold corresponds to a higher level of disinfection.

In addition, a physical experiment was conducted using an AUBO-i5 robotic arm with a UV light attached as its end effector (as seen in Figure 6.16). The UV light in this scenario automatically turns on when it is close to the object's target surface requiring disinfection, and shuts off when the robot moves away from the target.



Infection risk estimation

Object: table

Material: wood

Affordance: place

Infection risk: 1.526e-05

Disinfection parameters

Infection risk threshold: ● 1e-5 ○ 1e-6 ○ 1e-7

Dose (J/m^2): 6.37

UV rate constant, k (m^2/J): 0.06631

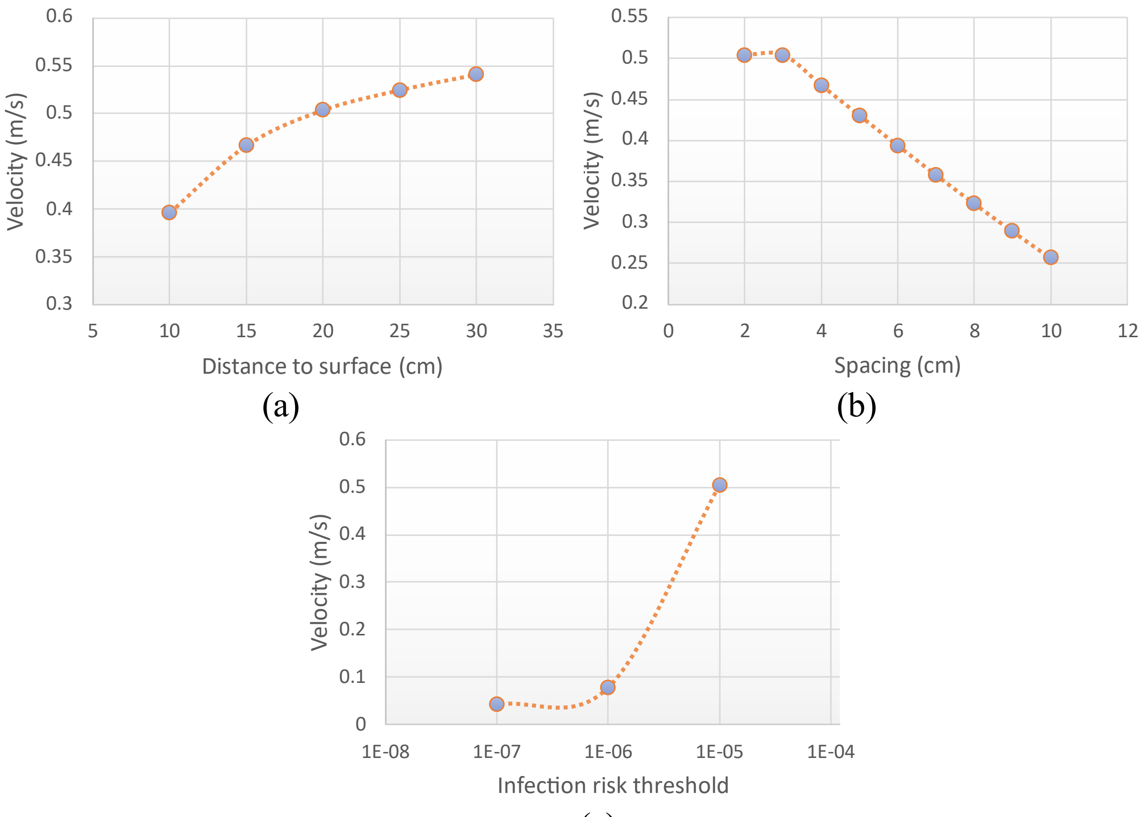
Distance to surface: 20.1 cm

Spacing: 3 cm

Plan and disinfect

Control

Figure 6.14. Results of robotic arm motion planning

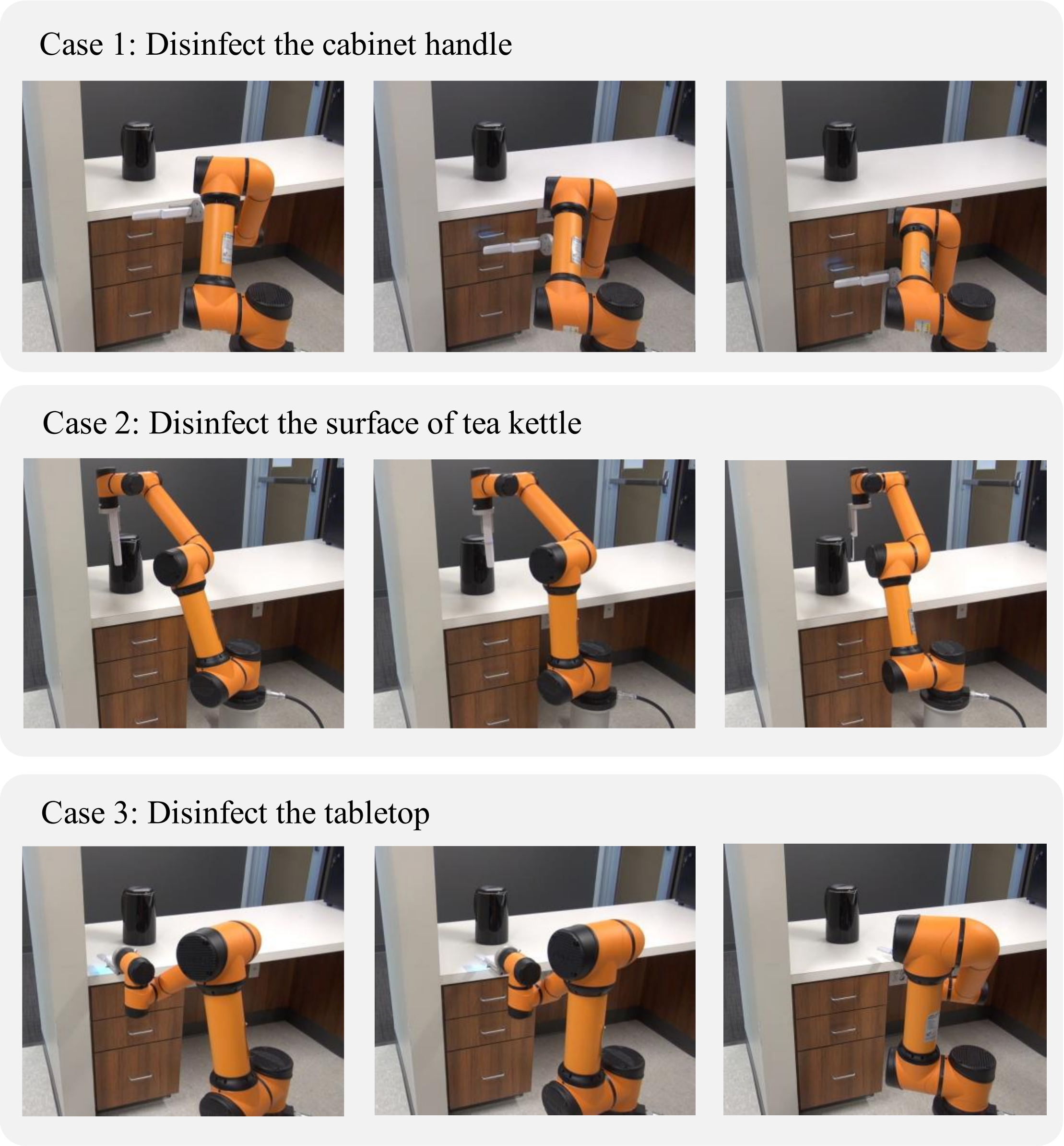


(a)

(b)

(c)

Figure 6.15. Effect of disinfection parameters on the scanning velocity of the robotic arm. (a) distance to surface; (b) spacing; and (c) infection risk threshold



Case 1: Disinfect the cabinet handle

Case 2: Disinfect the surface of tea kettle

Case 3: Disinfect the tabletop

Figure 6.16. Demonstration of robotic disinfection with an AUBO-i5 robotic arm.

