VIDEO PROCESSING FOR SAFE FOOD HANDLING

by

Chengzhang Zhong

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

School of Electrical and Computer Engineering

West Lafayette, Indiana

May 2021

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Amy R. Reibman

School of Electrical and Computer Engineering

Dr. Jan P. Allebach

School of Electrical and Computer Engineering

Dr. Mary L. Comer

School of Electrical and Computer Engineering

Dr. Mireille Boutin

School of Electrical and Computer Engineering

Approved by:

Dr. Dimitrios Peroulis

ACKNOWLEDGMENTS

First and foremost I would like to express my sincere gratitude to my advisor Professor Amy R. Reibman for her support of my PhD study and research, for her patience, motivation, and enthusiasm. She has taught me the methodology to understand and define research problems at high level. It was a great honor to be her student and a member of our video analytics for daily living lab (VADL).

I would like to say thank you to the rest of my committee, Professor Jan P. Allebach, Professor Mary L. Comer, and Professor Mireille Boutin for their insightful comments and advice.

I am extending my thanks to Professor. Amanda J Deering and Hansel Mina Cordoba from Department of food science at Purdue University. Thanks for offering me so much help in all my data collections for the hand-hygiene projects.

I am also thankful to all my lab mates, Biao Ma, He Liu, Chen Bai and Haoyu Chen, and all the junior lab members Shengtai Ju, Praneet Singh, and Manu Ramesh for their help of my research work, and memorable time we spent in the lab.

Finally, my thanks go to my family for their love and support, especially for giving me the opportunity to pursue my dream.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

GMP	good manufacturing practices
WHO	World Health Organization
CNN	Convolutional Neural Network
STIP	Spatial-time interest point
SVM	Support Vector Machine
HOG	Histograms of Oriented Gradient
HOF	Histograms of Optical Flow
MBH	Motion Boundary Histogram
IDT	Improved Dense Trajectory
LSTM	Long short-term memory
TRN	Temporal Relational Network
GTEA	Georgia Tech Egocentric Activity
ADL	Activities of Daily Living
RNN	Recurrent Neural Network
HMM	Hidden Markov Model
HOI	Human-object interaction
ROI	Region of interest
NLP	Natural Language Processing
UDA	Unsupervised Domain Adaptation
RF	random forest
FPS	frames per second
PV	percentage of units visited
IOU	Intersection Over Union
SGD	Stochastic Gradient Descent
RELU	rectified linear unit
MHI	motion history image
LBP	Local Binary Pattern
HSV	hue, saturation, value

TSN	temporal segment network
QP	Quantization Parameter
SIFT	Scale-invariant feature transform

ABSTRACT

A majority of foodborne illnesses result from inappropriate food handling practices. One proven practice to reduce pathogens is to perform effective hand-hygiene before all stages of food handling. In food handling, there exist steps to achieve good manufacturing practices (GMPs). Traditionally, the assessment of food handling quality would require hiring a food expert to conduct an audit, which is expensive in cost. Recently, recognizing activities in videos becomes rapidly growing field with wide-ranging applications. In this thesis, we propose to approach the assessment of food handling quality, especially hand-hygiene quality, with the video analytic methods of action recognition and action detection. Our approaches focus on hand-hygiene assessment with different requirements, which includes camera view and scenario variance.

For hand-hygiene with egocentric video data, we create a two-stage system to localize and recognize all the hand-hygiene actions in each untrimmed video. In the first stage, we apply a low-cost hand mask and motion histogram features to localize the temporal regions of hand-hygiene actions. In the second stage, we use the two-stream network model combined with a search algorithm to recognize all types of hand-hygiene actions that happen in the untrimmed video. For hand-hygiene with multi-camera view video data, we design a two-stage system that processes untrimmed video from both egocentric and third-person cameras. In the first stage, a low-cost coarse classifier efficiently localizes the hand-hygiene period; in the second stage, more complex refinement classifiers recognize seven specific actions within the hand-hygiene period. For hand-hygiene across different scenarios, we propose a multi-modalities frame work to recognize hand-hygiene actions in untrimmed video sequences. We explore the capability of each modality (RGB, optical flow, hand segmentation mask, and human skeleton joints) at recognizing certain subset of hand-hygiene actions. Then, we construct an individual CNN for each of these modalities and apply a hierarchical method to coordinate all the modalities to recognize each hand-hygiene action in the input untrimmed video.

1. INTRODUCTION

In this thesis, we introduce applications in food handling, which is mainly related the task of hand-hygiene recognition. This chapter introduces the general background of food handling, the importance of applying video monitoring to assist industrial food handling and the potential computer vision tasks. In Section 1.1, we introduce the general concepts of food handling and its connection with video-analytic tasks. In Section 1.2 and 1.3, we introduce the hand-hygiene and produce washing tasks in food handling. In Section 1.4, we list our contributions.

For the remainder of this thesis: Chapter 2 discusses all the relevant topics and works in action recognition research area. In Chapter 3, we introduce our collections of food handling data, for both hand-hygiene and produce washing tasks, with details about camera settings and data pre-processing. In Chapter 4, 5, and 6, we explain in detail our explorations for hand-hygiene action recognition task in three perspectives: ”hand-hygiene in egocentric video”, ”hand-hygiene with multi-camera views”, and ”hand-hygiene in cross-scenarios”. In Chapter 7, we show an additional exploration about the influence of video quality on the general action recognition task. In the last chapter, we summarize the content presented in this thesis.

1.1 General food handling concept

Food safety is a discipline that describes scientific methods to prevent contamination and foodborne illness at different stages of food production. The stages include, but are not limited to: food handling, food storage, equipment cleaning, and staff hygiene. In recent years, where the burden of foodborne illnesses is increasing, evidence indicates that the majority of food contamination is caused by inappropriate food manufacturing practices, involving workers with poor food handling skills [1]. Therefore, we consider video monitoring combined with video-analytic methods for food handling evaluation to be a fast and cost-efficient way to do self-assessment for food growers, processors, and/or handlers.

Figure 1.1 shows some of the general scenes in food handling industries. The events included in daily food handling contains but not limited to: produce packaging, produce washing, and hand-hygiene. Following the standard food handling policy, each of these events requires the food staff to obey restricted processing steps to handle the produce and their personal hygiene. From the perspective of video analytics, assessing the quality of food handling is similar to the task of action recognition or action detection. Each step in food handling can be considered to be an individual action class. And checking a sequence of food handling steps is equivalent to recognizing if a sequence of actions happens in the correct temporal order as well as if it lasts sufficiently long. This fact creates a connection between food handling assessment and video-analytic tasks.

In this thesis, the majority of explorations focus on the food handling event of ”hand-hygiene”, which connects to video-analytic tasks of action recognition and detection. Moreover, we also collected the video data of the food handling event ”produce washing” from a student farm.

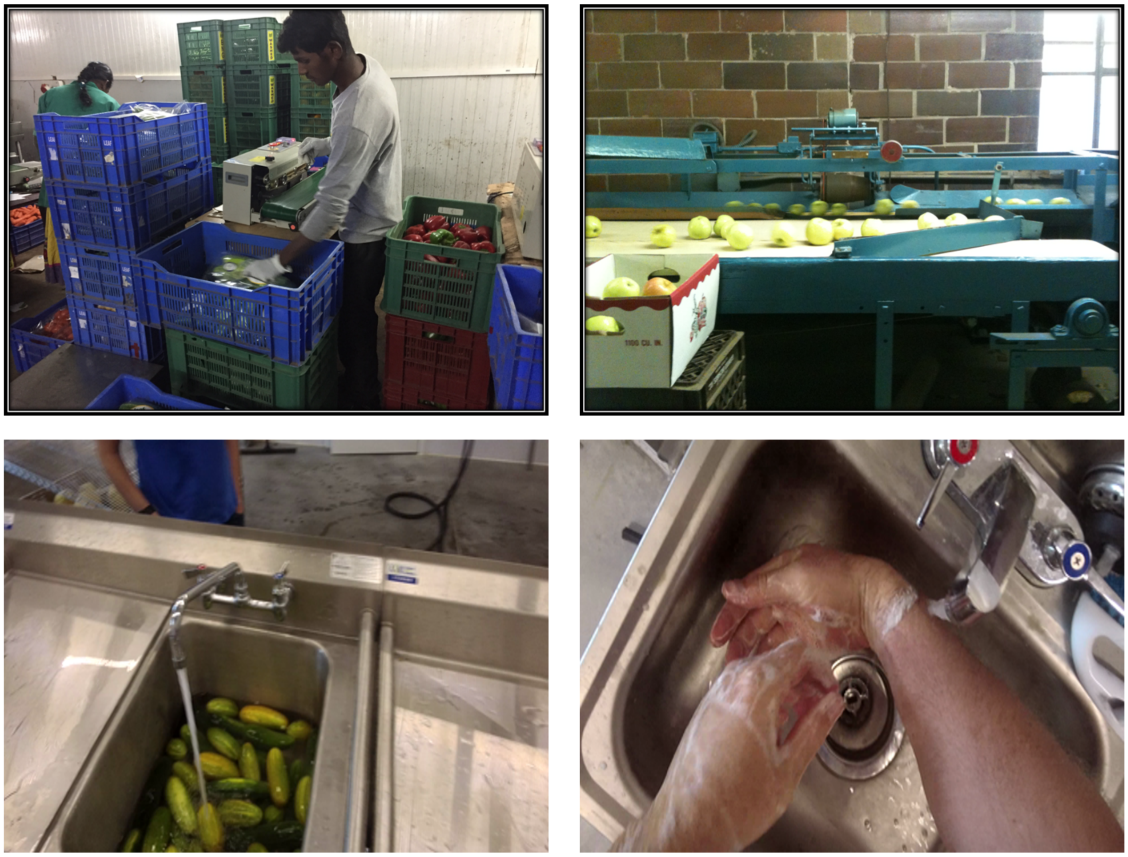


Figure 1.1. Industrial food handling scenes

1.2 Hand hygiene in food handling

In food handling, there are many steps to achieve good manufacturing practices (GMPs). Hand-hygiene is one of the most critical steps. Effective hand-hygiene can reduce food contamination by human pathogens, since this step reduces the likelihood that food handlers harbor pathogenic microorganisms on their hands and transfer them to food products [2]. According to the World Health Organization, there are 12 steps [3] a person should follow to perform effective hand-hygiene. As illustrated in Figure 3.8, the basic steps include: rinse hands, apply soap, rub hands with a variety of different motions, and dry hands. Our goal here was to use cameras to monitor hand-hygiene activities, to automatically identify both positive activities (like those in the figure) and mistakes that either prevent complete decontamination or lead to re-contamination. These mistakes include not rubbing the hands for the required amount of time, touching the faucet with the hands after washing, and not drying the hands.

There has recently been significant progress in automated methods for analyzing video content, a process called video analytics. Stationary cameras placed in a so-called third-person perspective have been used for surveillance, person and vehicle detection and re-identification, activity recognition, and anomaly detection. When recognizing activities of a person, third-person cameras have the advantage of viewing actions from the side. First-person, or egocentric cameras are mounted on the person performing the activity, often on their head or chest [4]. These cameras have the advantage of viewing the person's hands and any objects being manipulated, and are particularly useful to observe subtle hand motions and small objects. However, because they are mounted on a person, these cameras often move chaotically as the person moves. As a result, they may not capture the desired activities, and video processing methods like background subtraction and camera calibration become more difficult [4].

Recognizing activities in videos is a rapidly growing field with wide-ranging applications. Activity recognition techniques have been developed for both trimmed, and untrimmed videos. For trimmed videos, the task is to identify which isolated activity happens in a video clip that has been trimmed to contain only a single activity. For untrimmed videos, the task, which is often termed action detection, is not only to recognize the target action, but also to localize it temporally within the clip that may contain unrelated background actions. This process is termed temporal action localization and is often addressed by identifying temporal action proposals [5]–[7].

In this thesis, we discuss hand-hygiene action recognition in food industry environment with different situations. To simulate the food industry environment, we collect our hand-hygiene dataset in a college bathroom (Chapter 3.5) and a cooking class in Purdue University (Chapter 3.6). For hand-hygiene recognition methods, we start with a simple situation which uses only one egocentric camera to recognize hand-hygiene actions in fixed room scenario (Chapter 4). Then we discuss the advantage of different camera views in hand-hygiene actions, and further extend our method to collaborate multiple cameras for hand-hygiene recognition (Chapter 5). Moreover, we investigate hand-hygiene actions recognition in cross scenarios, and design a multi-modality based system to address this situation (Chapter 6).

1.3 Produce washing in food handling

Beside hand-hygiene actions, produce washing is also one of the most crucial steps for good manufacturing practices (GMPs). In general, produce washing includes but is not limited to the following actions: washing produce, washing containers, storing produce, and hand-hygiene. As a combination of multiple food handling steps, the complete produce washing procedures often last longer than half an hour.

In this thesis, we gathered the data collection of produce washing video in a student farm of Purdue University (Chapter 3.7), which involves multiple food staffs' daily produce washing recorded continuous as high resolution video. However, this thesis does not consider action detection or recognition applied to this dataset.

1.4 Contributions

This section summarizes the contributions of this thesis. There are three hand-hygiene recognition methods we proposed to address hand-hygiene with different situations and a collection of video data with produce washing actions. The contributions are listed as follows:

1. To support the exploration of our video analytic method on industrial food handling, we collect multiple datasets which cover the topic of hand-hygiene and produce washing. For hand-hygiene, our data collection includes college restrooms (Chapter 3.5) and cooking class (Chapter 3.6) environment. For produce washing, our data involves more than 10 hours of collection in a student farm (Chapter 3.7).

2. In the hand-hygiene recognition with egocentric video (Chapter 4), we design a two-stage system to localize the temporal regions of hand-hygiene actions and recognize them in untrimmed hand-hygiene egocentric videos. In the first stage, we extract a low-cost hand mask and motion histogram feature, and process the entire video to localize temporal regions which contain potential hand-hygiene actions. In the second stage, we use the temporal regions detected from the first stage as input. In this stage, we apply a two-stream network model combined with our searching algorithm to recognize all hand-hygiene actions that happen in the input untrimmed video.

3. In the hand-hygiene recognition with multiple camera views (Chapter 5), we first define different levels of tasks for hand-hygiene action recognition, according to their detection difficulty. To explore the influence of camera in hand-hygiene, we compare and evaluate the performance of deep-learning models on three different camera views to recognize trimmed hand-hygiene action video clips. We select the best two camera views for our final system design. As the final system, we propose a two-stage framework to recognize hand-hygiene actions from untrimmed video sequences. We combine two camera views to localize and recognize hand-hygiene actions. Taking advantage of the static third-person view camera, we use a low-complexity CNN model to localize the hand-hygiene period within an untrimmed video. Then we apply more complex CNN models to recognize the actual hand-hygiene action types within these candidate locations.

4. In the hand-hygiene recognition with multiple camera views (Chapter 5), we first evaluate the performance of multiple spatial-temporal action recognition models on same scenario hand-hygiene action recognition and detection task. We also analyze

and summarize the underlying reason for the hand-hygiene recognition model to fail at cross scene recognition. As the final system, we propose to use multi-modalities to create K individual classifiers that collaborate to perform cross scenario hand-hygiene recognition.

2. RELATED WORK

Activity recognition focuses on detecting the events inside a video clip and categorizing each into its activity category [8]. The research topic of action recognition can be extended into many real applications. For example, a surveillance camera placed on the street is capable at monitoring abnormal behaviors. Or automatically edit an untrimmed video of a sport game for highlight video segments. Basically, the expectation behind action recognition is to replace human labors with machine to achieve the same level of action understanding [9]. To achieve this goal, researchers have spent years on developing sufficient models to improve the detection performance. In recent years, the research focus of action recognition is updated from traditional representation and classifier combination into end-to-end trainable convolutional neural networks (CNN). Taking advantage of CNN structure and large scale data collection, the recognition accuracy has been improved significantly.

For food safety area, especially food handling, the major focus is on the interaction between food staff and produces. Similar to the core motivation behind action recognition, the implementation of effectively food handling requires large investigate of human labor as well. Begin with staff training to the final quality test, every steps in food handling could potentially involve hiring more than one food area expert, which is relatively expensive for small food business owner to handle. Therefore, our research is interested in applying action recognition method into food handling field to assistant or even replace human experts. This could potentially provide an alternative way for food business owners to self-audit food safety. Since the majority cost our action recognition was about cameras, sensors, and computers, it has money saving advantage over traditional human experts.

In this chapter, we present the overview of action recognition methods in computer vision and analyze the potential usage of them in food handling.

2.1 Traditional action recognition

Traditionally, action recognition relies on finding hand-craft robust feature representation and combine with machine learning classifiers. Researchers are interested in using a detector to identify key points among video clips and construct salient information around these locations. One of the classic methods is called spatial-time interest point (STIP) [10]. As Figure 2.1, STIP takes use of a Harris point operator to detect spatial-time interest point where the image value has significant changes. Later on, Laptev *et al.* [11] extended their space-time feature into space-temporal bag-of-features. The method constructed a vocabulary of features and mapped features into histograms of visual words. Through combining their new representations with a Non-linear Support Vector Machine (SVM), their experiments were able to achieve a state-of-the-art result on KTH dataset [12], which contains fundamental actions such as walking, running, and hand waving.

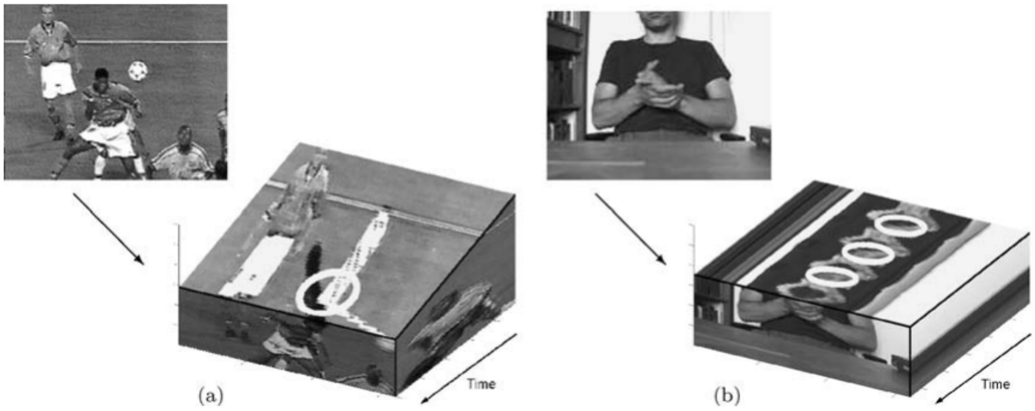


Figure 2.1. Illustration of strong spatio-temporal interest points. Originally shown in [10]

As the number of research work grow in this field, more feature descriptors are proposed to describe appearance and motion information. Histograms of Oriented Gradient (HOG) descriptors [13] was one of the most crucial features used for activity recognition. It was originally proposed for human detection for its efficiency on extracting appearance information. Moreover, Histogram of Optical Flow (HOF) and Motion Boundary Histogram (MBH) [14] are popular features used to extract motion information. As the pipeline bag-of-features became popular, an evaluation of combinations between features and detectors under bag-of-features and SVM setting was presented [15]. The result indicates the Histogram of Oriented Gradients (HOG) and the Histogram of Oriented Flow (HOF) are effective features for activity recognition.