Discussion

The robotic perception is computationally linked to robotic disinfection action in this chapter. Compared to manual disinfection, the adaptive system has the potential to not only ensure complete disinfection of contaminated surfaces, but also to improve the disinfection efficiency by increasing scanning velocity for low-risk surfaces. The proposed robot disinfection system has great potential to promote an intelligent robotic disinfection paradigm that goes well beyond existing systems that are perceived as roaming UV lights for coarse disinfection.

Many disinfection robots have already been deployed in infrastructure facilities like hospitals and schools. However, no disinfection robots to our knowledge possess the capabilities of the surrounding context, such as object type, object affordance, and object surface materials, to adapt disinfection modes and parameters, which largely restricts their disinfection efficiency. Our proposed framework can be integrated with existing disinfection robot platforms to improve their performance. For instance, in Hu et al. [161], the UV light wand was used as an end-effector of a robotic arm to disinfect contaminated surfaces in the built environment. With our newly developed framework, UV light parameters can be adapted based on the object surface characteristics with appropriate disinfection parameters for complete disinfection. As such, surfaces can be thoroughly disinfected and free of pathogens in sufficient numbers to prevent disease transmission.

There still exist some obstacles to the concrete operationalization of the proposed robotic disinfection system in the real world. First, controlled experiments need to be conducted to evaluate the effectiveness of the robot by measuring the surface pathogen concentration before and after disinfection. In addition, there still lacks evidence about how much contamination could lead to infection in humans. Second, there needs to be a validated, reproducible, and documented disinfection protocol for the robot. The development of such a protocol needs to have a close collaboration with the end-users, such as hospitals. As such, the robot design and protocol can be updated based on their feedbacks.

There remain several limitations that deserve future studies. First, this chapter investigates a particular pathogen on surfaces to demonstrate the computational feasibility and complete loop from robotic perception to robotic actions. However, many pathogens can cohabit on the same surfaces in healthcare facilities. In this case, the proposed method needs to be adapted for a multi-pathogen infectious disease system, which requires more research regarding pathogen dependency. Furthermore, for transmission risk, as types and frequencies of human activities and the diversity of environmental surfaces differs between settings, social and environmental contexts are of great importance in assessing the infection risk through fomite transmission. More advanced methods, such as the Environmental Infection Transmission System (EITS) modeling framework proposed in [206], could be explored to model more complex scenarios. Second, our current robotic arm implementation is conducted on planar surfaces, which are suitable for a variety of surfaces such as countertops, table surfaces, and seat bases. The planar assumption could lead to incomplete disinfection for areas below the surfaces. In future studies, how to generate a disinfection trajectory on curved surfaces is an interesting direction to be explored.

Conclusions

This study develops computational modeling of the interactions among surface, pathogens, and disinfection parameters to adapt disinfection actions. The fomite transmission model was adapted to estimate the infection risk for different surfaces and to quantitate the $\log_{10}$ reduction needed to reach the safety target levels. The results indicated that hard surfaces, such as plastic and metal, require a higher disinfection level, compared to soft surfaces, such as paper and fabric. The disinfection level was combined with the applicable mode to calculate disinfection parameters for the robot to implement. A semi-automatic robotic arm disinfection motion planning approach is proposed coupled with an interface to enable the operator to customize disinfection parameters. Both simulations and physical experiments were conducted to validate the proposed methods, which demonstrated the feasibility of intelligent robotic disinfection and highlighted the applicability in mass-gathering built environments.

CHAPTER SEVEN
SUMMARY

This study develops artificial intelligence (AI)-enabled robotic system that adapts to ambient environments, social context, and building dynamics for precise and complete disinfection, thus maintaining environment hygiene and health, and reducing unnecessary labor costs for cleaning and opportunity costs incurred from infections

Chapter One of the dissertation is devoted to providing a succinct overview of the key concepts, related studies, and knowledge gaps, as well as research objectives and contributions.

Chapter two of the dissertation contributes to developing a new multi-classifier decision fusion method to recognize human activity in healthcare facilities from still images. The method innovatively combines the output from CNN, ViT, and GNN to capture different features in images. Specifically, a graph classification network is designed to recognize human activity from scene graphs containing rich object and relationship features. Swin Transformer and ConvNeXt are trained to classify human activity from images. The decision fusion method integrates Dempster-Shafer theory and a weighted majority vote to combine the outputs from Swin Transformer, ConvNeXt, and GNN.

Chapter three of the dissertation contributes to creating a 3D indoor object mapping framework. The framework consists of object detection, SLAM, object coordinates estimation, and object clustering. Specifically, a novel real-time deep learning method is proposed based on YOLOv5 for object detection and classification. The object is detected as a bounding box in images and the point cloud corresponding to the bounding box is extracted. The point cloud is then filtered based on the range threshold and statistical method. The 3D coordinates of detected objects are estimated as the centroid of the filtered point cloud. The DBSCAN is utilized to cluster objects detected from different camera views.

Chapter four of the dissertation aims to detect and segment object affordance indicating potentially contaminated surfaces and project them into a 3D semantic map for precise disinfection. A deep learning network is designed based on DeepLabv3Plus to segment potentially contaminated surfaces in images. RTAB-Map SLAM method is adapted to track the pose of the camera while it is moving in the building. The 2D semantic label in the RGB image is projected onto the point cloud to generate 3D semantic information. The 3D semantic map is incrementally built combining the camera pose and semantic point cloud.

Chapter five of the dissertation aims to recognize material types of object surfaces requiring disinfection. A novel deep learning network is developed for material classification by integrating multi-level, multi-scale features. Specifically, low-level and high-level features are combined to capture semantic and texture information. The Atrous Spatial Pyramid Pooling (ASPP) module is used to extract multiscale features by resampling feature maps at multiple rates.

Chapter six of the dissertation develops a computational method to link object surface characteristics including object type, affordance, and material to robotic disinfection actions. In particular, a fomite transmission model is adapted in this dissertation to estimate the infection risk of a surface and the infection risk is used to estimate the required $\log_{10}$ reduction to achieve complete disinfection. The UV light disinfection is modeled to determine the optimal scanning velocity for complete disinfection. A robot control interface is developed to facilitate operators to customize disinfection parameters and select areas to be disinfected.

LIST OF REFERENCES

[1] E. Dong, H. Du, L. Gardner, An interactive web-based dashboard to track COVID-19 in real time, The Lancet Infectious Diseases. 20 (2020) pp. 533–534. https://doi.org/10.1016/S1473-3099(20)30120-1.

[2] N.H.L. Leung, Transmissibility and transmission of respiratory viruses, Nature Reviews Microbiology. (2021) pp. 1–18.

[3] M. Haque, M. Sartelli, J. McKimm, M.A. Bakar, Health care-associated infections – an overview, Infection and Drug Resistance. 11 (2018) pp. 2321. https://doi.org/10.2147/IDR.S177247.

[4] P.W. Stone, Economic burden of healthcare-associated infections: an American perspective, Expert Rev Pharmacoecon Outcomes Res. 9 (2009) pp. 417–422.

[5] H. Choi, P. Chatterjee, E. Lichtfouse, J.A. Martel, M. Hwang, C. Jinadatha, V.K. Sharma, Classical and alternative disinfection strategies to control the COVID-19 virus in healthcare facilities: a review, Environmental Chemistry Letters. (2021) pp. 1–7.

[6] M. Doll, M. Stevens, G. Bearman, Environmental cleaning and disinfection of patient areas, (2018). https://doi.org/10.1016/j.ijid.2017.10.014.

[7] M.M. Querido, L. Aguiar, P. Neves, C.C. Pereira, J.P. Teixeira, Self-disinfecting surfaces and infection control, Colloids and Surfaces B: Biointerfaces. 178 (2019) pp. 8–21.

[8] W.A. Rutala, D.J. Weber, Monitoring and improving the effectiveness of surface cleaning and disinfection, American Journal of Infection Control. 44 (2016) pp. e69–e76. https://doi.org/10.1016/J.AJIC.2015.10.039.

[9] A. Zemmar, A.M. Lozano, B.J. Nelson, The rise of robots in surgical environments during COVID-19, Nature Machine Intelligence. 2 (2020) pp. 566–572.

[10] L. Minh Dang, K. Min, H. Wang, M. Jalil Piran, C. Hee Lee, H. Moon, Sensor-based and vision-based human activity recognition: A comprehensive survey, Pattern Recognition. 108 (2020) pp. 107561. https://doi.org/10.1016/J.PATCOG.2020.107561.

[11] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Trans Pattern Anal Mach Intell. 32 (2009) pp. 1627–1645.

[12] M. Liang, X. Hu, Recurrent convolutional neural network for object recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015: pp. 3367–3375.

[13] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, Imagenet large scale visual recognition challenge, Int J Comput Vis. 115 (2015) pp. 211–252.

[14] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, Ssd: Single shot multibox detector, in: European Conference on Computer Vision, Springer, 2016: pp. 21–37.

[15] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: pp. 779–788.

[16] L. Zhang, X. Zhen, L. Shao, Learning object-to-class kernels for scene classification, IEEE Transactions on Image Processing. 23 (2014) pp. 3241–3253.

[17] M. Dixit, S. Chen, D. Gao, N. Rasiwasia, N. Vasconcelos, Scene classification with semantic fisher vectors, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015: pp. 2974–2983.

[18] W. Choi, Y.-W. Chao, C. Pantofaru, S. Savarese, Indoor scene understanding with geometric and semantic contexts, International Journal of Computer Vision. 112 (2015) pp. 204–220.

[19] S. Gupta, P. Arbeláez, R. Girshick, J. Malik, Indoor scene understanding with rgb-d images: Bottom-up segmentation, object detection and semantic segmentation, International Journal of Computer Vision. 112 (2015) pp. 133–149.

[20] M. Stark, P. Lies, M. Zillich, J. Wyatt, B. Schiele, Functional object class detection based on learned affordance cues, in: International Conference on Computer Vision Systems, Springer, 2008: pp. 435–444.

[21] Y. Zhu, A. Fathi, L. Fei-Fei, Reasoning about object affordances in a knowledge base representation, in: European Conference on Computer Vision, Springer, 2014: pp. 408–424.

[22] C. Ye, Y. Yang, R. Mao, C. Fermüller, Y. Aloimonos, What can i do around here? deep functional scene understanding for cognitive robots, in: 2017 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2017: pp. 4604–4611150904633X.

[23] J. Sawatzky, A. Srikantha, J. Gall, Weakly supervised affordance detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: pp. 2795–2804.

[24] M. Hassanin, S. Khan, M. Tahtali, Visual affordance and function understanding: A survey, ArXiv Preprint ArXiv:1807.06775. (2018).

[25] H. Zhang, J. Xue, K. Dana, Deep TEN: Texture Encoding Network, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2017: pp. 2896–2905. https://doi.org/10.1109/CVPR.2017.309.

[26] J. Xue, H. Zhang, K. Dana, Deep Texture Manifold for Ground Terrain Recognition, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. (2018) pp. 558–567. https://doi.org/10.1109/CVPR.2018.00065.

[27] W. Zhai, Y. Cao, Z.-J. Zha, H. Xie, F. Wu, Deep structure-revealed network for texture recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: pp. 11010–11019.

[28] Z. Chen, F. Li, Y. Quan, Y. Xu, H. Ji, Deep Texture Recognition via Exploiting Cross-Layer Statistical Self-Similarity, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021: pp. 5231–5240.

[29] CDC, Current HAI Progress Report, (2021). https://www.cdc.gov/hai/data/portal/progress-report.html (accessed March 9, 2022).

[30] F. de Castro Rodrigues Ferreira, M.P. Cristelli, M.I. Paula, H. Proença, C.R. Felipe, H. Tedesco-Silva, J.O. Medina-Pestana, Infectious complications as the

leading cause of death after kidney transplantation: analysis of more than 10,000 transplants from a single center, J Nephrol. 30 (2017) pp. 601–606.

[31] O. Assadian, S. Harbarth, M. Vos, J.K. Knobloch, A. Asensio, A.F. Widmer, Practical recommendations for routine cleaning and disinfection procedures in healthcare institutions: A narrative review, Journal of Hospital Infection. 113 (2021) pp. 104–114.

[32] M. Guettari, I. Gharbi, S. Hamza, UVC disinfection robot, Environmental Science and Pollution Research 2020 28:30. 28 (2020) pp. 40394–40399. https://doi.org/10.1007/S11356-020-11184-2.

[33] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu, Y. Liu, Deep learning for sensor-based human activity recognition: Overview, challenges, and opportunities, ACM Computing Surveys (CSUR). 54 (2021) pp. 1–40.

[34] G. Guo, A. Lai, A survey on still image based human action recognition, Pattern Recognition. 47 (2014) pp. 3343–3361.

[35] I. Rodríguez-Moreno, J.M. Martínez-Otzeta, B. Sierra, I. Rodriguez, E. Jauregi, Video Activity Recognition: State-of-the-Art, Sensors (Basel). 19 (2019). https://doi.org/10.3390/S19143160.

[36] Y. Wang, H. Jiang, M.S. Drew, Z.N. Li, G. Mori, Unsupervised discovery of action classes, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 2006: pp. 1654–1661. https://doi.org/10.1109/CVPR.2006.321.

[37] N. Ikizler, R.G. Cinbis, S. Pehlivan, P. Duygulu, Recognizing actions from still images, in: Proceedings - International Conference on Pattern Recognition, IEEE, 2008: pp. 1–4. https://doi.org/10.1109/icpr.2008.4761663.

[38] B. Yao, X. Jiang, A. Khosla, A.L. Lin, L. Guibas, L. Fei-Fei, Human action recognition by learning bases of action attributes and parts, in: 2011 International Conference on Computer Vision, IEEE, 2011: pp. 1331–13381457711028.

[39] Y. Yun, I.Y.-H. Gu, H. Aghajan, Riemannian manifold-based support vector machine for human activity classification in images, in: 2013 IEEE International Conference on Image Processing, IEEE, 2013: pp. 3466–34691479923419.

[40] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014: pp. 1717–1724.

[41] G. Gkioxari, R. Girshick, J. Malik, Contextual action recognition with r* cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2015: pp. 1080–1088.

[42] Z. Zhao, H. Ma, S. You, Single image action recognition using semantic body part actions, in: Proceedings of the IEEE International Conference on Computer Vision, 2017: pp. 3391–3399.

[43] F.S. Khan, J. Xu, J. van de Weijer, A.D. Bagdanov, R.M. Anwer, A.M. Lopez, Recognizing actions through action-specific person detection, IEEE Transactions on Image Processing. 24 (2015) pp. 4422–4432.