One of the most representative methods that follows this methodology is the Improved Dense Trajectory (IDT) [16], which created track points based on dense optical flow across time, and extracted HOG, HOF and MBH features around these salient points. In order to describe complex action types which might have various durations across time, they trained a codebook combined with a linear SVM classifier to make the final prediction. The algorithm achieved the state-of-art in many third-person data sets.

As indicated in Figure 2.2, the IDT method applies dense sampling to select interest points at different image scales for each frame. Between consecutive frames, a dense optical flow field is used to estimate the motion interest points. The position of each interest point is recorded and updated to construct their trajectories. Along with the shape of each trajectory, HOG, HOF and MBH features are extracted from 3D volumes. A technique of warping between consecutive frames is also applied to eliminate the effect of camera motion.

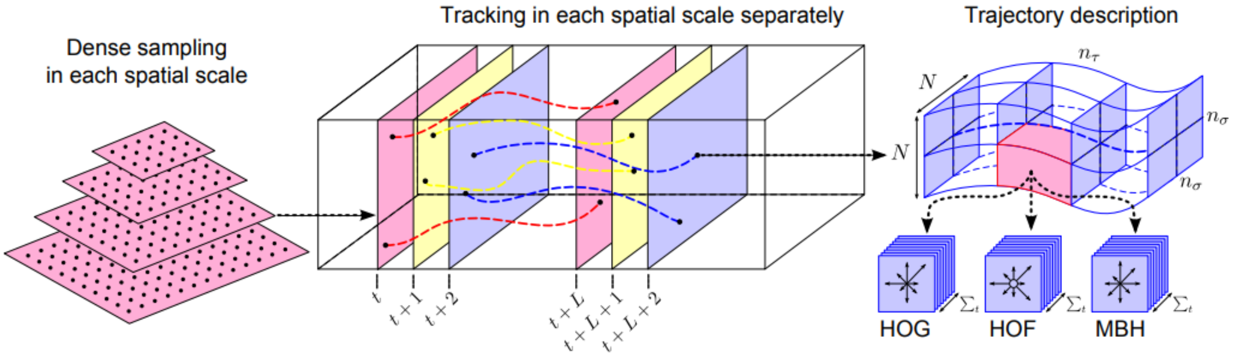


Figure 2.2. Illustration of extraction of descriptors HOG, HOF, and MHI. Originally shown in [17]

Overall, the method achieves good performance among many datasets back to that time. However, it is still difficult for hand crafted feature to be extended into many more different applications.

2.2 Deep learning based action recognition

In recent years, methods based on deep learning have largely replaced these traditional methods. In general, because traditional hand-crafted feature extraction algorithms are designed by human experts, it is difficult to customize them to different applications. In contrast, deep learning methods can overcome this limitation by using sufficient training data to learn ”customized” features. Among all different deep learning models, convolutional neural network (CNN) is the most commonly used structure in computer vision field.

To explain the working principle of CNN, we show a toy example of image classification. In Figure 2.3, the goal is to classify an input image as a shape of ”cross” or ”circle”. A CNN structure will first using different filters to convolve with the input image. Figure 2.3(a) applied the filter shape of ”slash”, ”cross” and ”back slash” to identify the patterns in the input image. From the resulted feature maps, we can observe these patterns are successful recognized and marked with high values. In Figure 2.3(b) the CNN further processing these features with max pooling to reduce the dimension. In the end, feature patterns from filters are ”summarized” by fully connected layer to get a final class label for the input image.

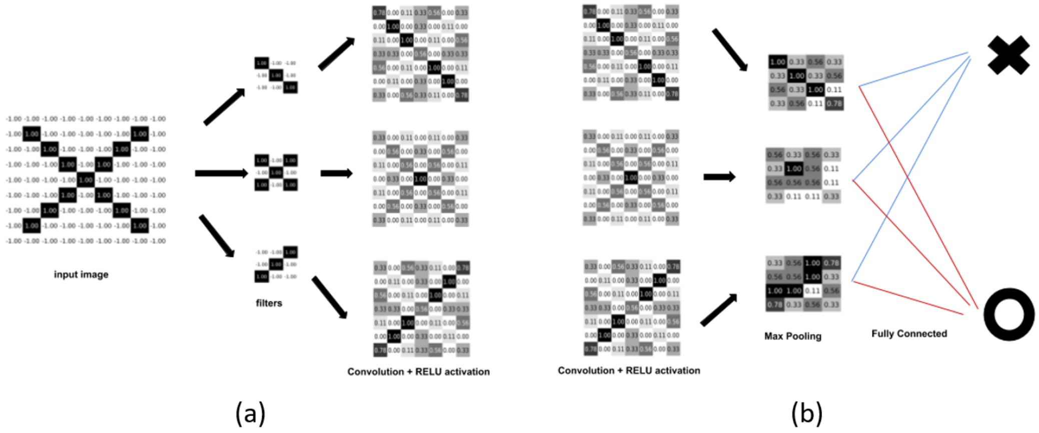


Figure 2.3. CNN convolution and relu activation

The design of convolution kernels filters in combine with non-linear functions and spatial pooling enables CNN to learn shallow features of texture and shape in its beginning layers. As the layer grows, the feature representation generated from previous layers contains encapsulated feature of the entire image. As Figure 2.4, an example of image classification in face class category. The early layers in CNN (top figure) learns body parts such as nose, eye, and mouth. In contrast, the late layer (bottom figure) represent features on the entire face.

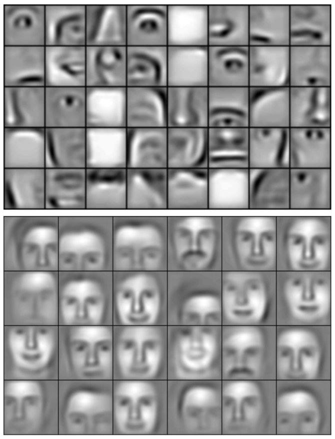


Figure 2.4. CNN face feature representation at different layer. Originally shown in [18]

Due to deep learning based method has more flexibility on its feature construction, with the support of sufficient training data, the performance of deep learning based method exceeds traditional hand-crafted feature in all perspectives. Convolutional neural network (CNN) models, such as AlexNet [19], VGGNet [20], and ResNet [21] has achieved good performance on image classification task. To incorporate temporal information, CNN model can be solely considered as feature extractor to extract spatial information. And combine with temporal models such LSTM [22], BLSTM [23], ConvLSTM [24], and TRN [25]. Moreover, researchers have also developed 3D CNN methods [26][27][28][29] for action recognition. 3D CNN processes video as fixed size input volumes which capable at capturing spatial and temporal information at once.

2.3 Egocentric action recognition

Beside the general action recognition with third person view video, egocentric action recognition also becomes a popular research topic. As portable camera equipment becomes available, researchers have begun to explore action recognition in egocentric videos. Egocentric videos in daily living scenes is explored in [30], using recordings from wearable cameras. Since the daily living scenes considered contain many hand/object interactions, the author proposes to learn an object model that takes into account whether or not the object is being interacted with.

Researchers continue to explore what key information is needed to recognize egocentric activities. One contribution [31] analyzes the important cues to recognize egocentric actions. As shown in Figure 2.5, they experiment with object, motion, and egocentric features and conclude that object cues are crucial clues to classify egocentric actions. This work is influential for pointing out the key factors in egocentric action recognition, which are appearance and motion. The appearance includes hand pose and object. The motion is further separated as head and hand motion. Following the idea of recognizing egocentric objects, Ma et al. [32] create an end-to-end CNN model which embeds hand and object information into one system. The model automatically segments the hand regions and localizes objects close to the hands. By merging these data with motion cues, the model achieves good performance. Moreover, hand poses and motion information is also considered in [33]. A compact EgoConvnet is constructed and fused with a two-stream network. However, most of the methods have been tested using datasets such as Georgia Tech Egocentric Activity (GTEA) [34] and Activities of Daily Living (ADL) [30]. These data sets contain clear object cues. For instance, detecting a dish object close to the hand region reveals a salient clue that the action is dish-washing or eating.

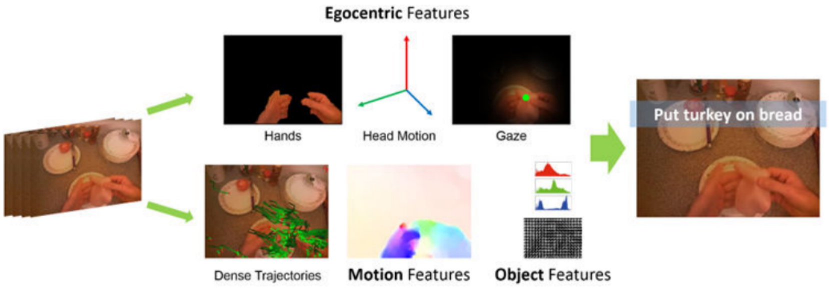


Figure 2.5. CNN face feature representation at different layer. Originally shown in [31]

2.4 Multi-modality action recognition

Beside the most commonly used image or video representation extracted from RGB modality as previous sections, researchers also explore on using other modalities for action recognition. Simonyan *et al.*[35] and Wang *et al.* [36] both applies optical flow modality, which captures pixel level motion information, to combine with RGB for action recognition. As indicated in Figure 2.6, the two stream network not only presents an additional modality for action recognition, but also addresses the concept of spatio-temporal by designing two separate models and merge them for final prediction.

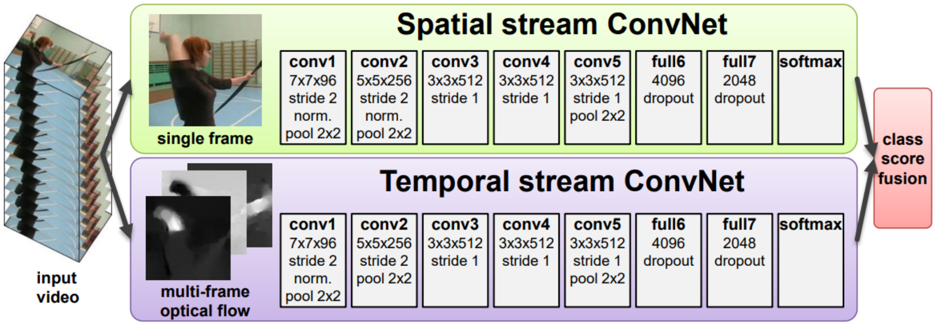


Figure 2.6. CNN face feature representation at different layer. Originally shown in [35]

Moreover, there exists works which apply skeleton joints [37][38][39] as the major modality for action recognition. The skeleton joints can capture the crucial semantic body joints which provides rich information for action recognition as indicated in Figure 2.7. Also, Meng *et al.* [40] intermediately apply skeleton joints as a indicator to spatial discriminative area in the image. This combination of both skeleton joints and RGB modalities also achieve good performance. Thus, the multi-modalities provide additional feature representations which compensates the shortage of RGB modality.

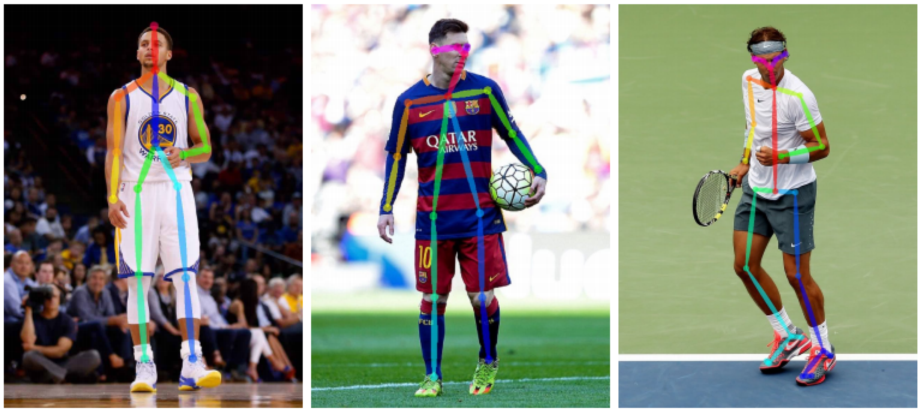


Figure 2.7. Skeleton joints detection on human. Originally shown in [41]

2.5 Pure gesture action recognition

Moreover, in addition to the exploration of actions that involve hand-to-object interaction as egocentric action in Section 2.3, actions that only involve pure hand gestures have also been studied. An important approach in this field is to use multi-modalities instead of solely RGB image as the input to the system. One of the crucial modalities is the skeleton joints, which offer a neat and accurate representation of human pose. A multi-model system was designed in [42] for gesture detection and recognition in the 2014 ChaLearn Looking at People dataset [43]. The goal is to recognize 20 Italian sign languages in a video sequence that is captured using a Kinect camera in third-person view. In their system, one of key steps is to do temporal segmentation by identifying the start and end frame of each gesture candidate. Taking advantage of the Kinect sensor, they used the skeleton joint locations as input features to obtain precise information associate with the hand gesture. Combining with an SVM model, the candidate gesture regions can be localized within an untrimmed video sequence. Another work [44] also illustrated the importance of using skeleton joints in recognizing hand gestures. The work is also targeting at the ChaLearn 2014 dataset with multi-model architecture. One of their modalities is to construct a pose descriptor from the skeleton joint locations to describe the global appearance. Recently, Pigou et al. [45] compared the performance of multiple architectures on gesture recognition datasets; their conclusion is that temporal convolution using a recurrent network achieves the best performance. In addition, their results indicate that depth and skeleton information can help improve the detection accuracy in general. Wu et al. [46] propose a Deep Dynamic Neural Networks (DDNN) that processes skeleton, depth, and RGB images as multi-modal inputs. The structure achieves better performance compared to those that only process a single input. Granger et al. [47] compared hybrid NN-HMM and RNN models for gesture recognition when the system input is only the body pose defined by the skeletal joints.

From the studies above, the importance of skeleton joints in recognizing pure hand gestures without the presence of objects is clear. However, skeleton joints extractions are not always feasible. From a camera perspective, all the videos recorded with a special camera like Kinect can naturally provide human skeleton information. While this type of camera has been used in research, it is still not widely used in practical video monitoring. An alternative approach for gathering skeleton data is to through video processing algorithms. OpenPose [48] and DeepLabCut [49] demonstrate the possibility of detecting 2d skeleton joints on humans and animals. However, depends on the data collection, there will always exist some mis-detections due to occlusion and camera angle constraints.

2.6 Human-object interaction

Human-object interaction (HOI) is also another research field [50][51][52] which focuses on a deeper understanding of action in scene, which is equivalent to a sub-task of action recognition. Instead of considering the entire image area, HOI starts by the localization of objects and human in the scene, and consider both separate object and human and the interaction between them. Through the localization, the model could focus on meaningful region which contributes to the final prediction result.

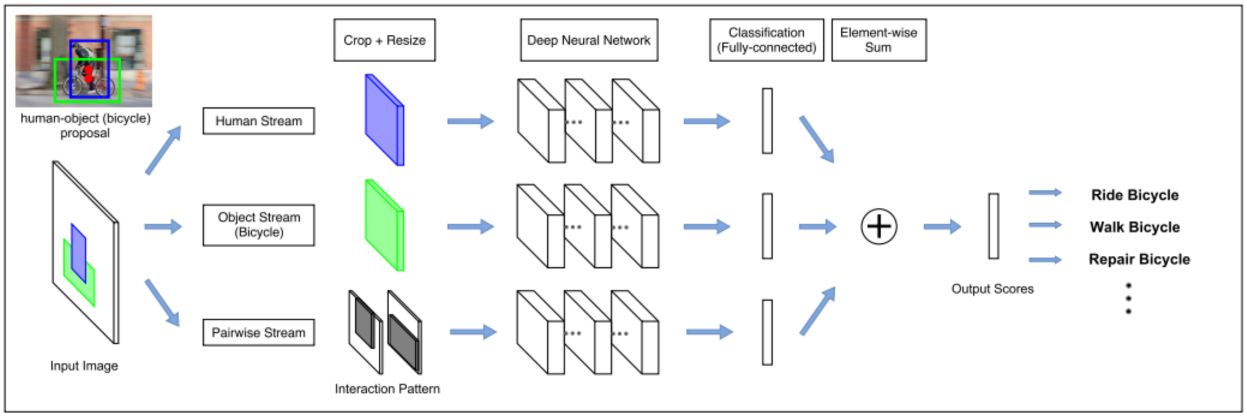


Figure 2.8. Skeleton joints detection on human. Originally shown in [50]

As indicated in Figure 2.8, a person riding bicycle is decomposed in to human, object, and interaction branches. For human and object, their region of interest (ROI) is localized and the corresponding models will only need to process everything within the ROI. This implementation can be understand as a ”hard attention” mechanism, every other object outside of ROI cannot involve into the final prediction, which prevents the irrelevant distraction from influencing the final result. For the branch of interaction, appearance information is completely removed and the interaction is solely represented by the spatial position of human and object.

From my perspective, the importance of HOI is its ”hard attention” idea to individually checking on human, object, and interaction, and summarize the results from all these three branches to predict final result. Compare to methods which ”focus” on entire image spatial region, HOI allows the model to pay attention only on task related ROI. This is extremely beneficial when we have the pre-knowledge on our task. For example, an action of ”play cell phone”, the most discriminative regions have to be on ”human hand” and ”cell phone”.