[44] A.R. Siyal, Z. Bhutto, S. Muhammad, A. Iqbal, F. Mehmood, A. Hussain, S. Ahmed, Still Image-based Human Activity Recognition with Deep Representations and Residual Learning, International Journal of Advanced Computer Science and Applications. 11 (2020). https://doi.org/10.14569/IJACSA.2020.0110561.

[45] A. Bera, Z. Wharton, Y. Liu, N. Bessis, A. Behera, Attend and guide (ag-net): A keypoints-driven attention-based deep network for image recognition, IEEE Transactions on Image Processing. 30 (2021) pp. 3691–3704.

[46] C. Bas, N. Ikizler-Cinbis, Top-down and bottom-up attentional multiple instance learning for still image action recognition, Signal Processing: Image Communication. (2022) pp. 116664. https://doi.org/10.1016/J.IMAGE.2022.116664.

[47] P. Khaire, P. Kumar, J. Imran, Combining CNN streams of RGB-D and skeletal data for human activity recognition, Pattern Recognition Letters. 115 (2018) pp. 107–116. https://doi.org/10.1016/j.patrec.2018.04.035.

[48] L. Guo, L. Wang, J. Liu, W. Zhou, B. Lu, HuAc: Human Activity Recognition Using Crowdsourced WiFi Signals and Skeleton Data, Wireless Communications and Mobile Computing. 2018 (2018). https://doi.org/10.1155/2018/6163475.

[49] R. Singh, R. Khurana, A.K.S. Kushwaha, R. Srivastava, Combining CNN streams of dynamic image and depth data for action recognition, Multimedia Systems. 26 (2020) pp. 313–322. https://doi.org/10.1007/s00530-019-00645-5.

[50] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, S. Xie, F.A. Research, A ConvNet for the 2020s, ArXiv. (2022). https://doi.org/10.48550/arxiv.2201.03545.

[51] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021: pp. 10012–10022.

[52] K. Tang, Y. Niu, J. Huang, J. Shi, H. Zhang, Unbiased scene graph generation from biased training, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: pp. 3716–3725.

[53] D. Xu, Y. Zhu, C.B. Choy, L. Fei-Fei, Scene graph generation by iterative message passing, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: pp. 5410–5419.

[54] K. Tang, H. Zhang, B. Wu, W. Luo, W. Liu, Learning to compose dynamic tree structures for visual contexts, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019: pp. 6619–6628.

[55] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D.A. Shamma, Visual genome: Connecting language and vision using crowdsourced dense image annotations, Int J Comput Vis. 123 (2017) pp. 32–73.

[56] P. Veličković, A. Casanova, P. Liò, G. Cucurull, A. Romero, Y. Bengio, Graph Attention Networks, 6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings. (2017). https://doi.org/10.48550/arxiv.1710.10903.

[57] S. Brody, U. Alon, E. Yahav, How Attentive are Graph Attention Networks?, in: International Conference on Learning Representations, 2022. https://doi.org/10.48550/arxiv.2105.14491.

[58] G. Rogova, Combining the results of several neural network classifiers, in: Classic Works of the Dempster-Shafer Theory of Belief Functions, Springer, 2008: pp. 683–692.

[59] J. Daniel, J.-P. Lauffenburger, Conflict management in multi-sensor dempster-shafer fusion for speed limit determination, in: 2011 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2011: pp. 987–9921457708914.

[60] L. van der Maaten, G. Hinton, Visualizing data using t-SNE., Journal of Machine Learning Research. 9 (2008).

[61] D. Kobak, P. Berens, The art of using t-SNE for single-cell transcriptomics, Nat Commun. 10 (2019) pp. 1–14.

[62] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, Pytorch: An imperative style, high-performance deep learning library, Adv Neural Inf Process Syst. 32 (2019) pp. 8026–8037.

[63] S. Ruder, An overview of gradient descent optimization algorithms, ArXiv Preprint ArXiv:1609.04747. (2016).

[64] G. Gkioxari, R. Girshick, J. Malik, Contextual action recognition with r* cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2015: pp. 1080–1088.

[65] Y. Zhang, L. Cheng, J. Wu, J. Cai, M.N. Do, J. Lu, Action recognition in still images with minimum annotation efforts, IEEE Transactions on Image Processing. 25 (2016) pp. 5479–5490.

[66] Z. Zhao, H. Ma, X. Chen, Semantic parts based top-down pyramid for action recognition, Pattern Recognition Letters. 84 (2016) pp. 134–141.

[67] W. Feng, X. Zhang, X. Huang, Z. Luo, Attention focused spatial pyramid pooling for boxless action recognition in still images, in: International Conference on Artificial Neural Networks, Springer, 2017: pp. 574–581.

[68] S. Yan, J.S. Smith, W. Lu, B. Zhang, Multibranch attention networks for action recognition in still images, IEEE Transactions on Cognitive and Developmental Systems. 10 (2017) pp. 1116–1125.

[69] X. Zheng, T. Gong, X. Lu, X. Li, Human action recognition by multiple spatial clues network, Neurocomputing. 483 (2022) pp. 10–21. https://doi.org/10.1016/J.NEUCOM.2022.01.091.

[70] G. Gkioxari, R. Girshick, J. Malik, Actions and attributes from wholes and parts, in: Proceedings of the IEEE International Conference on Computer Vision, 2015: pp. 2470–2478.

[71] Z. Zhao, H. Ma, X. Chen, Generalized symmetric pair model for action classification in still images, Pattern Recognition. 64 (2017) pp. 347–360. https://doi.org/10.1016/J.PATCOG.2016.10.001.

[72] Q. Li, Z. Han, X.-M. Wu, Deeper insights into graph convolutional networks for semi-supervised learning, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[73] U. Alon, E. Yahav, On the Bottleneck of Graph Neural Networks and its Practical Implications, in: International Conference on Learning Representations, 2021. http://arxiv.org/abs/2006.05205 (accessed March 8, 2022).

[74] K. Huslage, W.A. Rutala, E. Sickbert-Bennett, D.J. Weber, A quantitative approach to defining "high-touch" surfaces in hospitals, Infection Control & Hospital Epidemiology. 31 (2010) pp. 850–853. https://doi.org/10.1086/655016.

[75] P.C. Carling, M.F. Parry, L.A. Bruno-Murtha, B. Dick, Improving environmental hygiene in 27 intensive care units to decrease multidrug-resistant bacterial transmission, Critical Care Medicine. 38 (2010) pp. 1054–1059. https://doi.org/10.1097/CCM.0b013e3181cdf705.

[76] F.S. Bashiri, E. LaRose, P. Peissig, A.P. Tafti, MCIndoor20000: A fully-labeled image dataset to advance indoor objects detection, Data Brief. 17 (2018) pp. 71–75. https://doi.org/10.1016/j.dib.2017.12.047.

[77] A. Ismail, S.A. Ahmad, A. Che Soh, M.K. Hassan, H.H. Harith, MYNursingHome: A fully-labelled image dataset for indoor object classification, Data in Brief. 32 (2020) pp. 106268. https://doi.org/10.1016/j.dib.2020.106268.

[78] A. Vasquez, M. Kollmitz, A. Eitel, W. Burgard, Deep Detection of People and their Mobility Aids for a Hospital Robot, in: 2017 European Conference on Mobile Robots (ECMR), IEEE, 2017: pp. 1–7. https://doi.org/10.1109/ECMR.2017.8098665.

[79] F.M.T.R. Kinasih, C. Machbub, L. Yulianti, A.S. Rohman, Centroid-Tracking-Aided Robust Object Detection for Hospital Objects, in: 2020 6th International Conference on Interactive Digital Media (ICIDM), IEEE, 2020: pp. 1–5. https://doi.org/10.1109/ICIDM51048.2020.9339679.