2.7 Soft-attention in action recognition

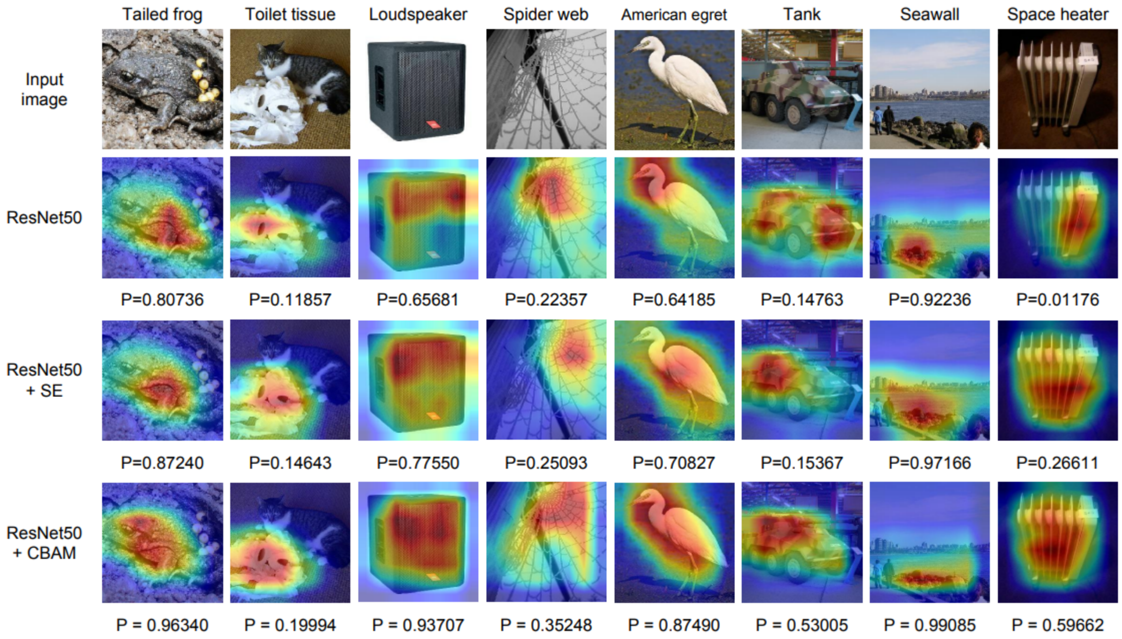


Figure 2.9. Attention heatmap visualization. Originally shown in [53]

The term ”attention” has been widely applied in the field of Natural Language Processing (NLP). In computer vision field, there also exists the discussion of attention and how does it affect the performance of various tasks. Figure 2.9 represents attention visualization for the task of image classification. The heatmaps are generated with GRAD-CAM [54]. The red color in each image covers the most discriminative spatial area contributes to the final prediction result. From top to bottom row, we observe that if the discriminative red area locates on the target object, the CNN model tends to make correct prediction. Because these discriminative regions don't have a certain shape such like rectangle, it is often addressed as "soft-attention". Many of the soft-attention based method [53][55] is attempting to guide the "soft-attention" to focus on the crucial objects in the image.

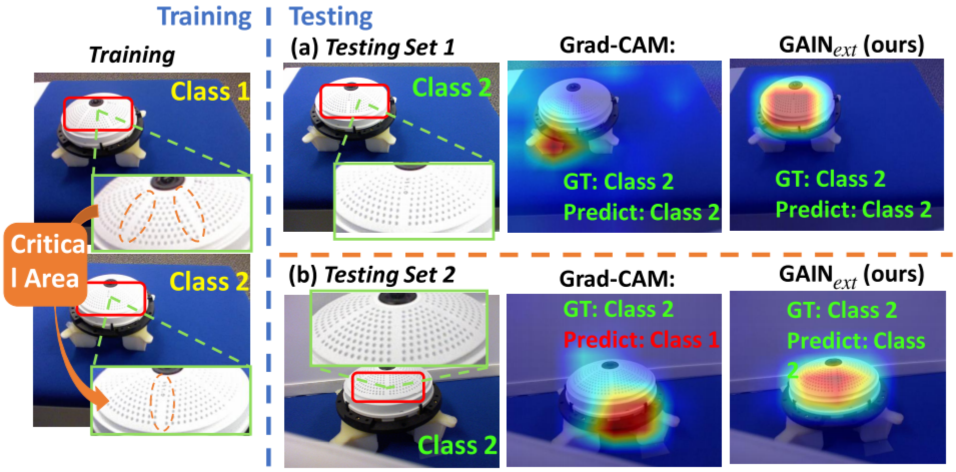


Figure 2.10. Attention influences image classification robustness. Originally shown in [56]

Moreover, the attention of CNN locates on the discriminative image region not only benefits the prediction result under the same data collection, but also improve the robustness when the input comes from unseen data. Li *et al.* [56] in Figure 2.10 demonstrates this idea with a industrial camera orientation classification experiment. The image classification classes are the two different orientations of a camera, which can only be distinguished by the gaps and small markers on the camera surface. With normal CNN model, the attention area locates on the bottom area of camera, which is less discriminative. As a result, this model success on classifying images from the same data collection as its training set. But when the testing data comes from a unseen data collection, the model failed completely. In constrast, with soft-attention based method, the model's attention is adjusted onto the discriminative gaps and markers, which is proved to have robust prediction result even when inferences on unseen data collection.

2.8 Domain adaptation

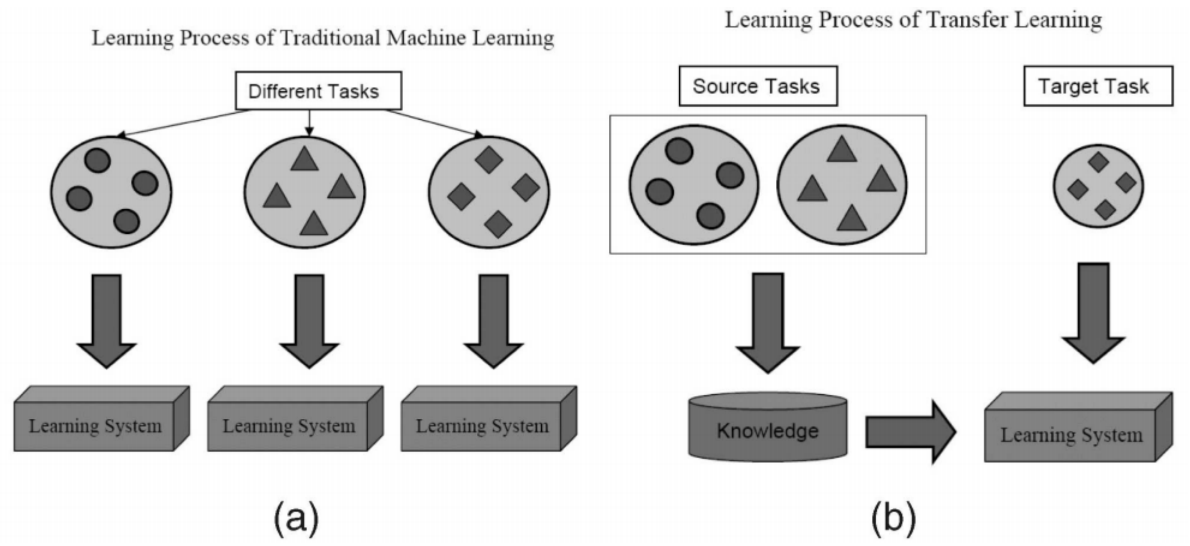


Figure 2.11. Difference in machine learning and transfer learning. Originally shown in [57]

Under ideal situation, the action recognition model assumes the train and test data come from the same data distribution. In reality, it is impossible to collect data from all potential scenarios the model could be deployed to. Therefore, it is unavoidable to have a performance drop if the CNN model is constructed on one scenario and deployed for another. Transfer learning is a research field to address this issue. As shown in Figure 2.11, the goal of transfer learning is to apply existed source data to solve target task with unseen data. A sub-topic related to computer vision is called domain adaption. In general, different scenarios can be addressed as different domains, and the goal is to construct a model on source domain data and apply to target domain. In the field of domain adaptation, there exists a sub-area called Unsupervised Domain Adaptation (UDA) that further assumes that target domain only has unlabeled data. The UDA is more related to realistic application with limited target domain availability. Researchers have developed works on UDA [58][59][60][61] for image classification task. Recently, the works [62][63][64] in UDA are extended to video classification task as well.

2.9 Action detection

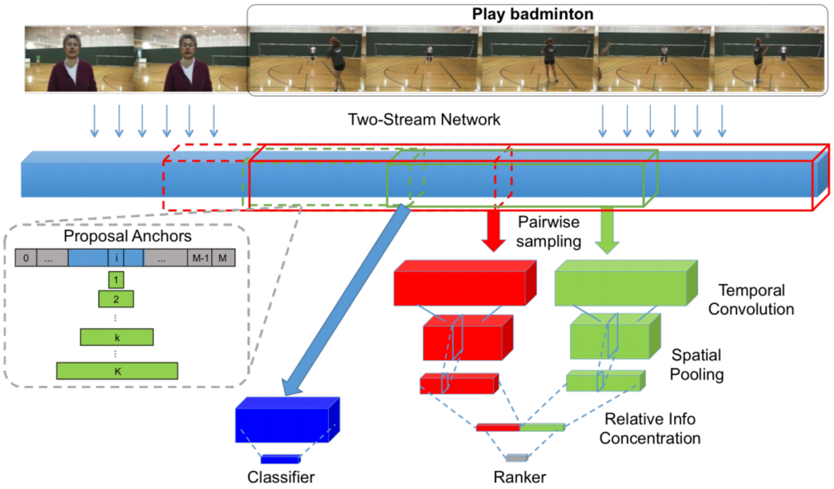


Figure 2.12. Pipeline of action detection with temporal proposal. Originally shown in [6]

Action recognition focuses on recognizing class category from a trimmed video clip, which includes one action. In reality, a regular video is high likely to include more than one action, and the start and end time of each action in unknown. Thus, researchers defined a task named ”action detection” to process on untrimmed video. The goal of action detection is not only to recognize all actions, but also localize each action in temporal. One typical approach is inspired [7][6][65][66] from object detection to use temporal proposal in action detection like Figure 2.12. Another approach [67][68] applies temporal convolution to build encoder-decoder structure to solve this problem.

3. DATASET

The goal of our study is to explore hand-hygiene actions in food handling procedures. However, there is no currently available dataset for this type of actions. To overcome the video data issue, we propose to create a hand-hygiene dataset to help explore video analytic methods for hand-hygiene action recognition. To throughly explore the how different camera angle might affect hand-hygiene recognition. We apply both egocentric and third person camera as our primary camera angle selection. In Chapter 3.1, we introduce the selected publish available datasets in egocentric and third person camera view. In Chapter 3.2, we introduce the camera settings in our data collection. In Chapter 3.3, we define three different levels of hand-hygiene tasks. In Chapter 3.4, we discuss our exploration on hand-hygiene in home scenario. In Chapter 3.5, we introduce our ”nelson100” hand-hygiene dataset collected in college bathroom. In Chapter 3.6, we introduce our ”class23” hand-hygiene dataset collected in college cooking class. In Chapter 3.7, we introduce our data collection of produce washing in a student farm of Purdue University.

3.1 Reference other action recognition datasets

3.1.1 Egocentric video dataset

In this section, we talks some of the current public available egocentric datasets, which we go through design purposes of these datasets and how they choice their camera settings.

Pirsiavash *et al.* [30] created a Activity for Daily Living (ADL) dataset. The dataset includes 20 subjects to perform a list of 18 daily activities, which includes but not limited to making food, laundry and using computer. They used one chest mounted GoPro camera to record 1280x960 resolution, 30 FPS egocentric video with wide angle. Each video last around 30 minutes.

Fathi *et al.* [34] collected the GTEA dataset with 7 daily activties performed by 4 subjects. These activities are all food making related and each of them contains a sequence of sub-actions about hand-to-object interaction. The camera they selected is GoPro mounted on a baseball cap. All the videos has 1280x720 resolution, recorded under 30 FPS.

Figure 3.1. Activity for Daily Living (ADL) dataset

Figure 3.2. GTEA dataset

Comparing with the above datasets, Lee *et al.* [69] created UTE dataset with more generalized daily activities types. The videos were not limited to in-door daily activities, but also out-door activities such as driving, shopping and attending lectures. Because of their special demands of long-time recording, the camera they used for video recording is Looxcie wearable camera. They exist 10 videos from 4 subjects in this dataset, each video in this dataset last for 3 to 5 hours with 320x480 resolution, 15 FPS.

Figure 3.3. UTE dataset

Besides using single egocentric videos, the CMU-MMAC databse [70] used 5 static cameras and 1 wearble cameras to record cooking activities. The wearable camera has a high resolution of 800x600/1024x768, 30 FPS.

Figure 3.4. CMU-MMAC dataset

From all these datasets listed above, we can conclude that egocentric videos are encouraged to be recorded under high resolution. The mount location of egocentric camera can be either head or body. If the activity requires the wearer for long time or secret recording, head mounted camera is preferred. Otherwise, body mounted camera is a good option.

3.1.2 Third person video dataset

Moreover, we also explore some third person action recognition datasets and reference their camera settings for data recording.

The UCF101 dataset [71] includes total number of 101 action and 13320 clips. All clips have fixed frame rate and resolution of 25 FPS and 320 × 240 respectively.

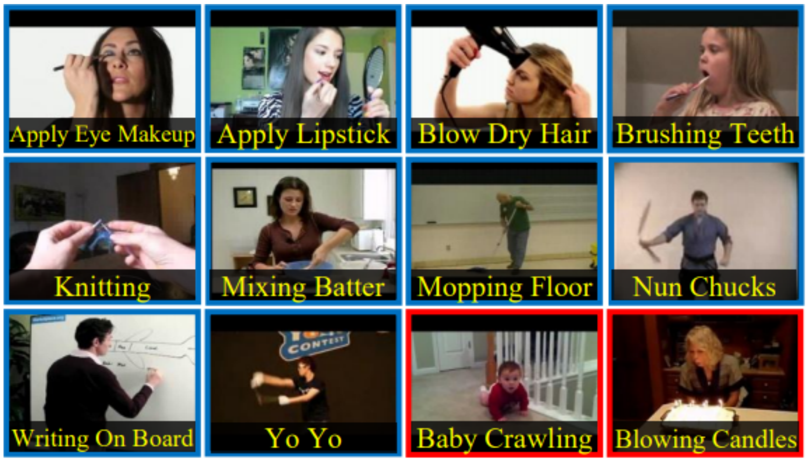


Figure 3.5. UCF101 dataset

The HMDB51 dataset [72] contains 51 distinct action categories. Each category has at least 101 clips. And the total number of video clips is 6766. Within each clip, the frame's height is scaled to 240 pixels and the width is scaled accordingly to maintain its aspect ratio.

For all these third person datasets, they include a large amount of video collections, which are gathered from youtube, or movies. And majority of these videos is focusing on sport, which a human takes use of its entire body to perform an action. Because of the original video quality and storge reason, all videos in these datasets have a relatively low resolution.

For our hand-hygiene data, the action is majorly focusing on the upperbody of human. Thus, it is necessary for us to record these actions with high resolution video, where motions from arms and hands can be clearly visualized. Also, human involves in hand-hygiene actions will not move with high speed, which support us to record with static camera at fixed

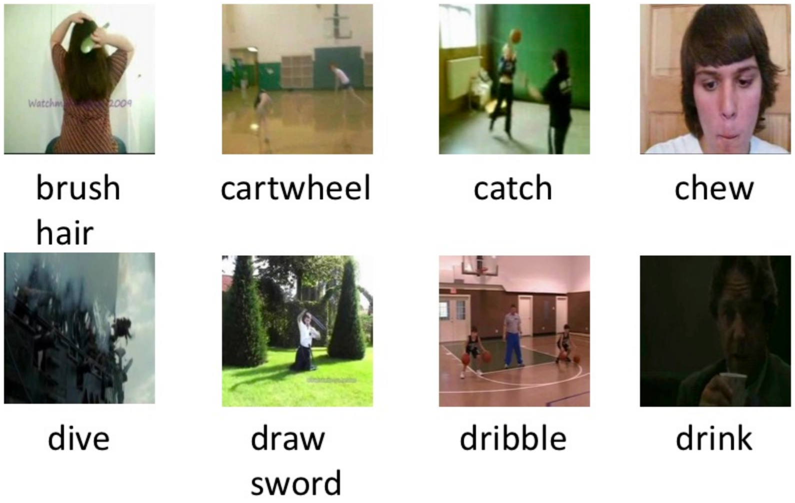


Figure 3.6. HMDB51 dataset

position. Moreover, due to the limitation of resource availability, we are not able to record large scale video as these datasets. High resolution videos in our dataset might not cause storage issue compare to their datasets.

Therefore, if we plan to record hand-hygiene action with third person view camera, we are encouraged to record with static, high resolution third person view camera.

3.2 Cameras

According to hand-hygiene techniques from WHO [3], the duration of the entire hand-hygiene procedures lasts 40-60 seconds, which is relatively short compare with daily activity recognition. This allows us to record high resolution videos as well as maintaining efficient camera battery. Based on this idea, we select to use GoPro hero 6 as our egocentric camera. GoPro hero 6 is able to record high quality videos with 1080p resolution under 30 FPS.

Table 3.1. Camera settings

Camera name	Camera type	Video quality
Chest camera	GoPro Hero 6	1080p, 30 FPS
Nose camera	IVUE Rincon	1080p, 30 FPS
Wall camera	GoPro Hero 6	1080p, 30 FPS

Besides the camera video quality, the location to mount a camera on the wearer is also important in video recording. As we discussed in Section 3.1.1, if the camera wearer doesn't record long-duration video, the body mount of a camera can provides stable camera views. However, we also argue that the head mount camera provides unique information during hand-hygiene procedures. For example, if the camera wearer rubs his/her hands while turning his/her head to talk to another person, the body camera may not capture this information effectively. On the other hand, a head mounted camera can capture the whole talking scene, which shows the camera wearer was distracted during hand-hygiene actions.

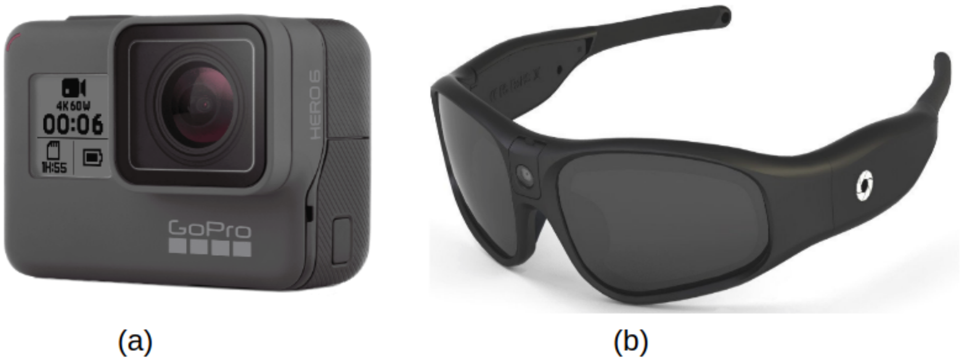


Figure 3.7. (a) GoPro hero 6 (b) IVUE camera

Based on these considerations, we design two different mount options for egocentric cameras, in Figure 3.7, on a subject's chest and head during video recording. The chest camera is a GoPro hero 6 with chest harness. Considering of the recording efficiency and comfort, we use IVUE camera to mount on the wearer's nose as the head camera. The IVUE camera works the same as wearing a glasses and provides high video quality of 1080p under 30 FPS.

Even though hand-hygiene actions concentrate on the interactions between hands which can be clearly recorded through egocentric camera view, we would still like to investigate the utility of applying third person camera view in this type of actions. Therefore, we use another GoPro camera to mounted on a static position near the subject during the hand-hygiene procedures. The camera records the same video quality as the two egocentric cameras.

3.3 Definition of hand-hygiene action and task levels

To the best of our knowledge, we are the first project to apply video analytic methods on hand-hygiene actions. Therefore, we want to define the different types of hand-hygiene actions and the purpose of analyzing these actions. We defined three different video analytic tasks to explore in hand-hygiene actions. Based on the level of difficulties of these tasks, we define them as *detail-level*, *standard-level*, and *detection-level* tasks.

According to the World Health Organization, there are 12 steps [3] a person should follow to perform effective hand-hygiene. As illustrated in Figure 3.8, the basic steps include: rinse hands, apply soap, rub hands with a variety of different motions, and dry hands.

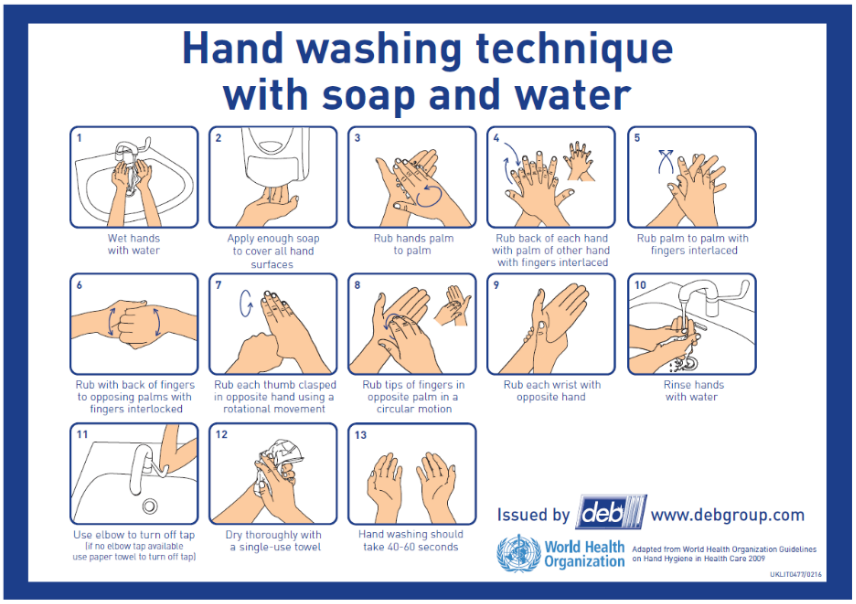

Hand washing technique
with soap and water
1
Wet hands with water
2
Apply enough soap to cover all hand surfaces
3
Rub hands palm to palm
4
Rub back of each hand with palm of other hand with fingers interlaced
5
Rub palm to palm with fingers interlaced
6
Rub with back of fingers to opposing palms with fingers interlocked
7
Rub each thumb clasped in opposite hand using a rotational movement
8
Rub tips of fingers in opposite palm in a circular motion
9
Rub each wrist with opposite hand
10
Rinse hands with water
11
Use elbow to turn off tap (if no elbow tap available use paper towel to turn off tap)
12
Dry thoroughly with a single-use towel
13
Hand washing should take 40-60 seconds
Issued by deb www.debgroup.com
World Health Organization
Adapted from World Health Organization Guidelines on Hand Hygiene in Health Care 2009
UKLIT0477/0216

Figure 3.8. Standard hand-hygiene steps.

Our goal here was to use cameras to monitor hand-hygiene activities, to automatically identify both positive activities (like those in the figure) and mistakes that either prevent complete decontamination or lead to re-contamination. These mistakes include not rubbing the hands for the required amount of time, touching the faucet with the hands after washing, and not drying the hands. In the following sub-sections, we define different hand-hygiene tasks, where each task represents one level of difficulty and includes an individual set of hand-hygiene actions.

3.3.1 Detail-Level Hand-Hygiene Task

In the detail-level task, the goal is to strictly follow each of the steps outlined by the World Health Organization in Figure 3.8; did a participant perform each of the 12 steps?

This detail-level task has the highest difficulty compared to other hand-hygiene tasks, especially for those actions that involve subtle motions of the hand and fingers, such as those illustrated in Figure 3.9a. As indicated in Figure 3.9b,c, the egocentric camera does not always capture the entire hand regions, because participants have different body size and personal habits. Therefore, it is inevitable that hand regions may be missing during the detailed hand-hygiene steps.

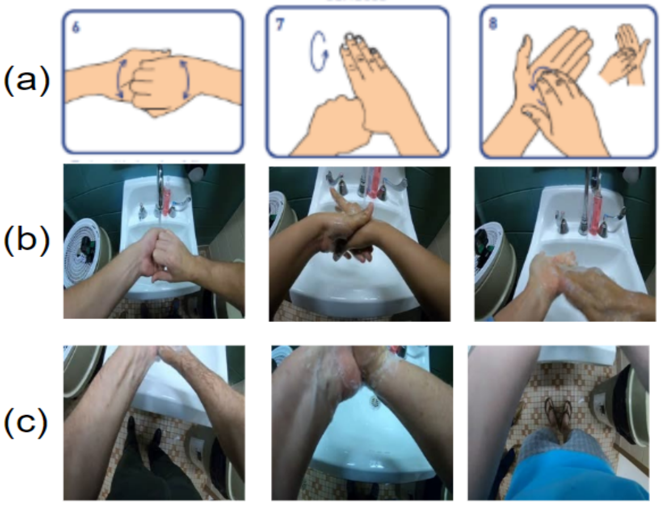


Figure 3.9. (**a**) Standard subtle actions between fingers. (**b**) Clear view of subtle actions. (**c**) Hands out of camera views.

Even if the entire hand regions are clearly recorded, recognizing actions with subtle finger and hand motions is still difficult. The temporal boundaries between actions of rub cross finger, rub palm, and rub thumb are difficult to distinguish even by a human expert. The method in [73] to recognize dynamic long-term motion is only likely to be able to recognize the entire action sequence. To accurately localize the boundary between these similar actions, we may need to apply an RGB-D sensor and construct hand-finger skeleton models [74].

3.3.2 Standard-Level Hand-Hygiene Task

In the standard-level hand-hygiene task, we focus on analyzing only the components of the 12 steps in Figure 3.8 that are most critical from a food-safety hand-hygiene perspective. As a result, we define the standard-level hand-hygiene task to distinguish among the 7 types of actions shown in Figure 3.10: touch faucet with elbow, touch faucet with hand, rub hands with water, rub hands without water, apply soap, dry hands, and a non-hygiene action. Essentially, the six rubbing actions are condensed into a single rubbing action, and we retain the critical components of applying water and soap, rubbing for an extended period, and drying the hands. In addition, this task includes identifying the action "touch faucet with hand", which must be avoided to prevent re-contamination after the hands have been rinsed with water [3].

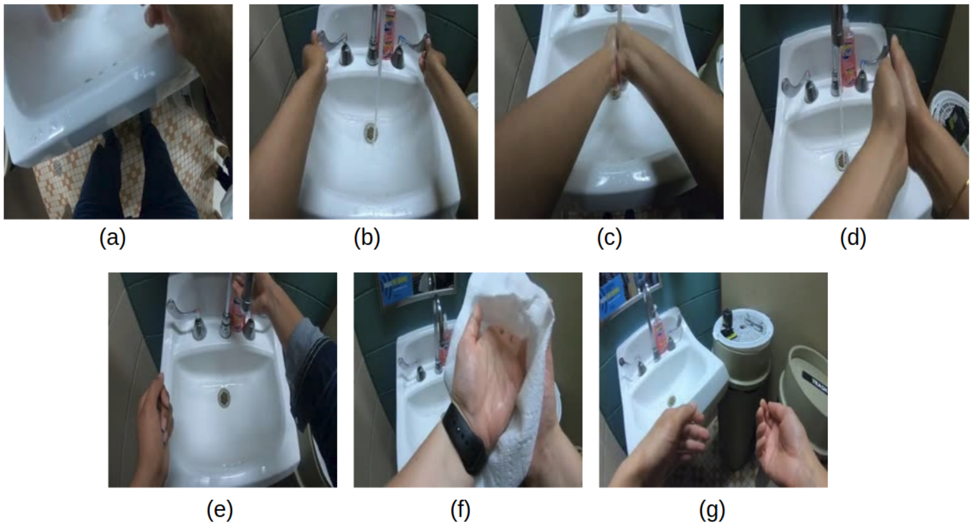


Figure 3.10. (**a**) Touch faucet with elbow, (**b**) touch faucet with hand, (**c**) rub hands with water, (**d**) rub hands without water, (**e**) apply soap, (**f**) dry hands with towel, (**g**) non-hand-hygiene action.

As mentioned above in Section 3.3.1, the subtle hand and finger motions may not be completely recorded for a variety of reasons. The standard-level task removes the need to distinguish among the subtle hand and finger motions, so a classifier for these 7 action classes will be more robust than for the detail-level task, both with respect to the variations of a participant's body size and to the hands not appearing in the camera view

3.3.3 Detection-Level Hand-Hygiene Task

In addition to the previous two task levels we defined, there exists a detection-level hand-hygiene task, which simply analyzes whether or not the hand-hygiene action happened. For this task, egocentric camera information regarding the hands is not necessary, and directly analyzing the third-person camera should achieve an acceptable result.

3.4 Hand-hygiene under home scenario

For many public available egocentric datasets, the recording environments are inside each subject's home apartment, especially for daily activity videos. For our hand-hygiene dataset, we need to decide what environments should we select to record our data.

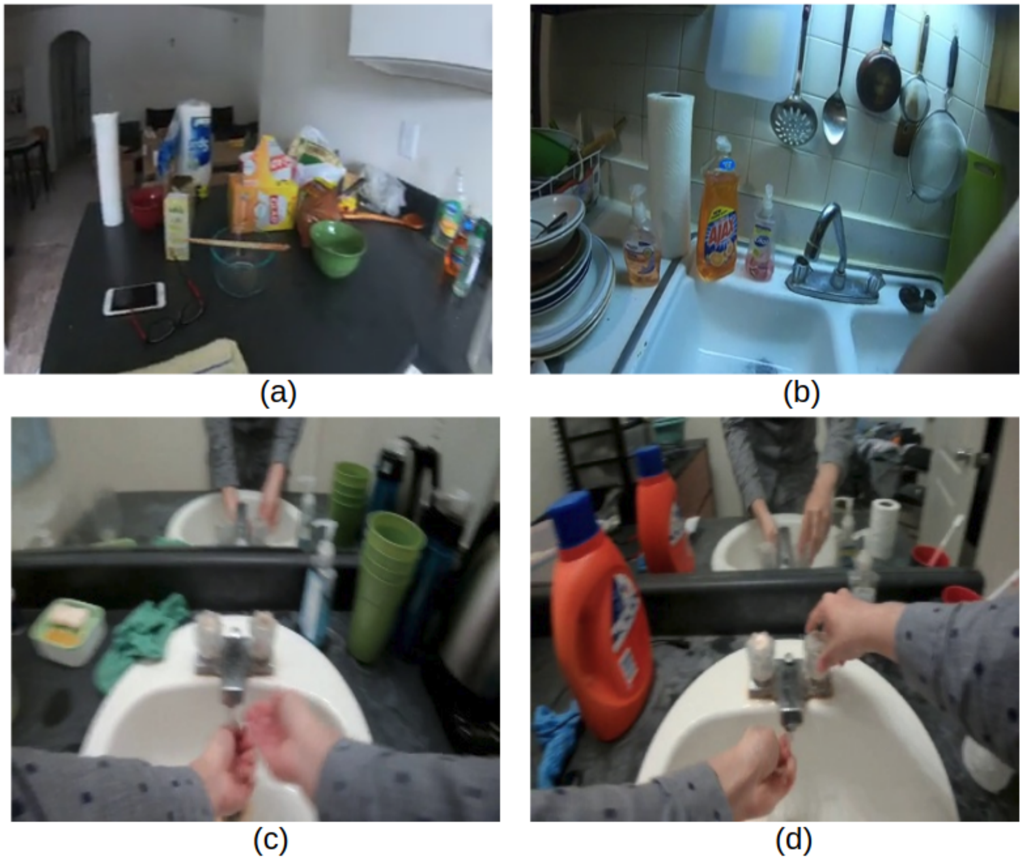


Figure 3.11. (a),(b) Objects in kitchen (c),(d) mirrors in bathrooms

We start trying our video recording at different subjects' home apartments. The locations of performing hand-hygiene actions are either in the kitchen or bathroom (Figure 3.11). However, we found the environments of each apartment varies a lot. There usually exists many cooking tools or food in the kitchen environment. These objects are different in types and most of them shouldn't appear at a standard industrial food handling factory. Moreover, videos recorded in apartment bathrooms usually include mirrors. The hands actions reflected in the mirrors are also recorded by cameras. This fact increased the video processing difficulty to correctly recognize hand actions. And automatic removal of mirror disturbances is not our priority target in this stage. Because of these reasons, recording inside home apartments is not a good choice to create our dataset.

We also needed to decide the number of participants for our study. Many egocentric datasets were constructed by few subjects repeat each activity several times. We attempted to record only few subjects and asked each subject to record more than 10 times for the whole hand-hygiene procedures.

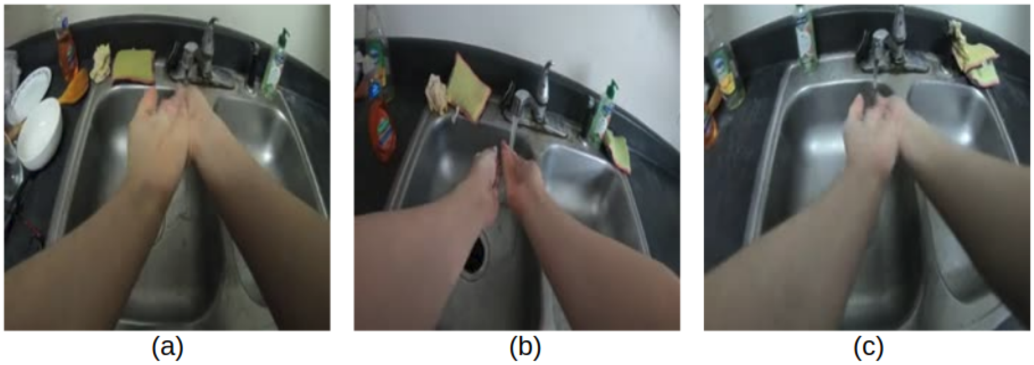


Figure 3.12. (a),(b),(c) rinse hand action repeated by one subject among 3 times

However, every subject has their own style of hand-hygiene actions. And a subject may not behave differently between each time of hand-hygiene. As it is indicated in Figure 3.12, a subject was doing hand rinsing among 3 different times of recording. In each time, the subject repeated for the same sequence of subtle motions. If we ask only few subjects to repeat multiple times hand-hygiene actions in their own styles, our dataset might contain too much repetitive hand-hygiene patterns. This is not an appropriate method to explore hand-hygiene actions.

Base on the discussion above, we decide to record our hand-hygiene dataset under a fixed, clear environment with large amount of subjects.

3.5 Hand-hygiene in college bathroom

3.5.1 Data collection

As the area of egocentric video becomes popular, researchers have published datasets [30], [34] for evaluating the performance of different egocentric action recognition methods. Many publicly available datasets involve only a few participants with recordings done inside home apartments. However, there exists significant differences between a home kitchen and an industrial food handling facility. Moreover, every participant has their own style of hand washing. We believe our data will generalize better if we involve more participants.

To ensure that our dataset includes enough variation between samples, we invited 100 participants and recorded the videos in two separate public restrooms with similar environments (All data collection took place in August 2018 and was done within the context of Purdue IRB # 1804020457.). All participants were allowed to wear any type of clothing, including watches and hand jewelry, and had varied ages, genders, and races. Each participant was recorded twice while they washed their hands. Initially, each participant performed a naive hand washing in the first room, according to their typical hand washing style. Then, the participant was asked to read the instructions for hand-hygiene shown in Figure 3.8. Finally, each participant was recorded washing their hands again in the second room. When the data were collected, all participants indicated their willingness to have their data published. However, not all participants agreed to a public dissemination of their data; therefore, our hand-hygiene dataset will not be made publicly available at this time.

Our overall goal was to design a method that will operate in any indoor environment, including temporary environments with portable wash-stations. A single camera mounted on the ceiling created a top-down view for the previous work in [75], [76]. However, the layout may not be consistent across all indoor environments; for example, the location of faucet, the height of the ceiling, the location (or existence) of a mirror, may all be different for different environments. Therefore, instead of designing a particular camera installation plan for every potential environment, we choose to collect our data in “portable” way. Thus, we use cameras that can be easily mounted on a participant or easily moved from one location to another.

Two type of cameras, egocentric and third-person, are applied in our data collection. An egocentric camera is capable of capturing subtle motions of hands and fingers, which provides supportive information to classify different types of hand actions. In contrast, a third-person camera is efficient at capturing a participant’s body motion as well as any interaction with the surroundings.

To explore the efficiency of third-person video and egocentric video in hand-hygiene actions, we used both egocentric cameras and a static third-person view camera for video recording. Each participant wore a GoPro camera with a harness on their chest as one egocentric camera and an IVUE glasses camera on their nose as another egocentric camera. The third-person view camera was a GoPro camera placed on top of a flat platform near the sink. We will refer these three camera views as “chest camera view”, “nose camera view”, and “wall camera view” for the rest of this paper. Each video has 1080p resolution, 30 FPS, and a wide viewing angle. A visualization of of this dataset can be found at Figure 3.10, which is recorded under egocentric chest camera view.

The entire dataset is recorded in Purdue University, Philip E. Nelson Hall of Food Science with 100 participants. We name this dataset as ”Nelson100” hand-hygiene dataset.

3.5.2 Data labeling and availability

Frame-level labeling In our ”Nelson100” dataset, all the hand-hygiene actions are labeled by human expert at a frame level for all three camera views. Due to the limited labeling resource, ”Nelson100” dataset is only labeled for standard level hand-hygiene task, which includes 6 types of hand-hygiene actions: ”touch faucet with elbow”, ”touch faucet with hand”, ”rub hands with water”, ”rub hands without water”, ”apply soap”, ”dry hands with towel”. Beside these hand-hygiene actions, all the other actions performed by participants are labeled as ”non-hygiene” actions. Therefore, the ”Nelson100” includes 7 types of actions in total.

Data availability The dataset consists video data to support both action recognition and detection tasks. For action recognition task, each of these hand-hygiene or non-hygiene actions are considered as ”trimmed video clips”, which will be applied to the task of action recognition. For action detection task, the videos which includes all actions of a person’s start recording to the end of recording, we refer these videos as ”untrimmed video” which will be applied to the task of action detection. Table 3.2 lists the number of trimmed video clips for each camera view. The train, validation, and test action sets are split with ratio 0.66 : 0.12 : 0.22 based on number of participates, where each participate might contribute different number of trimmed video clips. Video clips include invalid information due to inappropriate recording are deleted. For the number of untrimmed video, There are 44 of untrimmed videos each camera view.