[80] Y. Liu, Y. Liu, J. Tang, E. Yin, D. Hu, Z. Zhou, A self-paced BCI prototype system based on the incorporation of an intelligent environment-understanding approach for rehabilitation hospital environmental control, Computers in Biology and Medicine. 118 (2020) pp. 103618. https://doi.org/10.1016/j.compbiomed.2020.103618.

[81] F. Zhong, S. Wang, Z. Zhang, Y. Wang, Detect-SLAM: Making object detection and SLAM mutually beneficial, in: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2018: pp. 1001–1010. https://doi.org/10.1109/WACV.2018.00115.

[82] B. Bescos, J.M. Facil, J. Civera, J. Neira, DynaSLAM: Tracking, Mapping, and Inpainting in Dynamic Scenes, IEEE Robotics and Automation Letters. 3 (2018) pp. 4076–4083. https://doi.org/10.1109/LRA.2018.2860039.

[83] L. Nicholson, M. Milford, N. Sünderhauf, Quadricslam: Dual quadrics from object detections as landmarks in object-oriented slam, IEEE Robotics and Automation Letters. 4 (2018) pp. 1–8. https://doi.org/10.1109/LRA.2018.2866205.

[84] Y. Liu, M. Xu, G. Jiang, X. Tong, J. Yun, Y. Liu, B. Chen, Y. Cao, N. Sun, Z. Li, Target localization in local dense mapping using RGBD SLAM and object detection, Concurrency and Computation: Practice and Experience. 34 (2022) pp. e6655. https://doi.org/10.1002/cpe.6655.

[85] A. Rosinol, A. Violette, M. Abate, N. Hughes, Y. Chang, J. Shi, A. Gupta, L. Carlone, Kimera: From SLAM to spatial perception with 3D dynamic scene graphs, The International Journal of Robotics Research. 40 (2021) pp. 1510–1546. https://doi.org/10.1177/02783649211056674.

[86] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: pp. 779–788.

[87] G. Jocher, A. Stoken, A. Chaurasia, N. Jirka Borovec, TaoXie, Y. Kwon, K. Michael, C. Liu, J. Fang, A. V, L. Tkianai, YxNONG, P. Skalski, A. Hogan, J. Nadar, L.M. Imyhxy, ultralytics/yolov5: v6.0 - YOLOv5n “Nano” models, Roboflow integration, TensorFlow export, OpenCV DNN support (v6.0), Zenodo. (2021). https://doi.org/10.5281/zenodo.5563715.

[88] C.-Y. Wang, H.-Y.M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, I.-H. Yeh, CSPNet: A new backbone that can enhance learning capability of CNN, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020: pp. 390–391. https://doi.org/10.1109/CVPRW50498.2020.00203.

[89] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path aggregation network for instance segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: pp. 8759–8768. https://doi.org/10.1109/CVPR.2018.00913.

[90] M.-H. Guo, T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, T.-J. Mu, S.-H. Zhang, R.R. Martin, M.-M. Cheng, S.-M. Hu, Attention mechanisms in computer vision: A survey, Computational Visual Media. (2022) pp. 1–38. https://doi.org/10.1007/s41095-022-0271-y.

[91] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, T.-S. Chua, Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: pp. 5659–5667. https://doi.org/10.1109/CVPR.2017.667.

[92] J. Hu, L. Shen, G. Sun, Squeeze-and-Excitation Networks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, 2018: pp. 7132–7141. https://doi.org/10.1109/CVPR.2018.00745.

[93] Q. Hou, D. Zhou, J. Feng, Coordinate attention for efficient mobile network design, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021: pp. 13713–13722.

[94] J. He, S. Erfani, X. Ma, J. Bailey, Y. Chi, X.-S. Hua, $\alpha $-IoU: A Family of Power Intersection over Union Losses for Bounding Box Regression, Advances in Neural Information Processing Systems. 34 (2021).

[95] G. Bradski, A. Kaehler, OpenCV, Dr. Dobb's Journal of Software Tools. 3 (2000).

[96] M. Labbé, F. Michaud, RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation, Journal of Field Robotics. 36 (2019) pp. 416–446.

[97] R. Mur-Artal, J.D. Tardós, Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras, IEEE Transactions on Robotics. 33 (2017) pp. 1255–1262.

[98] N. Kejriwal, S. Kumar, T. Shibata, High performance loop closure detection using bag of word pairs, Robotics and Autonomous Systems. 77 (2016) pp. 55–65. https://doi.org/10.1016/j.robot.2015.12.003.

[99] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, ORB: An efficient alternative to SIFT or SURF, in: 2011 International Conference on Computer Vision, Ieee, 2011: pp. 2564–25711457711028.

[100] E. Schubert, J. Sander, M. Ester, H.P. Kriegel, X. Xu, DBSCAN revisited, revisited: why and how you should (still) use DBSCAN, ACM Transactions on Database Systems (TODS). 42 (2017) pp. 1–21.

[101] M.C. Fitzpatrick, C.T. Bauch, J.P. Townsend, A.P. Galvani, Modelling microbial infection to address global health challenges, Nature Microbiology. 4 (2019) pp. 1612–1619. https://doi.org/10.1038/s41564-019-0565-8.

[102] W.C.W.S. Putri, D.J. Muscatello, M.S. Stockwell, A.T. Newall, Economic burden of seasonal influenza in the United States, Vaccine. 36 (2018) pp. 3960–3966. https://doi.org/10.1016/j.vaccine.2018.05.057.

[103] CDC, 2019-2020 U.S. Flu Season: Preliminary Burden Estimates, (2020). https://www.cdc.gov/flu/about/burden/preliminary-in-season-estimates.htm (accessed August 17, 2020).

[104] E.J. Septimus, J. Moody, Prevention of device-related healthcare-associated infections, F1000Res. 5 (2016).

[105] C. Siemaszko, Coronavirus forces New York City subways, trains to clean up their act, NBC News. (2020).

[106] H. Leone, Every public and private school in Illinois is closed because of the coronavirus. Here’s what you need to know., Chicago Tribune. (2020).

[107] K.R. Bright, S.A. Boone, C.P. Gerba, Occurrence of bacteria and viruses on elementary classroom surfaces and the potential role of classroom hygiene in the spread of infectious diseases, The Journal of School Nursing. 26 (2010) pp. 33–41.

[108] G. Kampf, D. Todt, S. Pfaender, E. Steinmann, Persistence of coronaviruses on inanimate surfaces and its inactivation with biocidal agents, Journal of Hospital Infection. (2020).

[109] A.S. for Microbiology, How quickly viruses can contaminate buildings -- from just a single doorknob, ScienceDaily. (2014).

[110] O. Dumas, R. Varraso, K.M. Boggs, C. Quinot, J.-P. Zock, P.K. Henneberger, F.E. Speizer, N. Le Moual, C.A. Camargo, Association of Occupational Exposure to Disinfectants With Incidence of Chronic Obstructive Pulmonary Disease Among US Female Nurses, JAMA Netw Open. 2 (2019) pp. e1913563–e1913563.

[111] T. Weinmann, J. Gerlich, S. Heinrich, D. Nowak, E. Von Mutius, C. Vogelberg, J. Genuneit, S. Lanzinger, S. Al-Khadra, T. Lohse, Association of household cleaning agents and disinfectants with asthma in young German adults, Occupational and Environmental Medicine. 74 (2017) pp. 684–690.

[112] A. Begić, Application of Service Robots for Disinfection in Medical Institutions, in: International Symposium on Innovative and Interdisciplinary Applications of Advanced Technologies, Springer, 2017: pp. 1056–1065.