Table 3.2. Number of trimmed video clips for each camera view: out of parentheses hand-hygiene actions only; in parentheses: hand-hygiene and non-hygiene actions

Camera	Train	Validation	Test
Chest	947 (1357)	144 (208)	307 (427)
Nose	947 (1358)	144 (208)	307 (427)
Wall	942 (1314)	144 (202)	307 (427)

3.5.3 Potential challenge of the dataset

Hand-hygiene actions are unique due to their specialty in viewing angles and subtle motions. Hands reaching out of the camera view is one of the problems that usually occurs in hand-hygiene action videos, especially if the video is recorded under egocentric view. Figure 3.13 shows an example of the out-of-view problem during hand-hygiene. The camera was mounted on the actor’s chest with a harness. Because of the difference in people’s body shape and height, it is difficult to track the actor’s hands all time. For instance, in Figure 3.13(a), the actor’s is much higher than the washing sink. Therefore, when he bowed to approach the sink, the camera view was shifted to record his legs. Moreover, in Figure 3.13(d), when the actor's height is close to the washing sink, it is easy for him/her to put the arms in front of camera when rubbing hands.

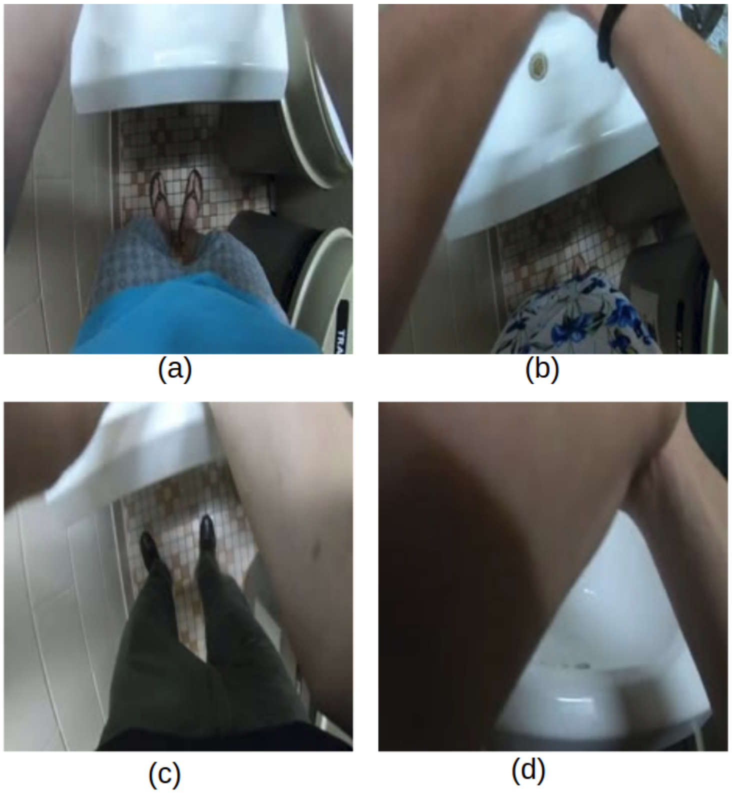


Figure 3.13. Hand-hygiene actions occluded in egocentric view

The illumination condition is also an important factor for egocentric videos. For our hand-hygiene videos, we maintained the lighting condition at the same level for all the participants. However, because different participants have different heights, when the camera view gets close to the sink, the shadow of arms became explicitly recorded. For example, in Figure 3.14, the arms' shadow appeared on white sinks. These shadows have impact on the motion computation in video processing, which can potentially disturb the detection results.

The most challenging part in hand-hygiene action recognition is to detect the subtle motions, especially in the pair of action rinse hands and rub hands. It is required by hand-hygiene procedures to rub hands with different poses. To detect the strength used in hand rubbing, we defined two action classes: rub hands and rinse hands. Rub hands describes that

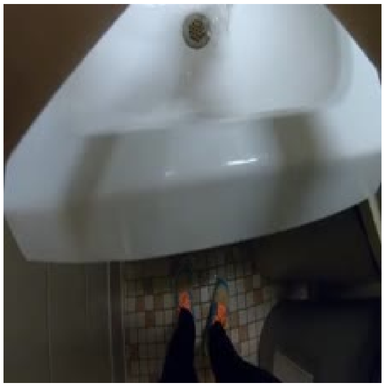


Figure 3.14. shadow of the arms on the washing sink

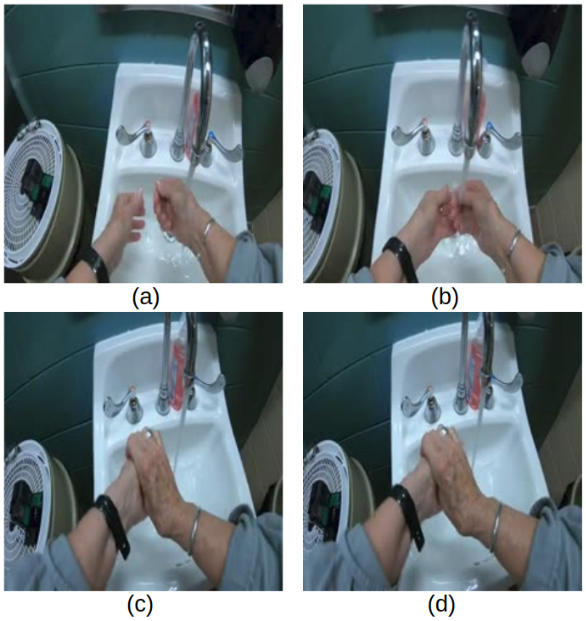


Figure 3.15. (a),(b) rinse hands action (c),(d) rub hands action

the participant use quite a mount of strength to rub hands without pose limitations. On the other hand, rinse hands describes that the participant uses little strength at hand-to-hand interactions. Normally, the participant will only let the water flow through his or her hands.

However, in the real scenario, the participant normally switches between the actions of rub hands and rinse hands in a short time period, which makes it difficult to identify each action. Moreover, the appearance of rub hands action and rinse hands action are very similar, which causes confusion when using appearance information from these actions.

3.6 Hand-hygiene in food class

Beside our ”nelson100” dataset introduced in Section 3.5, we extend our exploration into a food class in Purdue University. In this section, we introduce our new ”Class 23” hand-hygiene dataset. We first introduce our video collection steps and data pre-processing. Then, we define the hand-hygiene task which the dataset targets at and the related adaptations for this dataset. Finally, we describe the ground truth labeling and data availability for action recognition and detection tasks.

3.6.1 Data collection

As we introduce in Section 3.1, the published available food related datasets have a different focus than our food safety research. Georgia Tech Egocentric Activity (GTEA) [34] and Activities of Daily Living (ADL) dataset [30] collected a dozen of people’s daily activities in home scenario. MPII cooking activity dataset [77], 50salads dataset [78], and Breakfast Actions Dataset [79] recorded food handling and cooking in home scenario. However, professional food safety facilities share a different scenario than the general home kitchen. To simulate the food safety environment, we collected 100 participants perform hand-hygiene in college bathrooms as our ”nelson100” dataset. In this section, we extend our data collection scenario into professional food handling laboratory.

In our new data collection, we invited 23 students who participate a cooking class of Purdue University to participate in our data collection. Therefore, we name the dataset as ”class23”. In the cooking class, students are required to perform hand-hygiene before they start the lab portion of the course. To reduce waiting time, students are split into two groups to do hand-hygiene in two different rooms, namely ”room1” and ”room2”. Each room has one hand-hygiene sink, and students line up to wash hand one after another. All students follow the strict laboratory policy to wear a lab coat and a bouffant cap during the entire lab section. Students are aware of standard hand-hygiene procedures. During the video recording, students perform hand-hygiene by themselves without the supervision or disruption from the instructor. Therefore, ”non-hygiene” behaviors such as ”talking to each other” and ”walking around the room” are also captured in our data collection.

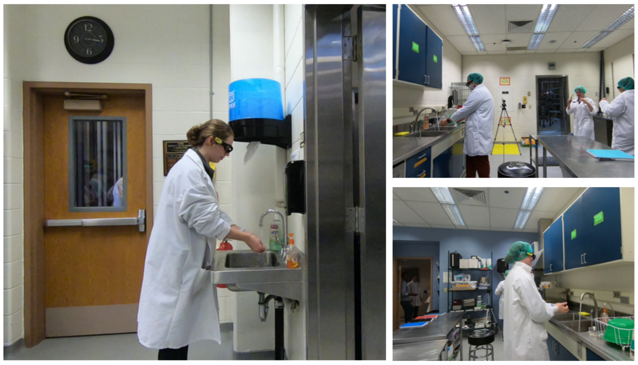


Figure 3.16. Data image; (a) Left: room1 camera1. (b) Right top: room2 camera1. (c) Right bottom: room2 camera2

The two rooms have a distinct layout; the location of the sink and the configuration of objects around the sink are different in each room. This affects our camera placement in each room. Figure 3.17 indicates the layout of ”room1”, where the sink is located at the corner of laboratory. Thus, we can only place the camera on one side of the sink for video recording. The angle between ”sink to camera” and ”camera to human” is approximately 90 degree. However, as indicated in Figure 3.18, the sink in ”room2” is inset into a countertop, which prevents us from placing the camera at 90 degree as in ”room1”. The two cameras in ”room2” are placed on each side of the person (labeled ”human” in the figure). The angle between ”sink to camera” and ”camera to human” is about 70 degree. A frame from each resulting video is shown in Figure 3.16. Due to the clear variability across these 3 camera views, we define 3 scenarios in this context: ”room 1 camera1”, ”room2 camera1”, and ”room2 camera2”.

All the three third-person view cameras for video recording were GoPro camera placed on top of a tripod. Each video is has 1080p resolution, 30 FPS. We collect 5 days of video data, which is 5 times of cooking class, for our ”class23” dataset[1]. In each day’s recording, ”room1” and ”room2” may have a different number of students based on the way the group was split into two rooms.

Data pre-processing The collection of videos is pre-processed to remove those time periods where students or staff unintentionally occluded the camera view for a long period. We also

[1] ↑Data collection stopped when the class shifted unexpectedly to all-remote learning in March 2020.

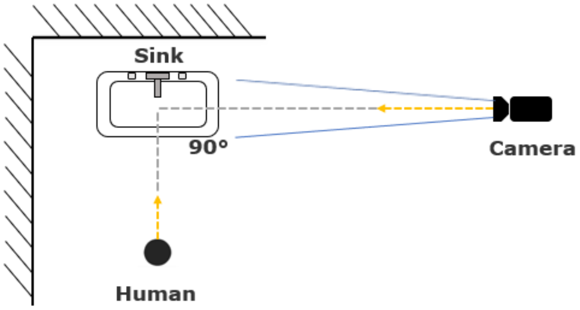


Figure 3.17. Room1 layout; Side-camera view with 90 degree angle

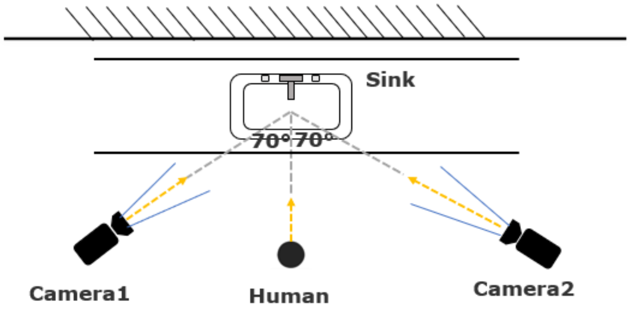


Figure 3.18. Room2 layout; Two side-camera views with approximate 70 degree angle

remove those videos for which the tripod was incorrectly positioned. All cameras were mounted and set to start recording before students arrived. After the last students finished his/her hand-hygiene, the instructor manually stop all the camera's recording. To reduce the redundant content in our data collection, we further cut each camera's video by person, where each video starts when a person begins hand-hygiene, and ends after the person finishes. After removing unsatisfied videos, a total of 63 person's hand-hygiene video remain across all three camera views.

Spatial region pre-processing The original video data collected in our dataset are under 1920 x 1080 spatial size. In each video, all the students appear in the room are being recorded. However, the hand-hygiene action recognition only applies to the student who is standing near the sink area. To construct a valid hand-hygiene recognition system, the first step is to identify the person near the sink. This task can be potentially achieved by major object localization methods, such as Faster RCNN [80], YOLO [81], and SSD [82]. Because this is the first time we collect hand-hygiene data under different rooms with different layout, illumination, and object configuration, instead of focusing on designing good performance object detector for a sub-task, we would like to first explore the feasibility of hand-hygiene recognition on these data. Therefore, we manually labeled the region of interest (ROI) of sink area for each day and each ”scenario” video. And each video’s region of interest (ROI) area is cropped and resized into 224 x 224 size. For the rest of this paper, unless specifically explained, these cropped and resized ROI images or videos are the default input to all experiments. The visualization of these cropped ROI images are shown in Figure 3.19.

3.6.2 Hand-hygiene action definition

After data collection, we want to define hand-hygiene actions types in our dataset.

Hand-hygiene task selection Reference the previous Chapter 3.3, there are three types of video analytic tasks for hand-hygiene. Based on their difficult level, we name them as ”detail level”, ”standard level”, and ”detection level” tasks. ”Detail level” hand-hygiene recognition is a task to recognize 12 different hand-hygiene actions. Some of these actions involve subtle motions of hands and fingers, which are hard to record even under egocentric camera. The ”detail level” hand-hygiene recognition is not an appropriate task for our side-view third person camera dataset. Therefore, we focus on the ”standard level” hand-hygiene task, which includes 6 different hand-hygiene actions, which are ”touch faucet with elbow”, ”touch faucet with hand”, ”rub hands with water”, ”rub hands without water”, ”apply soap”, and ”dry hands with paper towel”.

Action type adjust Moreover, we adjust the action types in ”standard level” hand-hygiene task to adapt to the class23 dataset. Due to the behavior of the students, the action set in different ”scenario” are not the same. In ”room2 camera1” and ”room2 camera2”, the location of paper towel is out of the view of both camera. Therefore, none student performs ”dry hands with paper towel” under those two cameras. Also, there exist only few examples of ”dry hands with paper towel” actions in ”room1 camera1”. Based on our previous definition in Chapter 3.3, the action of ”dry hands with paper towel” is best described under egocentric camera recording. Because the location of paper towel is uncertain and the participant could walking as well as wiping their hands. For the reasons above, we will not explore ”dry hands with paper towel” as hand-hygiene action in this paper and it will be considered as a ”non-hygiene” action. Moreover, it is quite surprising that none of the students in 5 days recording performed the action of ”touching faucet with elbow”. Therefore, we will only focus on 4 types of hand-hygiene actions in this work, which are ”touch faucet with hand”, ”rub hands with water”, ”rub hands without water”, and ”apply soap”.

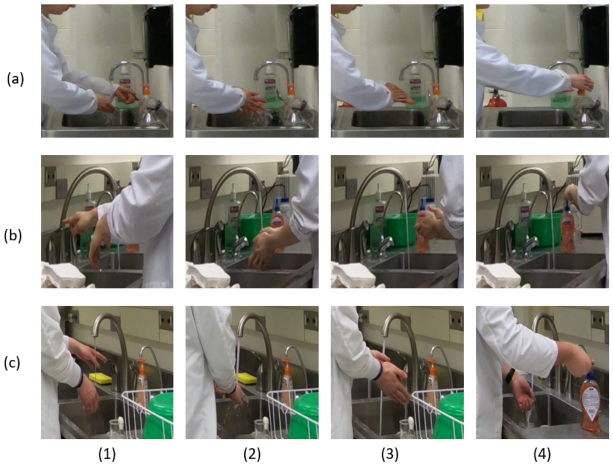


Figure 3.19. Cropped Region of interest on sink with 4 hand-hygiene actions: (1) ”touch faucet with hand” (2) ”rub hands with water” (3) ”rub hands without water” (4) ”apply soap”; Each row (a) Room1 camera1, (b) Room2 camera1, (c) Room2 camera2

3.6.3 Data labeling and availability

Frame-level labeling In our ”class23” dataset, all the hand-hygiene actions are labeled by human expert at a frame level. Beside hand-hygiene actions, ”non-hygiene” actions such as ”swing hands”, ”grab paper towel”, ”dry hands with paper towel”, and ”camera occlusion” are also labeled at the frame-level.

Data availability Our dataset consists video data to support both action recognition and detection tasks. For action recognition task, each of these hand-hygiene or non-hygiene actions are considered as ”trimmed video clips”, which will be applied to the task of action recognition for the remaining of this paper. For action detection task, the videos which includes all actions of a person's start to end hand-hygiene steps as described in 3.6.1, we refer these videos as ”untrimmed video” which will be applied to the task of action detection for the remaining of this paper. Table 3.3 lists the number of trimmed video clips for each scenario. The train, validation, and test action sets are split with ratio 0.66 : 0.12 : 0.22. For the number of untrimmed video, There are 9, 10, 15 of untrimmed videos for testing purposes in room1 camera1, room2 camera1, and room2 camera2, respectively.

Table 3.3. Number of trimmed video clips for each scenario: out of parentheses hand-hygiene actions only; in parentheses: hand-hygiene and non-hygiene actions

Scene	Train	Validation	Test
Room1 cam1	127 (177)	16 (26)	60 (80)
Room2 cam1	55 (78)	9 (12)	34 (48)
Room2 cam2	96 (114)	14 (17)	53 (65)

3.7 Produce washing in student farm

In previous sections, we introduced our data collections for the topic of hand-hygiene action recognition in different scenarios. However, if we proceed from the whole picture of food handling, hand-hygiene is only one of the steps which often happens during food handling. To explore more complicate food handling situation, we also collected video data from a student farm for produce washing.

3.7.1 Data collection

In this data collection, we invite the staff in a student farm of Purdue University to collaborate with us to record their daily produce washing steps. In the student farm, the produce washing is a daily based activity. In the morning, all the produce in the farm are harvested and collected in different baskets. The staff is responsible to wash all the produce and store them into the storage. After washing the produce, the containers and work stations should also be cleaned follow standard procedures.

To collect the realistic produce washing video data, we applied 4 different cameras to record the produce washing steps. Among all the 4 cameras, two of them are egocentric view and the rest are static third person view. The two egocentric cameras applied the same chest and nose camera setting as discussed in Section 3.2. The two third person view cameras also follow the same setting as the wall camera introduced in Section 3.2. To explore the appropriate camera placement in the produce washing room, the two third person view cameras are localized in two separate positions close to the window and corner of the room.