[113] D. Gibson, S. Kendrick, E. Simpson, D. Costello, R. Davis, A. Szetela, M. McCreary, J. Schriber, Implementation of Xenon Ultraviolet-C Disinfection Robot to Reduce Hospital Acquired Infections in Hematopoietic Stem Cell Transplant Population, Biology of Blood and Marrow Transplantation. 23 (2017) pp. S472.

[114] M. Hui, Hong Kong's subway is sending robots to disinfect trains of coronavirus, Quartz. (2020).

[115] V. Prabakaran, M.R. Elara, T. Pathmakumar, S. Nansai, hTetro: A tetris inspired shape shifting floor cleaning robot, in: 2017 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2017: pp. 6105–6112150904633X.

[116] P. Veerajagadheswar, M.R. Elara, T. Pathmakumar, V. Ayyalusami, A tiling-theoretic approach to efficient area coverage in a tetris-inspired floor cleaning robot, IEEE Access. 6 (2018) pp. 35260–35271.

[117] V. Prabakaran, M.R. Elara, T. Pathmakumar, S. Nansai, Floor cleaning robot with reconfigurable mechanism, Automation in Construction. 91 (2018) pp. 155–165.

[118] M.A.V.J. Muthugala, A. Vengadesh, X. Wu, M.R. Elara, M. Iwase, L. Sun, J. Hao, Expressing attention requirement of a floor cleaning robot through interactive lights, Automation in Construction. 110 (2020) pp. 103015.

[119] G. Grisetti, C. Stachniss, W. Burgard, Improved techniques for grid mapping with rao-blackwellized particle filters, IEEE Transactions on Robotics. 23 (2007) pp. 34–46.

[120] B. Steux, O.T. El Hamzaoui, A SLAM algorithm in less than 200 lines C-language program, Proceedings of the Control Automation Robotics & Vision (ICARCV), Singapore. (2010) pp. 7–10.

[121] S. Kohlbrecher, J. Meyer, T. Graber, K. Petersen, U. Klingauf, O. Von Stryk, LNAI 8371 - Hector Open Source Modules for Autonomous Mapping and Navigation with Rescue Robots, 2013.

[122] F. Pomerleau, F. Colas, R. Siegwart, S. Magnenat, Comparing ICP variants on real-world data sets, Autonomous Robots. 34 (2013) pp. 133–148.

[123] W. Hess, D. Kohler, H. Rapp, D. Andor, Real-time loop closure in 2D LIDAR SLAM, in: 2016 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2016: pp. 1271–12781467380261.

[124] T. Schneider, M. Dymczyk, M. Fehr, K. Egger, S. Lynen, I. Gilitschenski, R. Siegwart, maplab: An open framework for research in visual-inertial mapping and localization, IEEE Robotics and Automation Letters. 3 (2018) pp. 1418–1425.

[125] Y. Lin, F. Gao, T. Qin, W. Gao, T. Liu, W. Wu, Z. Yang, S. Shen, Autonomous aerial navigation using monocular visual-inertial fusion, Journal of Field Robotics. 35 (2018) pp. 23–51.

[126] C. Kerl, J. Sturm, D. Cremers, Dense visual SLAM for RGB-D cameras, in: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2013: pp. 2100–21061467363588.

[127] T. Luddecke, F. Worgotter, Learning to Segment Affordances, 2017. https://doi.org/10.1109/ICCVW.2017.96.

[128] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba, Scene parsing through ADE20K dataset, 2017. https://doi.org/10.1109/CVPR.2017.544.

[129] T. Lüddecke, F. Wörgötter, Learning to Label Affordances from Simulated and Real Data, (2017). https://doi.org/10.48550/arXiv.1709.08872.

[130] T. Lüddecke, T. Kulvicius, F. Wörgötter, Context-based affordance segmentation from 2D images for robot actions, Robotics and Autonomous Systems. 119 (2019) pp. 92–107. https://doi.org/10.1016/j.robot.2019.05.005.

[131] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer Verlag, 2015: pp. 234–241. https://doi.org/10.1007/978-3-319-24574-4_28.

[132] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, 2016. https://doi.org/10.1109/CVPR.2016.90.

[133] P.O. Pinheiro, T.Y. Lin, R. Collobert, P. Dollár, Learning to refine object segments, in: European Conference on Computer Vision, 2016: pp. 75–91. https://doi.org/10.1007/978-3-319-46448-0_5.

[134] G. Ghiasi, C.C. Fowlkes, Laplacian pyramid reconstruction and refinement for semantic segmentation, in: European Conference on Computer Vision, Springer, 2016: pp. 519–534.

[135] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018: pp. 801–818.

[136] M. Tan, Q. V. Le, EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks, 36th International Conference on Machine Learning, ICML 2019. 2019-June (2019) pp. 10691–10700.

[137] A. Hornung, K.M. Wurm, M. Bennewitz, C. Stachniss, W. Burgard, OctoMap: An efficient probabilistic 3D mapping framework based on octrees, Autonomous Robots. 34 (2013) pp. 189–206. https://doi.org/10.1007/s10514-012-9321-0.

[138] P.E. Hart, N.J. Nilsson, B. Raphael, A Formal Basis for the Heuristic Determination of Minimum Cost Paths, IEEE Transactions on Systems Science and Cybernetics. 4 (1968) pp. 100–107. https://doi.org/10.1109/TSSC.1968.300136.

[139] D. Fox, W. Burgard, S. Thrun, The dynamic window approach to collision avoidance, IEEE Robotics and Automation Magazine. 4 (1997) pp. 23–33. https://doi.org/10.1109/100.580977.

[140] X. Peng, Z. Tang, F. Yang, R.S. Feris, D. Metaxas, Jointly optimize data augmentation and network training: Adversarial data augmentation in human pose estimation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: pp. 2226–2234.

[141] H. Noh, S. Hong, B. Han, Learning deconvolution network for semantic segmentation, in: Proceedings of the IEEE International Conference on Computer Vision, 2015: pp. 1520–1528.

[142] Y. Li, H. Qi, J. Dai, X. Ji, Y. Wei, Fully convolutional instance-aware semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: pp. 2359–2367.

[143] M. Kampffmeyer, A.-B. Salberg, R. Jenssen, Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2016: pp. 1–9.

[144] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch, in: NIPS 2017 Workshop, 2017.

[145] T. Tieleman, G. Hinton, Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude, COURSERA: Neural Networks for Machine Learning. 4 (2012) pp. 26–31.

[146] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, Ieee, 2009: pp. 248–2551424439922.

[147] C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals, Understanding deep learning requires rethinking generalization, ArXiv Preprint ArXiv:1611.03530. (2016).

[148] A. Roy, S. Todorovic, A multi-scale cnn for affordance segmentation in rgb images, in: European Conference on Computer Vision, Springer, 2016: pp. 186–201.

[149] D. Lewis, Why indoor spaces are still prime COVID hotspots., Nature. 592 (2021) pp. 22–25.

[150] C.J. Donskey, Does improving surface cleaning and disinfection reduce health care-associated infections?, Am J Infect Control. 41 (2013) pp. S12–S19.

[151] A.W.H. Chin, J.T.S. Chu, M.R.A. Perera, K.P.Y. Hui, H.-L. Yen, M.C.W. Chan, M. Peiris, L.L.M. Poon, Stability of SARS-CoV-2 in different environmental conditions, The Lancet Microbe. 1 (2020) pp. e10.

[152] G.U. Lopez, C.P. Gerba, A.H. Tamimi, M. Kitajima, S.L. Maxwell, J.B. Rose, Transfer efficiency of bacteria and viruses from porous and nonporous fomites to

fingers under different relative humidity conditions, Applied and Environmental Microbiology. 79 (2013) pp. 5728–5734. https://doi.org/10.1128/AEM.01030-13.