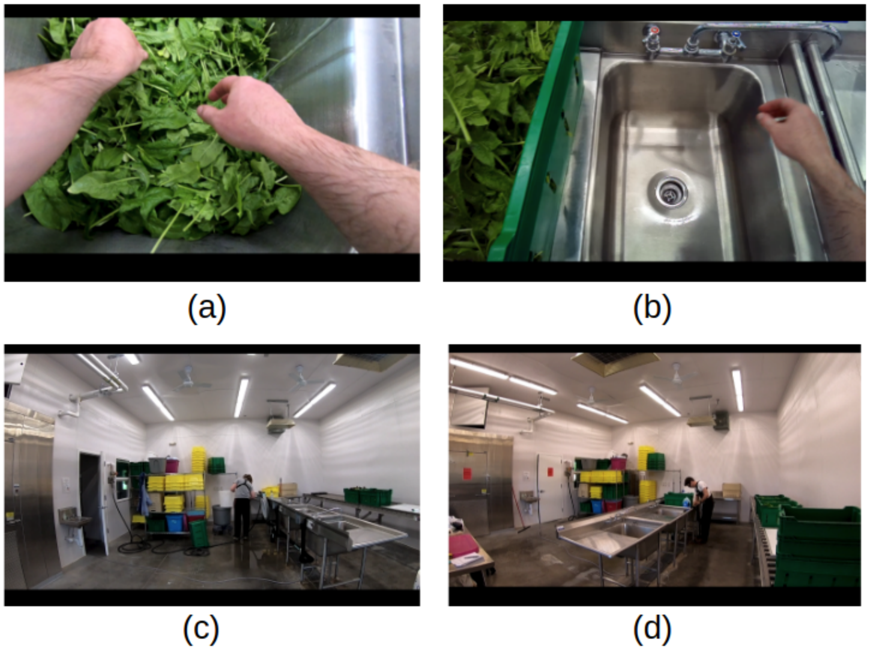

(a)
(b)
(c)
(d)

Figure 3.20. 4 cameras video frame display: (a) chest camera (b) nose camera (c) third person camera at window (d) third person camera at corner

Before the produce washing begins, the food staff is asked to wear a chest and a nose camera on him/her. All 4 cameras are recording continuously until the staff claims the produce washing is finished. The video frames from all 4 cameras are shown in Figure 3.20.

We collected the total video data more than 10 hours for each camera view of continuous produce washing. The general video duration for each single produce washing is approximate 30 minutes, which is relatively long compare to hand-hygiene actions.

3.7.2 Potential task

Due to the constraint on time schedule, the data we collected in produce washing are not being explored with any experiment. Therefore, in this section, we will analyze the potential topic to explore with this data collection.

The produce washing is required to follow a long sequence of standard produces, which could last more than half an hour to finish are required steps. The actions happen during produce washing could include but not limited to: wash produce, wash hands, clean containers, put produce into storage room, and staff takes a break. Part of these actions is shown in Figure 3.21.

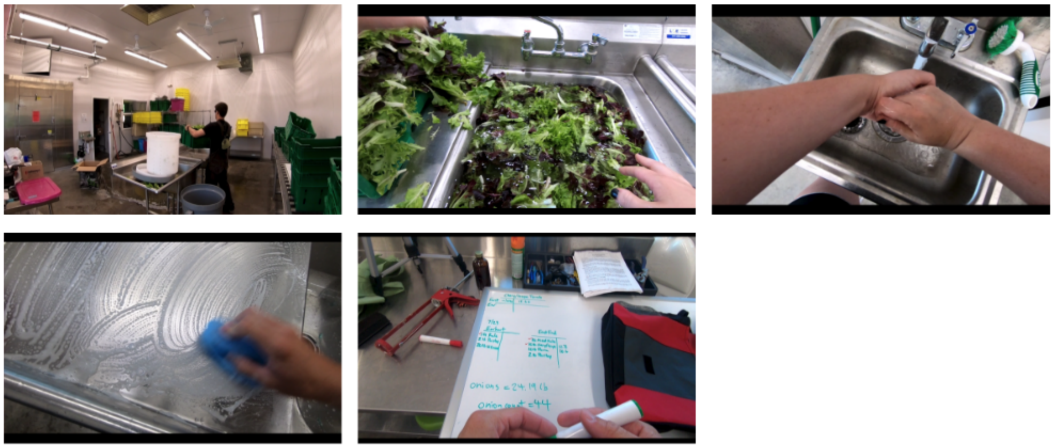


Figure 3.21. Action video frames in produce washing.

Besides these meaning actions which contributes to produce washing quality, there could also exist potential distractions during produces washing. Behaviors such as staff chatting, arguing as Figure 3.22, or visitors break into the room could affect the efficiency of produce washing.

From food handling perspective, the distractions happen during produce washing are inappropriate and should be reported to the supervisor in the time. However, it is inefficient for human labors to watch such a long duration video and summarize all the accidents within it. Therefore, with the support of video monitoring, we can apply computer vision algorithm on these videos to automatically summarize all the events occur in the video.

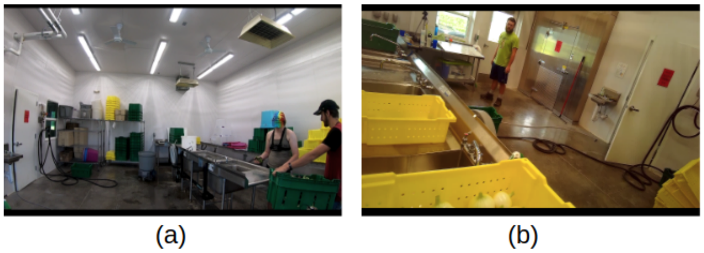


Figure 3.22. Distraction video frames in produce washing: (a) chatting. (b). arguing.

The most relevant topic in computer vision which matches our desire is video summarization [83][84][85] which automatically generates brief summarization from long video. The detailed application of using video summarization for produce washing videos is expected to be explored in the future.

4. HAND-HYGIENE IN EGOCENTRIC VIDEO

This chapter introduces hand-hygiene recognition method by using only egocentric video data. The majority content in this chapter is also covered in our previous work [86]. Egocentric video is recorded by mounting wearable cameras on human body. This video type contains rich body and camera motion, which matches the characteristic of hand-hygiene actions. In this chapter, we apply the chest camera view video data from our ”nelson100” dataset as mentioned in Chapter 3.5 to explore hand-hygiene action recognition. The hand-hygiene action recognition task is at the difficulty level of ”standard level” as discussed in Chapter 3.3. In Chapter 4.1, we discuss the adjustment of ”nelson100” data and ”standard level” hand-hygiene task in this work. In Chapter 4.2, we introduce the overall system design for hand-hygiene recognition in egocentric video. In Chapter 4.3, we introduce and compare the methods of two-stream network and LSTM. In Chapter 4.4, we discuss the processing details on untrimmed hand-hygiene videos.

4.1 Action set adjustment

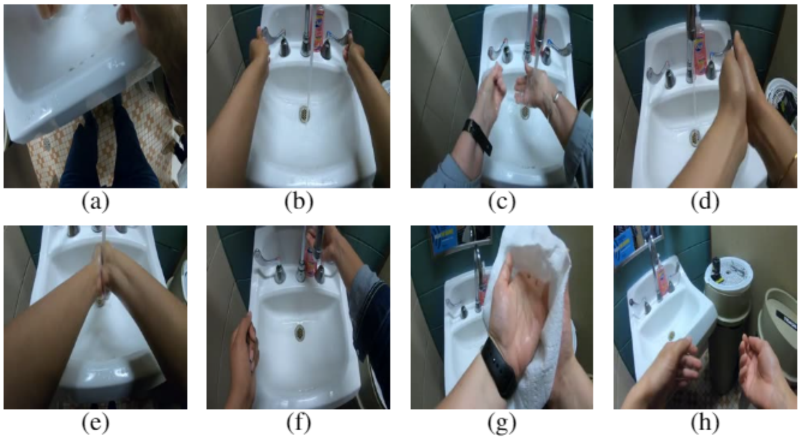


Figure 4.1. Action set: (a) touch faucet with elbow (b) touch faucet with hand (c) rinse hands (d) rub hands without water (e) rub hands with water (f) apply soap (g) dry hands with paper towel (h) non-hand hygiene action

Because this work was done on the early stage of our entire hand-hygiene exploration framework, the action set we explored is not exactly the same but close to the ”standard level” hand-hygiene task. To explore the strength of hand rubbing during the hygiene steps, we further divided the action ”rub hands with water” in our ”standard level” task into ”rub hands with water” and ”rinse hands”.

The detailed definition of all hand-hygiene actions in this chapter is the following: A subject should not touch the faucet with their hands, to avoid re-contamination [3]. Therefore, we need to distinguish whether the subject touched the faucet with hand or with an elbow. Moreover, it is also important to detect the strength used to rub hands. We enable this by labelling an action of rinse hand, where the subject rubs hands with little strength. Furthermore, the subject should apply soap before hand washing and dry their hands after hand washing. When soap is applied, the subject needs to rub hands without keeping their hands in water. Based on these principles, we define and label 8 actions as indicated in Figure 4.1, which are: touch faucet with elbow, touch faucet with hand, rinse hands, rub hands without water, rub hands with water, apply soap, dry hands, and a background non-hygiene action. All 8 actions are manually labelled at the frame-level.

Moreover,the ”nelson100” videos are recorded under 1080p resolution with 30 FPS and wide viewing angle. To increase processing speed, we further down-sampled these videos to 480×270 resolution for all the usage in this chapter.

4.2 Two stage hand-hygiene system

4.2.1 System design background

Activity recognition for untrimmed video clips is often termed temporal action proposals or temporal action localization [87]. For hand-hygiene videos, our goal is to localize temporal regions where hand-hygiene actions happen in untrimmed videos. Then, by applying an action classifier on these targeted short segments, we will be able to identify what hand-hygiene actions have been performed by a participant.

Our hand-hygiene videos contain densely-distributed hand actions with an average of 5 different types of actions per video. Non-hygiene actions such as standing or walking around can happen anytime during the video. Thus, it is difficult for coarse-level temporal proposal methods [6][7] to localize hand-hygiene actions in our videos. Moreover, the average duration of an untrimmed hand-hygiene video is around 1 minute. Therefore, the temporal segmentation method [88] designed for long duration egocentric videos is also not applicable here.

4.2.2 System basic description

We propose a two-stage system to localize and detect hand-hygiene actions from untrimmed videos as shown in Figure 4.2.

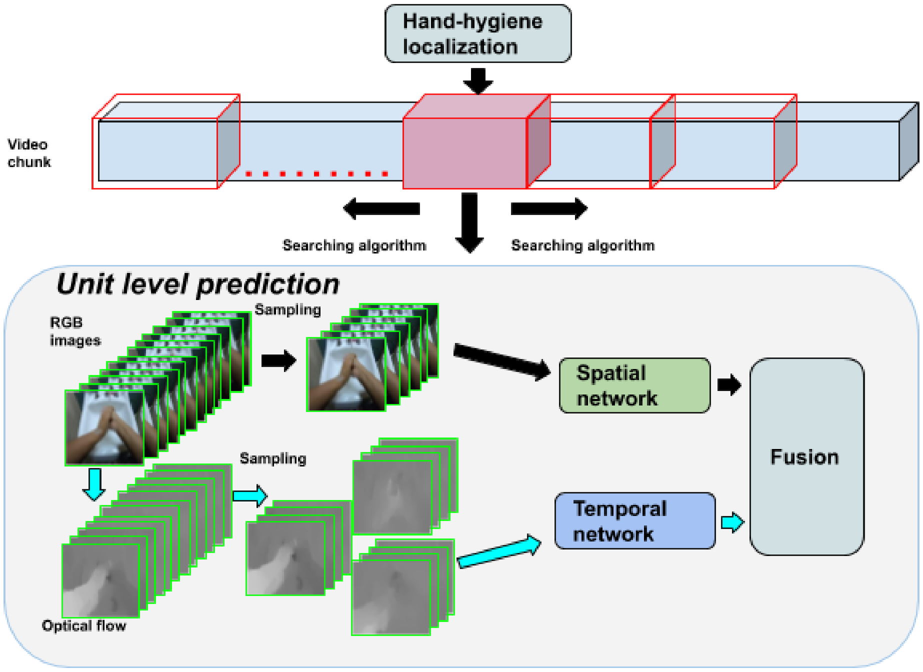


Figure 4.2. Two-stage prediction system pipeline

In the first stage of our system, we localize the temporal interval where hand-hygiene actions happen inside the untrimmed video. Hand-hygiene actions are dominated by hand and arm motion, which can be interpreted as the appearance of hands, arms and their related motion patterns. We divide our 8 types of actions into two categories. First, actions containing strong motions, including rinse hands, rub hands without water, rub hands with water and wipe hands, are considered as action class ”1”. The other four types of actions, including non-hygiene actions, are labelled as action class ”0”. We apply a low cost hand mask and motion histogram features to process the input video. And the goal of this first stage is to correctly predict these labels at a frame-level inside the whole untrimmed video. The implementation details are explained in Section 4.4.1 and 4.4.2.

In the second stage, we use a two-stream network to make a fine-level prediction on all 8 action classes in our data. Using the location information from the first stage, we apply a deep learning model to predict on unit of 30 frames. By combing this model with a certain searching algorithm, we are able to localize and identify all the hand-hygiene actions that happen inside the input video. The construction of deep learning model is introduced in Section 4.3. The implementation detail of the system's second stage and overall performance are explained in Sections 4.4.3 and 4.4.4.

4.3 Hand-hygiene action classification

Based on our system design, we need to construct a robust model which is capable to recognize all hand-hygiene actions. In this section, we explore the performance of the two-stream network on recognizing actions in trimmed hand-hygiene video clips.

4.3.1 Two-stream convolutional neural network

Hand-hygiene actions are composed of hand and arm motions, which lack meaningful objects that might reveal clues about action itself [32]. In this Section, we would like to apply a deep learning based model to learn deep feature representations to distinguish all 8 types of actions.

The two-stream network has demonstrated its effectiveness in activity recognition in third-person videos [35]. The two-stream network considers both appearance and motion information by separately constructing a spatial-stream ConvNet and a temporal-stream ConvNet. The spatial-stream ConvNet takes RGB images as inputs, which provides appearance information in the scene. On the other hand, the temporal-stream ConvNet takes chunks of optical flow images between consecutive frames as inputs. These optical flow images provide strong clues to the motion information that exists in the video.

Optical flow is a commonly used method to describe motion of objects in visual scenes. Assume the brightness condition is constant, for an image I, the change of intensity at

$I(x, y, t)$ between two consecutive images are represented as Δx, Δy and Δt. Due to the unchanged intensity of the same pixel over consecutive images, we can have the equation as $I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$. By eventually solving this equation, we get V_x, V_y as the velocity in x, y direction, which are the optical flows. However, solving this equation is not a straight forward problem and requires additional conditions.

Basically, optical methods can be divided into 2 categories, dense optical flow and sparse optical flow. For dense optical flow methods, they manage to process all the pixels in an certain image area or the whole image. In contrast, sparse optical flow methods only track small amount of pixels, which can be selected by algorithm such as Harris Corner detector. In our experiment, we compared two different dense optical flow methods: Farneback optical flow [89] and TV-L1 optical flow [90]. As it is indicated in Figure 4.3, the Farneback optical flow images can barely detected the optical flow on the arm skin region, therefore, only the flow of border arms are correctly computed. As TV-L1 optical flow, all the optical flows on object regions can be found. Therefore, we choose TV-L1 method to compute optical flow images.

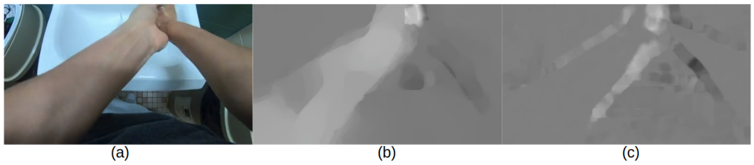


Figure 4.3. Optical flow comparison:(a) RGB image.(b) TV-LV1 optical flow.(c) Farneback optical flow.

After getting prediction score separately from spatial and temporal network, the final prediction result is generated from a score fusion of these two individual networks.

For our experiment, we use the method of Wang *et al.* [91] with implementation [92], which applies deeper network structures and takes advantage of a small learning rate and more data augmentation techniques.

4.3.2 Experiments on two-stream network

We split the 200 videos in our dataset into training and testing sets with 135 videos and 65 videos respectively. All videos are trimmed into clips where each clip includes only one action from beginning to end, which result in 1380 training video clips and 675 testing video clips.

For training, we use the pre-trained ResNet 152 [21] from ImageNet [93] for both the spatial and temporal networks with fine-tuning on the 8 action classes. Input video with 480 × 270 are down-sampled to resolution 224 × 224 to fit the ResNet.

For testing, we apply both the sparse [91] and dense sampling strategies. For the sparse sampling, only 25 frames with equal distance step are selected from each input video clip. For dense sampling, all frames are selected. The two-stream network model predicts each selected frame individually and uses the average prediction score from these frames as the prediction for the input video.

Table 4.1. Two-stream network performance

Model	Accuracy
Spatial Network sparse	85.3%
Spatial Network dense	86.4%
Temporal Network sparse	84.4%
Temporal Network dense	86.8%
Fusion sparse	87.3%
Fusion dense	87.7%

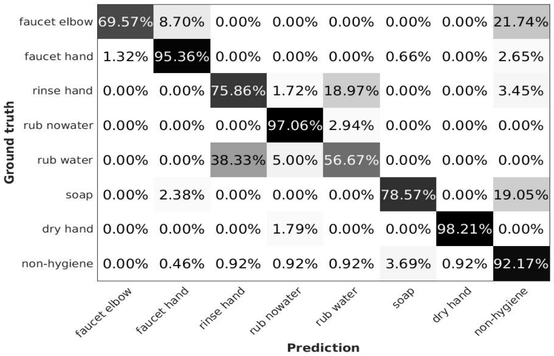


Figure 4.4. Confusion matrix for two-stream network fusion, dense sampling.

The results in Table 4.1 show the average detection accuracy among all 675 video clips. The dense sampling only outperforms the sparse sampling by 0.4 % after score fusion. Therefore, sparse sampling is a better strategy for its faster processing speed and minor sacrifice on detection accuracy.

A prediction confusion matrix for dense sampling after score fusion is shown in Figure 4.4. We observe that the trained deep model performs well on several of the actions with over 90% accuracy. However, for the action pair of rinse hands and rub hands with water, many participants switch between these two actions in a short period of time, which caused difficulty in creating ground truth labels. Therefore, the trained model makes mistakes on recognizing these two actions.

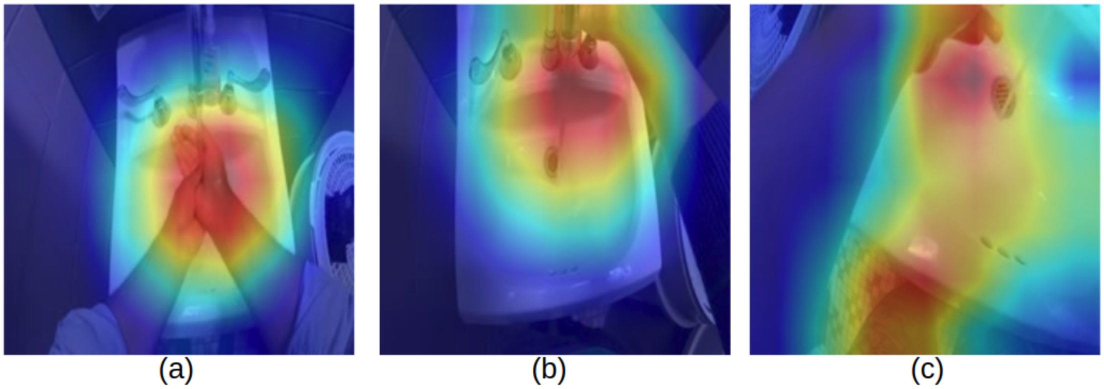


Figure 4.5. Grad cam [54] results of (a) rub hands with water (b) apply soap (c) touch faucet with elbow

To understand what the two-stream model has learned, we use Grad-cam [54]. Figure 4.5 shows these heat maps, where the highlighted region indicates saliency for a target class. In Figure 4.5 (a), the trained model successfully captures hand related regions to recognize rub hand with water. In Figure 4.5 (b)(c), however, the chest camera angle hasn't completely captured the entire action of applying soap or touching faucet with elbow. As a result, the trained model makes mistakes by recognizing these two actions as non-hygiene actions.

4.3.3 Convolutional neural network combine with LSTM

Beside the two stream network, we also explore the performance of CNN network combines with LSTM (Long-Short Term Memory). For hand-hygiene actions, they usually require the subject to repetitively performing some basic movements. For example, when rubbing hands, the subject will repetitively do the left-to-right or right-to-left hands rubbing. Therefore, we think it is worth to study this latent variable in time sequence to classify hand-hygiene actions.

One of the most efficient methods to the gradient issue is called Long-Short Term Memory (LSTM). As it is indicated in 4.1, the LSTM is constructed in a combination of input gate i_t, forget gate f_t, output gate o_t and input modulation gate g_t. All the i_t, f_t, o_t, g_t, c_t and h_t have the same dimensional d. The hidden units in LSTM structure is computed through memory cells c_t. The gradient flows allowed to pass through the cell unit c_t in controlled through forget gate f_t and input gate i_t. Each of these gates generate a value between 0 to 1 as rate to selectively forget its previous memory and study from its current input.

$$\begin{aligned}
\mathrm{i}_t &= \sigma(W_{x\mathrm{i}}x_t + W_{h\mathrm{i}}h_{t-1} + b_{\mathrm{i}}),\\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f),\\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o),\\
g_t &= tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c),\\
c_t &= f_t \odot c_{t-1} + \mathrm{i}_t \odot g_t,\\
h_t &= o_t \odot tanh(c_t),\\
y_t &= tanh(W_{hy}h_t + b_y)
\end{aligned} \tag{4.1}$$

As it is defined in [94], there are 3 types of LSTM structures. These structures are different by the format of input and output, which can be described as multiple inputs to single output, single inputs to multiple outputs and multiple inputs to multiple outputs.

Each of these LSTM structure in Figure 4.6 fits for one type of computer vision problems. For the problem of activity recognition, the general idea is to describe the particular action

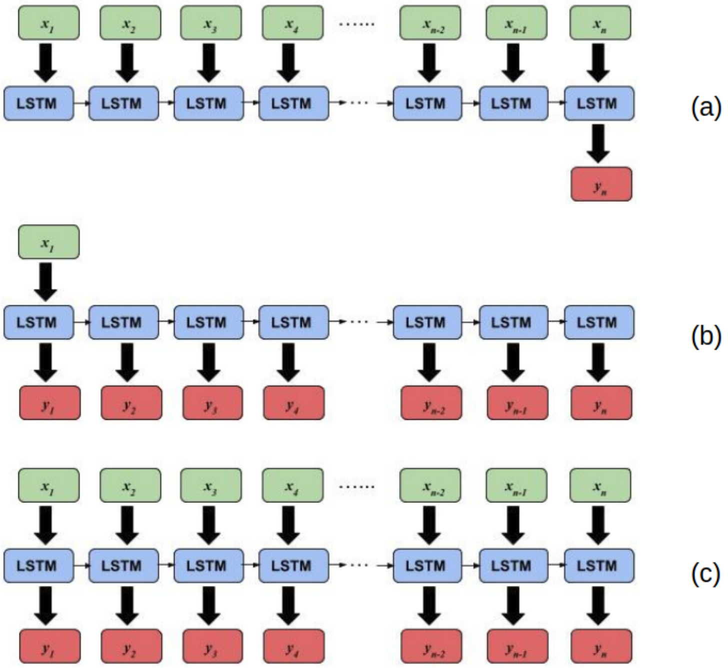

x_1 x_2 x_3 x_4 …… x_{n-2} x_{n-1} x_n
LSTM LSTM LSTM LSTM … LSTM LSTM LSTM
y_n
(a)
x_1
LSTM LSTM LSTM LSTM … LSTM LSTM LSTM
y_1 y_2 y_3 y_4 y_{n-2} y_{n-1} y_n
(b)
x_1 x_2 x_3 x_4 …… x_{n-2} x_{n-1} x_n
LSTM LSTM LSTM LSTM … LSTM LSTM LSTM
y_1 y_2 y_3 y_4 y_{n-2} y_{n-1} y_n
(c)

Figure 4.6. Three LSTM structures

type from an input video. As it shown in Figure 4.6 (a), the first LSTM structure can take multiple input video frames and generates an action label as output.

For our hand-hygiene videos, we would like to apply the sequential input to sequential output format, as it is indicated in Figure 4.6 (c), which allows us to track on the prediction result in each time step.

4.3.4 Implementation details of Long-Short Term Memory model

In this section, we explains the extraction of input features to LSTM structure. An LSTM based model usually combined with deep CNN features to recognize temporal latent information among actions. For our data, we would like to extract deep CNN feature for every video frame to be used for LSTM training and testing.

Deep CNN features has been proved its efficiency in recognizing third person view actions [94][95]. Inspired from their works, we would like to test its performance when applies to our hand-hygiene actions.

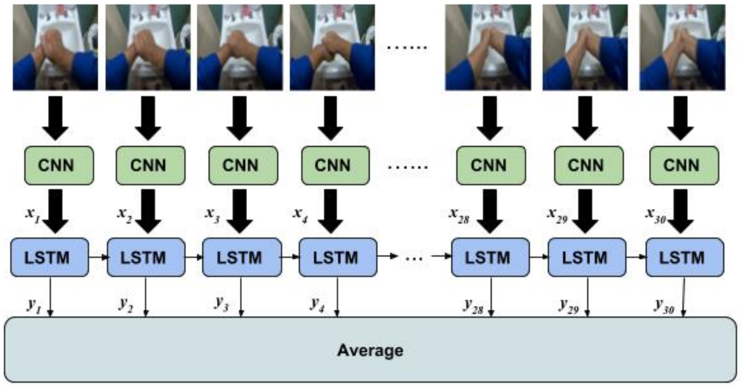


Figure 4.7. CNN feature + LSTM structure

We use ResNet 152 [21] pre-trained on ImageNet as the feature extractor to LSTM due to its effectiveness on the in ILSVRC 2015 classification task. For a video frame at time t, we re-size its spatial dimension into 224x224 as an input image to ResNet 152 and collect output features with dimension 1x2048 from it. The feature vector extracted at time step t is used as the input x_t to the LSTM. For the selection of hidden unit h_t dimension d, we select it to be $\frac{1}{4}$ of the input vector as 516 to force the model to select and summarize information from input feature.

As in figure 4.7, the LSTM model takes both x_t and the previous hidden states h_{t-1} into consideration to cre and outputs a prediction label vector y_t. The length of y_t is the number of action classes N, and the predicted value at index i of y_t indicates the confidence score for class i. For our dataset, we have N=8 number of actions.

When the procession at time step t finished, we continually input the feature vector from next time step x_{t+1} into the LSTM, where x_t and x_{t+1} are features extracted from consecutive frames in a video at time t and $t + 1$. Through continually input feature vectors from time t to $t + L$, we receive a sequence of output predictions, where L is the fixed number of time steps included in an LSTM.

After processing all the time steps in LSTM, we collected totally L prediction vectors $[y_1, y2, ..., yL], y_{\mathrm{i}} \in R^8$. Corresponding to our 30 FPS video, we set the length of L equals to

30 to match the length of 1 second time period. Because of the activation function *tanh*, the values in y_{i} are match into the range $[-1,1]$. Through averaging 30 y_{i} vectors and applies softmax, which maps the confidence score into prob-abilities, we get the final classification result for this chunk of 30 frames video clips.

4.3.5 Training and testing of LSTM

In this stage, we only use the chest camera view to test the efficiency of LSTM. Because of the camera location, some of our videos contain significant occlusion and cannot be used. After removing these occluded videos, 191 videos remain. For the experiments in this paper, we split the dataset into training and testing parts with a 2:1 ratio. The training part contains 134 videos and the testing part has 57 videos. For the convenience of using existing CNN for feature extraction, we downsample all videos to a resolution of 224x224.

Each of our videos in training set has been cut into video clips according to our ground truth annotation. Each video clip only contains one action from begin to end. Inside a video clip, we extract overlapped video frames with length 30 frames with stride 1 as training samples as indicated in Figure 4.8. This strategy guarantees all the variations of a 30 frames duration video chunk in an action type can be learned by the LSTM model.

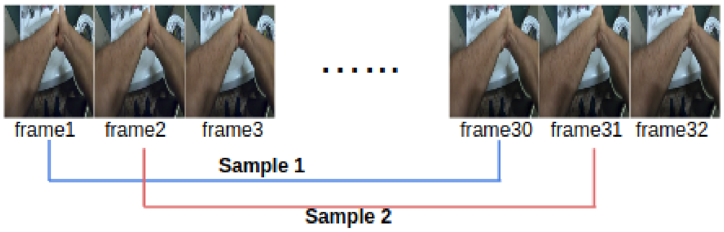


Figure 4.8. Training sample creation

For the loss function, we use cross-entropy loss (4.2), where y_{ti} represent the prediction label vector at time step t for class i. The loss equals the sum of all the time steps L=30 among all classes N=8.

$$L = \sum_{t=0}^{t=29} \sum_{\mathrm{i}=0}^{\mathrm{i}=7} y_{\mathrm{ti}} log(softmax(y_{\mathrm{ti}})) \tag{4.2}$$

Table 4.2. LSTM comparison in number of layers

Model	Accuracy
1 layer LSTM	72.9%
3 layers LSTM	74.3%
5 layers LSTM	73.6%

Moreover, we also test the variations on LSTM structure which includes number of layers. It has been explored that deep in LSTM layers can efficiently use parameters by distributing them in several layers [96]. Therefore, we varied the number of layers in LSTM with 1, 3 and 5. When LSTM has multiple layers, we add a drop rate of 0.5 in training to prevent over-fitting.

In testing stage, each video clip is trimmed into 30 frames video overlapped video chunk with stride 1. The averaging prediciton result among all these video chunks is the prediction result of the test video.

The results in 4.2 show that having multiple layers increases detection accuracy. However, detection accuracy no longer improves when increasing the number of layers from 3 to 5.

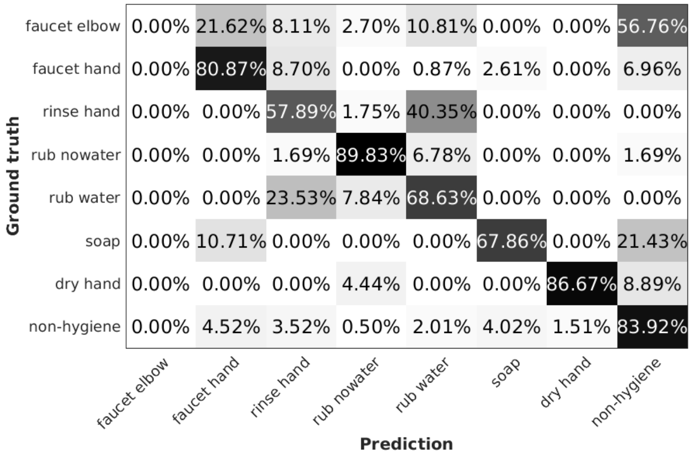


Figure 4.9. 3 layers LSTM result confusion matrix

If we take a look at the confusion matrix result in Figure 4.9, the worst performance of classification comes from action class of faucet elbow. More than half of this action has been mis-classified as non-hygiene actions. One reason is because of the camera view on chest can not capture the participant's arm or hands during this action. On the other hand, the lack of training samples is also an important factor. Among 134 training samples, there exists only 20 video clips of faucet elbow actions.

4.3.6 Comparison of LSTM and two stream network

In this section, we compare the LSTM model with two-stream network. To make this comparison valid, we trained the two-stream networks by fine-tuning the pre-trained ResNet 152 with our hand-hygiene dataset and using the fine-tuned networks as feature extractors for the LSTM. We also constructed another LSTM model which takes a stack of optical flow images with the same size as the two-stream temporal network

Table 4.3. Two stream networks and LSTM comparison

Model	Accuracy
3 layers LSTM RGB	80.2%
3 layers LSTM Optical Flow	80.5%
3 layers LSTM fusion	81.6%
Two-stream network spatial	86.4%
Two-stream network temporal	86.8%
Two-stream network fusion dense	87.7%

The results in Table 4.3 show that the two-stream network achieves higher detection accuracy than LSTM model. Therefore, we prefer to use two-stream network as our hand-hygiene recognition model for the rest of this work.

4.4 System implementation details

4.4.1 Hand-hygiene localization

Hand mask Hand poses are good indicators of hand-hygiene actions. Especially when rinsing or rubbing hands, the two hands overlapping each other create distinguishable patterns. In our work, we applied the pixel-level hand detection method [97] to generate hand masks. We train the model with a set of 134 images of manually labelled segmented hand regions under different illumination conditions. The resulting hand masks are gray-scale images with size $S_m \times S_n$.

Inspired by the work of Singh *et al.* [33], we create a network structure to predict frame-level ”0” or ”1” action using hand mask as features. The network is composed of 2 Conv layers followed by RELU, max pooling and LRN (local response normalization) and 2 fully connected (fc) layers. The network takes L hand masks as input. In our training stage, a cross entropy loss is applied as well as a dropout 0.5 to avoid over-fitting. In testing, the softmax score from the last fc layer indicates the prediction result.

Motion histogram Motion is also a good indicator of hand-hygiene actions. We create a optical flow histogram feature within the hand mask region to represent motion patterns. Applying the hand masks generated on dense optical flow images, we create two optical flow histograms with bin size B for both region inside and outside hand mask. Within each region, we count the magnitude and angle of optical flow for each pixel i.

$$M_{\mathrm{i}} = \sqrt{gx_{\mathrm{i}}^2 + gy_{\mathrm{i}}^2}, \theta_{\mathrm{i}} = tan^{-1}(\frac{gy_{\mathrm{i}}}{gx_{\mathrm{i}}}) \tag{4.3}$$

The pixel with θ_{i} angle that falls into the range of $[\frac{b-1}{B}\boldsymbol{\pi}, \frac{b}{B}\boldsymbol{\pi})$ contributes to the bin b with magnitude M_{i}, where $1 \leq b \leq B$. To overcome the problem of hand mask size variation, the final sum value for each bin b is normalized by dividing the total number of pixels in its corresponding region. The result histograms for hand masked region and outside hand mask region at frame t are represented as $H_{ht} = [h_{ht,1}, h_{ht,2}, ..., h_{ht,B}]$ and $H_{bt} = [h_{bt,1}, h_{bt,2}, ..., h_{bt,B}]$. The concatenation of these two histograms creates a motion representation at frame t. We also compute the ratio $R_t = \frac{\sum_{\mathrm{i}=1}^{B} h_{ht,\mathrm{i}}}{\sum_{\mathrm{j}=1}^{B} h_{bt,\mathrm{i}}}$ and hand motion sum $S_t = \sum_{\mathrm{i}=1}^{B} h_{ht,\mathrm{i}}$ as two additional features. The final representation of motion histogram at frame t is $H_t = [H_{ht}, H_{bt}, R_t, S_t]$ with size $1 \times 2B + 2$.

For classification, we apply a Random Forest classifier with 30 estimators and max depth 40 to learn the motion histogram patterns.

4.4.2 Hand-hygiene localization testing

In this section, we test the performance of the hand mask and motion histogram feature on localizing hand-hygiene actions from untrimmed videos.

Training For the efficiency of system design, we split the untrimmed 65 videos, with resolution 480×270, from 100 people's testing dataset into 43 and 22 videos for training and testing the hand localization system. To increase the processing speed of the hand-hygiene localization, the hand masks are generated with size of 32×18 and 64×36 in this experiment. Motion histogram features are generated on 480×270 dense optical flow images and applied previous generated hand masks,which resized to 480×270, on it. The hand mask network is trained under batch size 128 and learning rate $1\mathrm{e}^{-5}$ with a stack of $L = 5$ hand masks. The Random Forest classifier is trained with three bin size options: 9,12 and 16.

Testing The testing experiment is done on 22 untrimmed videos with the label "0","1" as positive and negative labels on every frame. The hand mask network slides through the whole video and predicts using an overlapped stack of hand masks. The Random Forest classifier predicts on every frame of each video. For each testing video, we count the TP (true positive), TN (true negative), FP (false positive), and FN (false negative) at the frame-level. The performance of each classifier is measured by the accuracy $= \frac{tp+tn}{tp+tn+fp+fn}$ and true negative ratio $= \frac{tn}{tn+fp}$.

Table 4.4. Classifier comparison

Model	Accuracy	True negative ratio
9 bins motion hist	73.7%	74.0%
12 bins motion hist	74.7%	71.6%
16 bins motion hist	75.0%	70.4%
32x18x5 hand mask network	78.9%	74.1%
64x36x5 hand mask network	80.7%	76.8%

Table 4.4 indicates the average accuracy and true negative ratio among 22 testing videos. We notice that hand mask network outperforms the combination of motion histogram with Random Forest, and the input stack with hand-mask size $64 \times 36 \times 5$ is the best option. However, since the hand mask feature only reveals appearance information, mistakes can be made when a participant holds his/her hands in a overlapped manner without motion.

In the final design of hand-hygiene localization, we first apply the hand mask network to predict frame-level negative label "0" and positive label "1". Then we re-check the positive predicted frames with motion histogram and Random Forest classifier. A frame is predicted as positive label "1" only when it is confirmed by both classifiers. Otherwise, a frame is marked as negative label ”0”. The detailed performance of this structure will be explained in the next section.