[153] B. Caputo, E. Hayman, P. Mallikarjuna, Class-specific material categorisation, in: Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, IEEE, 2005: pp. 1597–1604076952334X.

[154] S. Lazebnik, C. Schmid, J. Ponce, A discriminative framework for texture and object recognition using local image features, in: Toward Category-Level Object Recognition, Springer, 2006: pp. 423–442.

[155] J. Kang, Y.-J. Park, J. Lee, S.-H. Wang, D.-S. Eom, Novel leakage detection by ensemble CNN-SVM and graph-based localization in water distribution systems, IEEE Transactions on Industrial Electronics. 65 (2017) pp. 4279–4289.

[156] M. Cimpoi, S. Maji, I. Kokkinos, A. Vedaldi, Deep Filter Banks for Texture Recognition, Description, and Segmentation, International Journal of Computer Vision. 118 (2016) pp. 65–94. https://doi.org/10.1007/s11263-015-0872-3.

[157] Y. Song, F. Zhang, Q. Li, H. Huang, L.J. O'Donnell, W. Cai, Locally-transferred fisher vectors for texture classification, in: Proceedings of the IEEE International Conference on Computer Vision, 2017: pp. 4912–4920.

[158] S. Bell, P. Upchurch, N. Snavely, K. Bala, Material recognition in the wild with the Materials in Context Database, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 07-12-June (2015) pp. 3479–3487. https://doi.org/10.1109/CVPR.2015.7298970.

[159] J. Xue, H. Zhang, K. Dana, K. Nishino, Differential Angular Imaging for Material Recognition, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2017: pp. 6940–6949. https://doi.org/10.1109/CVPR.2017.734.

[160] W. Zhai, Y. Cao, J. Zhang, Z.J. Zha, Deep multiple-attribute-perceived network for real-world texture recognition, Proceedings of the IEEE International Conference on Computer Vision. 2019-Octob (2019) pp. 3612–3621. https://doi.org/10.1109/ICCV.2019.00371.

[161] D. Hu, H. Zhong, S. Li, J. Tan, Q. He, Segmenting areas of potential contamination for adaptive robotic disinfection in built environments, Building and Environment. 184 (2020) pp. 107226. https://doi.org/10.1016/j.buildenv.2020.107226.

[162] L.C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018: pp. 801–918. https://doi.org/10.1007/978-3-030-01234-2_49.

[163] S. Wold, K. Esbensen, P. Geladi, Principal component analysis, Chemometrics and Intelligent Laboratory Systems. 2 (1987) pp. 37–52. https://doi.org/10.1016/0169-7439(87)80084-9.

[164] L. Sharan, R. Rosenholtz, E.H. Adelson, Accuracy and speed of material categorization in real-world images, J Vis. 14 (2014) pp. 12.

[165] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, A. Vedaldi, Describing textures in the wild, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. (2014) pp. 3606–3613. https://doi.org/10.1109/CVPR.2014.461.

[166] P. Mallikarjuna, A.T. Targhi, M. Fritz, E. Hayman, B. Caputo, J.-O. Eklundh, The kth-tips2 database, Computational Vision and Active Perception Laboratory, Stockholm, Sweden. (2006) pp. 1–10.

[167] T.Y. Lin, S. Maji, Visualizing and Understanding Deep Texture Representations, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2016-Decem (2016) pp. 2791–2799. https://doi.org/10.1109/CVPR.2016.305.

[168] X. Dai, J. Yue-Hei Ng, L.S. Davis, FASON: First and second order information fusion network for texture recognition, in: Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017: pp. 6100–6108. https://doi.org/10.1109/CVPR.2017.646.

[169] Y. Hu, Z. Long, G. AlRegib, Multi-level texture encoding and representation (multer) based on deep neural networks, in: 2019 IEEE International Conference on Image Processing (ICIP), IEEE, 2019: pp. 4410–44141538662493.

[170] C. Poynton, Digital video and HD: Algorithms and Interfaces, Elsevier, 20120123919320.

[171] MI, Disinfectant Robot Market - Growth, Trends, COVID-19 Impact, and Forecasts (2021 - 2026), (2021). https://www.mordorintelligence.com/industry-reports/disinfectant-robot-market (accessed July 13, 2021).

[172] M. Fleming, A. Patrick, M. Gryskevicz, N. Masroor, L. Hassmer, K. Shimp, K. Cooper, M. Doll, M. Stevens, G. Bearman, Deployment of a touchless ultraviolet light robot for terminal room disinfection: The importance of audit and feedback, American Journal of Infection Control. 46 (2018) pp. 241–243. https://doi.org/10.1016/j.ajic.2017.09.027.

[173] Rachel Gordon, CSAIL robot disinfects Greater Boston Food Bank, (2020). https://news.mit.edu/2020/csail-robot-disinfects-greater-boston-food-bank-covid-19-0629 (accessed November 24, 2021).

[174] C. McGinn, R. Scott, N. Donnelly, K.L. Roberts, M. Bogue, C. Kiernan, M. Beckett, Exploring the Applicability of Robot-Assisted UV Disinfection in Radiology, Frontiers in Robotics and AI. 7 (2020) pp. 590306. https://doi.org/10.3389/FROBT.2020.590306.

[175] James Vincent, Toyota’s robot butler prototype hangs from the ceiling like a bat, (2020). https://www.theverge.com/2020/10/1/21496692/toyota-robots-tri-research-institute-home-helping-gantry-ceiling-machine (accessed November 24, 2021).

[176] B. Ramalingam, J. Yin, M. Rajesh Elara, Y.K. Tamilselvam, M. Mohan Rayguru, M.A. Muthugala, B. Félix Gómez, A human support robot for the cleaning and maintenance of door handles using a deep-learning framework, Sensors. 20 (2020) pp. 3543.

[177] Fraunhofer IPA, Intelligent robots for targeted combating of viruses and bacteria , (2021). https://www.ipa.fraunhofer.de/en/press-media/press_releases/intelligent-

robots-for-targeted-combating-of-viruses-and-bacteria.html (accessed November 24, 2021).

[178] Y. Abu-Zidan, K. Nguyen, P. Mendis, S. Setunge, H. Adeli, Design of a smart prefabricated sanitising chamber for COVID-19 using computational fluid dynamics, Journal of Civil Engineering and Management. 27 (2021) pp. 139–148. https://doi.org/10.3846/jcem.2021.14348.

[179] Y.-L. Zhao, H.-P. Huang, T.-L. Chen, P.-C. Chiang, Y.-H. Chen, J.-H. Yeh, C.-H. Huang, J.-F. Lin, W.-T. Weng, A Smart Sterilization Robot System with Chlorine Dioxide for Spray Disinfection, IEEE Sensors Journal. (2021).

[180] S. Thakar, R.K. Malhan, P.M. Bhatt, S.K. Gupta, Area-coverage planning for spray-based surface disinfection with a mobile manipulator, Robotics and Autonomous Systems. (2021) pp. 103920.

[181] L. Guo, Z. Yang, L. Guo, L. Chen, Z. Cheng, L. Zhang, E. Long, Study on the decay characteristics and transmission risk of respiratory viruses on the surface of objects, Environmental Research. 194 (2021) pp. 110716. https://doi.org/10.1016/J.ENVRES.2021.110716.

[182] A.K. Pitol, T.R. Julian, Community Transmission of SARS-CoV-2 by Surfaces: Risks and Risk Reduction Strategies, Environmental Science & Technology Letters. 8 (2021) pp. 263–269. https://doi.org/10.1021/ACS.ESTLETT.0C00966.

[183] W. Kowalski, Mathematical Modeling of UV Disinfection, in: Ultraviolet Germicidal Irradiation Handbook, Springer, 2009: pp. 51–72.

[184] W. Kowalski, Ultraviolet germicidal irradiation handbook: UVGI for air and surface disinfection, Springer science & business media, 20103642019994.

[185] P. Beeson, B. Ames, TRAC-IK: An open-source library for improved solving of generic inverse kinematics, in: IEEE-RAS International Conference on Humanoid Robots, IEEE Computer Society, 2015: pp. 928–935. https://doi.org/10.1109/HUMANOIDS.2015.7363472.

[186] C.R. Bergeron, C. Prussing, P. Boerlin, D. Daignault, L. Dutil, R.J. Reid-Smith, G.G. Zhanel, A.R. Manges, Chicken as reservoir for extraintestinal pathogenic Escherichia coli in humans, Canada, Emerg Infect Dis. 18 (2012) pp. 415.

[187] Y. Pan, D. Zhang, P. Yang, L.L.M. Poon, Q. Wang, Viral load of SARS-CoV-2 in clinical samples, The Lancet Infectious Diseases. 20 (2020) pp. 411–412. https://doi.org/10.1016/S1473-3099(20)30113-4.

[188] J.E. Wißmann, L. Kirchhoff, Y. Brüggemann, D. Todt, J. Steinmann, E. Steinmann, Persistence of Pathogens on Inanimate Surfaces: A Narrative Review, Microorganisms 2021, Vol. 9, Page 343. 9 (2021) pp. 343. https://doi.org/10.3390/MICROORGANISMS9020343.

[189] J. Virtanen, K. Aaltonen, I. Kivistö, T. Sironen, Survival of SARS-CoV-2 on Clothing Materials, (2021). https://doi.org/10.1155/2021/6623409.

[190] N. Castaño, S.C. Cordts, M.K. Jalil, K.S. Zhang, S. Koppaka, A.D. Bick, R. Paul, S.K.Y. Tang, Fomite Transmission, Physicochemical Origin of Virus–Surface Interactions, and Disinfection Strategies for Enveloped Viruses with Applications to SARS-CoV-2, ACS Omega. 6 (2021) pp. 6509–6527. https://doi.org/10.1021/ACSOMEGA.0C06335.

[191] W.R. Richter, M.M. Sunderman, M.Q.S. Wendling, S. Serre, L. Mickelsen, R. Rupert, J. Wood, Y. Choi, Z. Willenberg, M.W. Calfee, Evaluation of altered environmental conditions as a decontamination approach for nonspore-forming biological agents, J Appl Microbiol. 128 (2020) pp. 1050–1059.

[192] S.L. Warnes, Z.R. Little, C.W. Keevil, R.R. Colwell, Human Coronavirus 229E Remains Infectious on Common Touch Surface Materials, (2015). https://doi.org/10.1128/mBio.01697-15.

[193] D.C. Esteves, V.C. Pereira, J.M. Souza, R. Keller, R.D. Simões, L.K.W. Eller, M.V.P. Rodrigues, Influence of biological fluids in bacterial viability on different hospital surfaces and fomites, Am J Infect Control. 44 (2016) pp. 311–314.

[194] A. Pintola, Relation of Wettability of Surfaces to Virus Survival Times, The University of Akron, 2021.

[195] C.E. Anderson, A.B. Boehm, Transfer Rate of Enveloped and Nonenveloped Viruses between Fingerpads and Surfaces, Applied and Environmental Microbiology. 87 (2021) pp. AEM-01215. https://doi.org/10.1128/AEM.01215-21.

[196] A.P. Harvey, E.R. Fuhrmeister, M.E. Cantrell, A.K. Pitol, J.M. Swarthout, J.E. Powers, M.L. Nadimpalli, T.R. Julian, A.J. Pickering, Longitudinal Monitoring of SARS-CoV-2 RNA on High-Touch Surfaces in a Community Setting, Environmental Science and Technology Letters. 8 (2021) pp. 168–175. https://doi.org/10.1021/acs.estlett.0c00875.

[197] J.S. Abrahão, L. Sacchetto, I.M. Rezende, R.A.L. Rodrigues, A.P.C. Crispim, C. Moura, D.C. Mendonça, E. Reis, F. Souza, G.F.G. Oliveira, I. Domingos, P.V. de Miranda Boratto, P.H.B. Silva, V.F. Queiroz, T.B. Machado, L.A.F. Andrade, K.L. Lourenço, T. Silva, G.P. Oliveira, V. de Souza Alves, P.A. Alves, E.G. Kroon, G. de Souza Trindade, B.P. Drumond, Detection of SARS-CoV-2 RNA on public surfaces in a densely populated urban area of Brazil: A potential tool for monitoring the circulation of infected patients, Science of The Total Environment. 766 (2021) pp. 142645. https://doi.org/10.1016/J.SCITOTENV.2020.142645.

[198] A. Cinar, E. ONBAŞI, Monitoring environmental microbiological safety in a frozen fruit and vegetable plant, Food Science and Technology. 41 (2020) pp. 232–237.

[199] M.C. Collivignarelli, A. Abbà, I. Benigna, S. Sorlini, V. Torretta, Overview of the main disinfection processes for wastewater and drinking water treatment plants, Sustainability. 10 (2018) pp. 86.

[200] A.K. Pitol, H.N. Bischel, T. Kohn, T.R. Julian, Virus transfer at the skin–liquid interface, Environ Sci Technol. 51 (2017) pp. 14417–14425.

[201] W. AuYeung, R.A. Canales, J.O. Leckie, The fraction of total hand surface area involved in young children’s outdoor hand-to-object contacts, Environ Res. 108 (2008) pp. 294–299.

[202] EPA, Exposure Factors Handbook 2011 Edition (Final Report), Washington, DC, 2011.

[203] A.N.M. Kraay, M.A.L. Hayashi, N. Hernandez-Ceron, I.H. Spicknall, M.C. Eisenberg, R. Meza, J.N.S. Eisenberg, Fomite-mediated transmission as a sufficient pathway: a comparative analysis across three viral pathogens, BMC

Infectious Diseases. 18 (2018) pp. 540. https://doi.org/10.1186/s12879-018-3425-x.

[204] A.P. Harvey, E.R. Fuhrmeister, M.E. Cantrell, A.K. Pitol, J.M. Swarthout, J.E. Powers, M.L. Nadimpalli, T.R. Julian, A.J. Pickering, Longitudinal Monitoring of SARS-CoV-2 RNA on High-Touch Surfaces in a Community Setting, Environmental Science and Technology Letters. 8 (2021) pp. 168–175. https://doi.org/10.1021/acs.estlett.0c00875.

[205] P. Mudgal, F. Breidt, S.R. Lubkin, K.P. Sandeep, Quantifying the Significance of Phage Attack on Starter Cultures: a Mechanistic Model for Population Dynamics of Phage and Their Hosts Isolated from Fermenting Sauerkraut, Applied and Environmental Microbiology. 72 (2006) pp. 3908–3915. https://doi.org/10.1128/AEM.02429-05.

[206] S. Li, J.N.S. Eisenberg, I.H. Spicknall, J.S. Koopman, Dynamics and control of infections transmitted from person to person through the environment, Am J Epidemiol. 170 (2009) pp. 257–265.

VITA

Da Hu was born in Neijiang City, Sichuan Province, China. In 2022, he was granted a doctoral degree in Civil Engineering with a concentration in Construction Engineering at the University of Tennessee, Knoxville (UTK). He holds a master’s degree in Civil Engineering from Texas Tech University and a master’s degree in Disaster Prevention and Reduction Engineering and Protective Engineering from the University of Chinese Academy of Sciences. His research interests include automation in construction and robotics and sensing.