4.4.3 Hand-hygiene search and detection

In this Section, we describe the second stage of our two stage hand-hygiene system. As it has been shown that two-stream network has a reasonable performance on recognizing trimmed hygiene videos, we would like to use this model as a unit level detector to further process untrimmed hand-hygiene videos.

Location unitization We consider an untrimmed video composed by non-overlapped units. Each unit has 30 frames, which is 1 second in under 30 FPS. We start by assigning each unit with a unified label of ”0” or ”1”. Based the frame-level prediction from the first system stage, if a unit contains more than 15 frames of positive label ”1”, the unit is marked with ”1”. Otherwise, it will be marked as ”0”.

Unit level prediction The unit with positive label ”1” indicates those actions with strong hand motion. We start to check these locations first. To recognize all 8 action classes, we employ the pre-trained two stream network in Section 4.3.1 with a sparse sampling strategy. We sample 10 RGB images and 3 non-overlapped 10 pairs of optical flow images for each frame unit. The spatial network and temporal network individually predict using their sampled inputs and fuse the results with equal weights for the final prediction.

Searching algorithm There exist 7 types of hand-hygiene actions to recognize. However, due to short duration and indistinctive motion patterns, actions of applying soap, touching faucet with hands and touching faucet with elbow are categorized into class ”0” in the localization step. These actions normally happen before or after the actions labeled in ”1”. Therefore, we designed a searching algorithm to find all 7 types of hand-hygiene actions. The algorithm iteratively searches the surrounding unit of each label ”1” unit and makes predictions using the two-stream network model. The algorithm stops when it reaches non-hygiene actions on both left and right-side unit. After finishing the searching algorithm, each unit visited has been predicted with a result label and the unvisited units are automatically considered as non-hygiene actions.

4.4.4 System testing

Testing of the overall two-stage system is applied on the same 22 untrimmed videos in Section 4.4.2. To evaluate a video's prediction accuracy, we compare the prediction result with our frame-level ground truth labels. We map unit-level prediction result into a frame-level result by replicating each unit's result by 30 times.

The system performance is evaluated by frame-level accuracy $= \frac{tp+tn}{tp+tn+fp+fn}$ and the percentage of units visited (PV). We introduce the PV to measure the system efficiency. A high frame-level accuracy with a low PV value indicates the system was effective at localizing hand-hygiene actions and avoiding non-hygiene regions. For comparison, we create a baseline by applying two-stream network model to densely predict all non-overlapping units in each untrimmed video.

Table 4.5. Average performance on 22 untrimmed videos. H: **H**and mask network localization, M: **M**otion histogram localization, S: **S**earching algorithm with two-stream network recognition

Methods	Accuracy	PV
Baseline	79.3%	100.0%
H+S	79.3%	81.5%
H+M+S	78.6%	76.4%

As indicated in Table 4.5, the baseline system that checked every unit in each video obtains an average accuracy of 79.3%, which is lower than the performance on Section **??** due to the strict frame-level comparison. When applying the hand mask network only on the first system stage of localization, the PV drops from 100% to 81.5% while maintaining the same accuracy as the baseline system. This proves that the hand-hygiene localization stage helps to avoid processing the non-hygiene action. By applying the hand mask network with motion histogram, the PV further drops to 76.4% while sacrificing 0.7% detection accuracy. It is worth to note that the average percentage hand-hygiene actions occupied in the 22 untrimmed videos is 71.3%, which is the upper bound for the PV value.

4.5 Conclusion

In this chapter, we introduced our new hand hygiene-egocentric two stage system to localize and recognize hand-hygiene actions in untrimmed hand-hygiene video. The system consists of two stages. In the first stage, our system takes of the hand mask and motion histogram feature to localize hand-hygiene actions temporally. In the second stage, we expanded the two-stream network model to combine with a searching algorithm to recognize all the hand-hygiene actions in the video. The system has achieved an acceptable performance.

5. HAND-HYGIENE IN MULTIPLE CAMERA VIEWS

In this chapter, we compare different camera views in hand-hygiene actions and propose a two-stage system, which involves multiple camera views, to detect hand-hygiene actions in untrimmed video. The majority content in this chapter is also covered in our previous work [98]. As we discussed in Chapter 4, we are capable at recognizing hand-hygiene actions under a egocentric camera view with chest mount camera. However, the efficiency of other camera views is not explored. Stationary cameras placed in a so-called third-person perspective have been used for surveillance, person and vehicle detection and re-identification, activity recognition, and anomaly detection. When recognizing activities of a person, third-person cameras have the advantage of viewing actions from the side view. First-person, or egocentric cameras are mounted on the person performing the activity, often on their head or chest [4]. These cameras have the advantage of viewing the person's hands and any objects being manipulated, and are particularly useful to observe subtle hand motions and small objects. However, because they are mounted on a person, these cameras often move chaotically as the person moves. As a result, they may not capture the desired activities, and video processing methods like background subtraction and camera calibration become more difficult [4]. In this chapter, we extend hand-hygiene action recognition from solely using egocentric video into a combination of using both egocentric and third-person video. In Chapter 5.1, we introduce the overview of our system design. In Chapter 5.3, we describe our experiment on evaluating deep learning model performance on trimmed action clips. In Chapter 5.4, we evaluate the performance of the entire system. In Chapter 5.5, we conclude our work with a brief discussion and summary.

5.1 System overview

In this section, we describe a basic two-stage system to detect standard-level hand-hygiene actions, namely, the 7 actions described in Chapter 3.3, that occur in real-life scenarios, i.e., in untrimmed videos. The data involved in this work comes from the chest, nose, and wall camera view videos in our ”nelson100” dataset as mentioned in Chapter 3.5. Our two-stage design is motivated by the desire to apply low-complexity processing during a first pass,

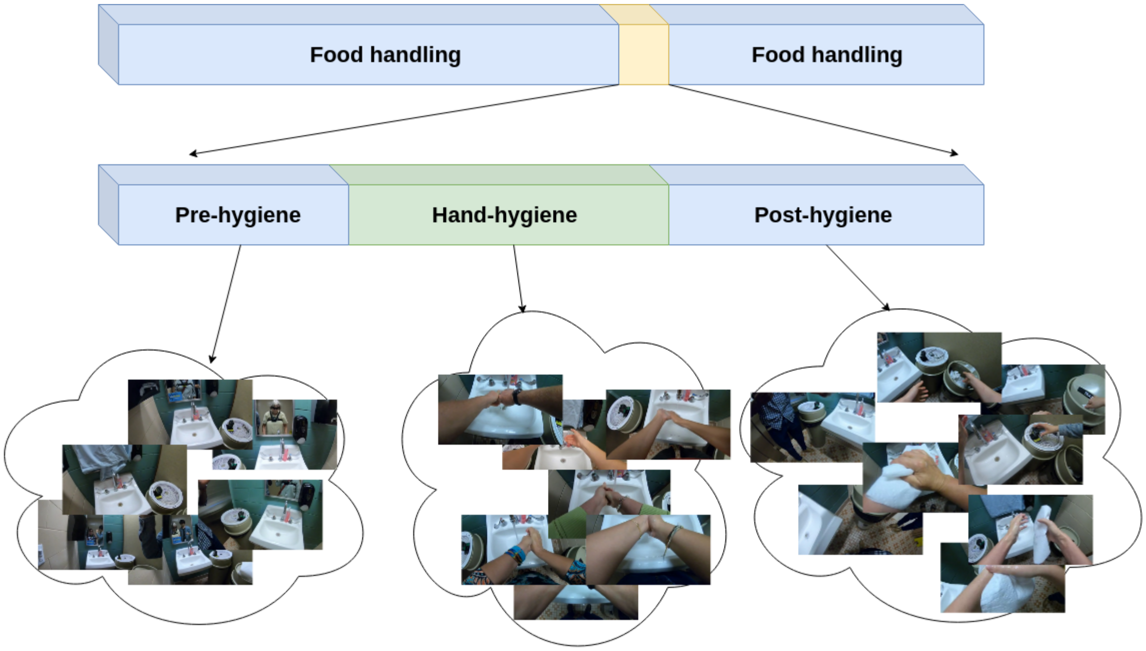


Figure 5.1. Hand-hygiene periods in untrimmed video.

with the goal to reduce the amount of video that later must be processed with more complex methods in the second stage. Both stages process each of the wall and chest camera outputs, and the specific tasks of each stage are motivated from our experiments described below in Section 5.3. In particular, the two stages and selection of which camera should be applied to a certain hand-hygiene stage are motivated by our experiment (described below in Section 5.3.1) which explores which camera is most effective for each action, and by our observations from Figure 5.1 above.

Recall that most hand-based actions are densely located in the hand-hygiene period and partially distributed within the post-hygiene period. All the remaining video content consist of non-hygiene actions with unknown and variable duration. Thus, it would be inefficient to densely process the entire untrimmed video with a computationally-complex CNN model. Recall that an untrimmed video clip contains more than one action type, while a trimmed video clip contains only one action.

The first stage of the system consists of two so-called coarse classifiers; one processes the wall video and one processes the chest video. They each densely process the entire untrimmed video and localize potential candidates for the temporal regions that might contain standard-level hand-hygiene actions. In the second system stage, we apply two so-called refinement classifiers that only process the candidate locations identified in the first stage.

Specifically, in the first system stage, as shown in Figure 5.2, we apply the wall coarse classifier to densely process the entire untrimmed video in non-overlapping 30 frame units. Even if a non-overlapping split might cut a small portion of consecutive hand-hygiene action into two different units, this will have little affect on our goal here, since when our system is applied in practice, whether the hands are detected as rubbing for 10 or 10.1 s will not influence on the final hand-hygiene quality. The entire untrimmed video is then divided into pre-hygiene, hand-hygiene, post-hygiene regions, and candidate regions of the "faucet elbow" action. The pre-hygiene region will not be further processed later in the system, but the other regions will be processed by subsequent specifically-targeted classifiers. Design considerations for the wall coarse classifier are described in Section 5.3.9.

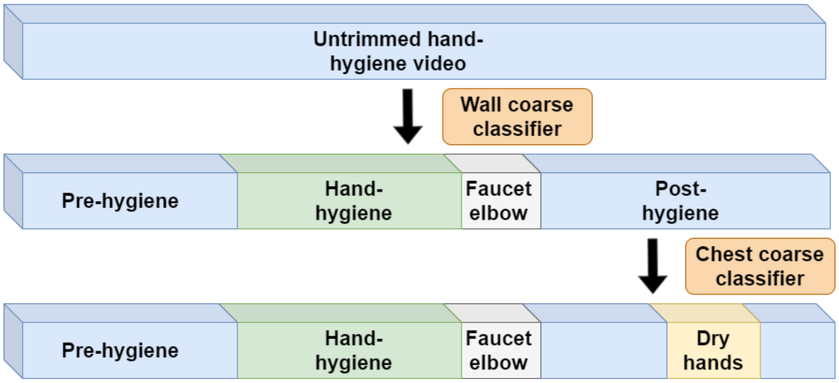


Figure 5.2. System stage 1: Untrimmed hand-hygiene video processing with coarse classifiers.

Additionally in the first system stage, the chest coarse classifier processes only the region identified as "post-hygiene". Its goal is specifically to identify whether the action of "dry hands" happened or not. Further detail on its design is provided in Section 5.3.10.

In the second system stage, shown in Figure 5.3, we apply two refinement classifiers. The first wall-refinement classifier only processes the short temporal region that was identified by the first-stage wall coarse processor as being a candidate region for the "faucet elbow" action. Its goal is simply to verify the existence of the "faucet elbow" action and further refine its temporal location.

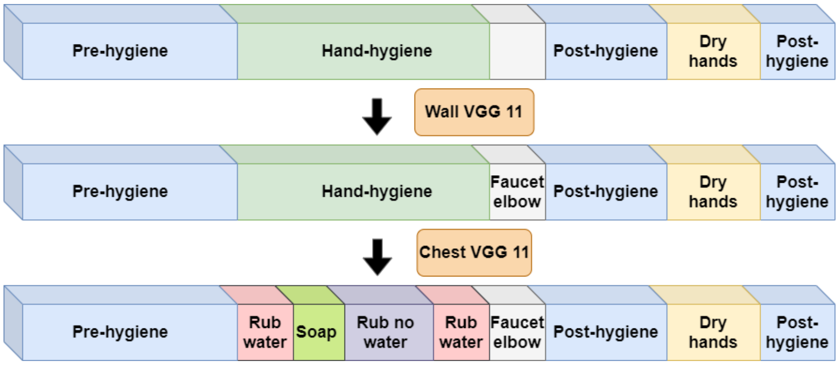


Figure 5.3. System stage 2: Trimmed hand-hygiene video processing with refinement classifiers.

The final classifier uses the chest view to refine the actions that take place in the hand-hygiene region. Its goal is to label every time unit according to each of the 7 actions in the standard-level hand-hygiene task described in Chapter 3.3. The design of this classifier is considered in Section 5.3 below. In particular, as we show in Section 5.3.1, the chest camera view provides rich details for hand actions during the hand-hygiene period. Therefore, this classifier is well suited for identifying the actions that the earlier classifiers have not considered, namely, the 4 actions “touch faucet with hand”, “rub hands with water”, “apply soap”, and “rub hands without water”.

However, some actual hand-hygiene periods may have been misidentified as non-hygiene in the first system stage. This is illustrated in Figure 5.4. To compensate for this possibility, we expand the hand-hygiene temporal region by applying an iterative search method. In particular, we apply the chest refinement classifier to all time units to the left and right of any identified hand-hygiene time unit. This continues recursively, until this classifier labels a time unit as either a non-hygiene action or “touch faucet with elbow”, and all the initially-labeled hand-hygiene units have been processed. An illustration of the final temporal region searched by the chest refinement classifier is indicated in Figure 5.4b.

To summarize, the overall system takes as input the untrimmed video that contains both hygiene and non-hygiene actions. Four actions must be identified as to whether they happen or not. These are the actions of “dry hands”, “touch faucet with elbow”, “touch faucet with hand”, and “apply soap”. The first is detected by the chest coarse classifier, the second

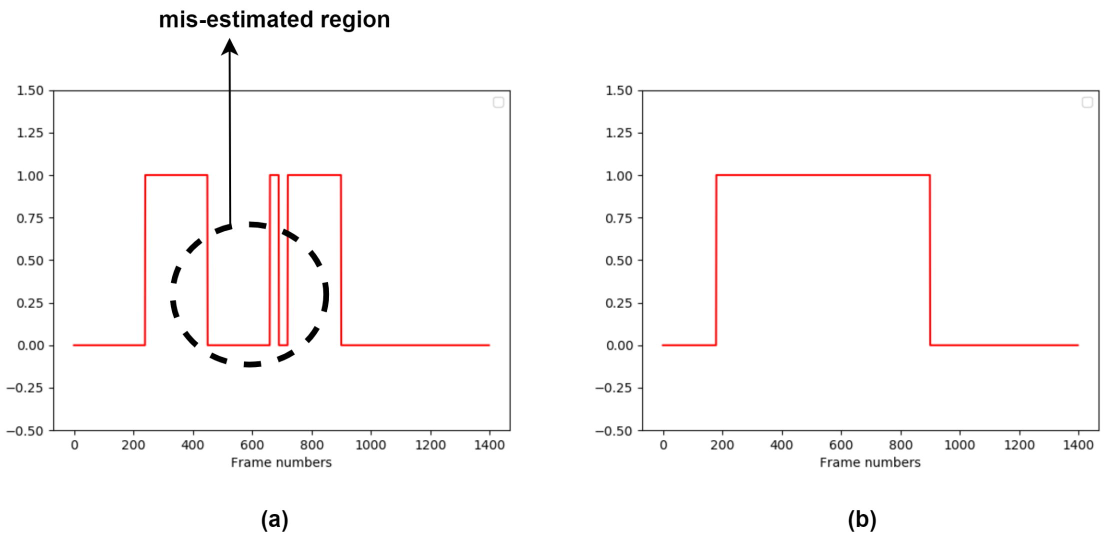


Figure 5.4. Temporal location of hand-hygiene temporal (**a**) produced by the first system stage (**b**) and final searched region. Each plot shows the confidence score of deciding "hand hygiene" or not as a function of the number of frames. The dashed circle indicates the region that is misidentified by the wall coarse classifier.

by the refinement wall classifier, and the latter two by the refinement chest classifier. For the remaining two actions, "rub hands without water" and "rub hands with water", both of which are identified by the refinement wall classifier, it is important to verify that they lasted for at least 20 s. During our final system evaluation in Section 5.4, we will consider estimates of this duration as a measure of performance. However, in the next section, which explores detailed questions about how to design each of the four classifiers, we consider only detection and recognition accuracy.

5.2 Evaluating Hand Hygiene for a Real Application

In this chapter, we focus on the standard-level hand-hygiene task, which considers 7 different action types. In a real application for hand-hygiene verification, a user would like to know if some actions happened for a sufficiently long time, and if other actions happened at all. Therefore, performance evaluation of a system requires distinct measurements depending on the action type. For example, both "rub hands with water" and "rub hands without water" reflect the participants' effort to clean their hands. Thus, it is important for us not only to detect the existence of these actions, but also to determine how long they last. Three actions, “touch faucet with elbow”, “apply soap”, and “dry hands”, help sanitize and prevent re-contamination; therefore, we only need to confirm the existence of these actions. However, “touch faucet with hands” actually re-contaminates hands, and should be identified if it happens. Meanwhile, the background non-hand-hygiene actions do not influence the hand washing quality, and these are included in the set of actions for completeness.

Based on these observations, we evaluate the hand-hygiene performance of a participant by evaluating whether a correct decision was made regarding which action occurs during each second of the video. Thus, we divide each target video into non-overlapping units of 30 consecutive frames, which corresponds to 1 s in time. Each unit is labeled with only one action type by counting the most frequent action type of each frame among all 30 frames. This assumes a detector makes one prediction during each unit. To achieve this, a detector can predict an action for 30 frames individually and average the confidence scores to create a prediction result for the unit, for example.

The top and bottom of Figure 5.5 illustrate the unit-based ground truth and prediction results, respectively. The region between the dashed lines is the intersection between ground truth label and system prediction of a hand-hygiene action. If, for this particular action, we only need to assess whether it happened or not, we can simply verify the existence of an intersection region. However, if for this action, we need to assess how long it lasted, we evaluate the prediction result by the Jaccard index, which is also known as the Intersection Over Union (IOU). The Jaccard index is defined as $J = (R_n \cap B_n)/(R_n \cup B_n)$, where R_n is the number ground-truth units and B_n is the number of predicted units, for a particular action.

5.3 Design and Evaluation of Individual Classifiers in the Two-Stage System

In this section, we explore the design of each individual classifier in the the two-stage system described in the previous section. We take an experimental approach to address the following questions:

- Question 1: Which camera is most informative for which actions? (Section 5.3.1)

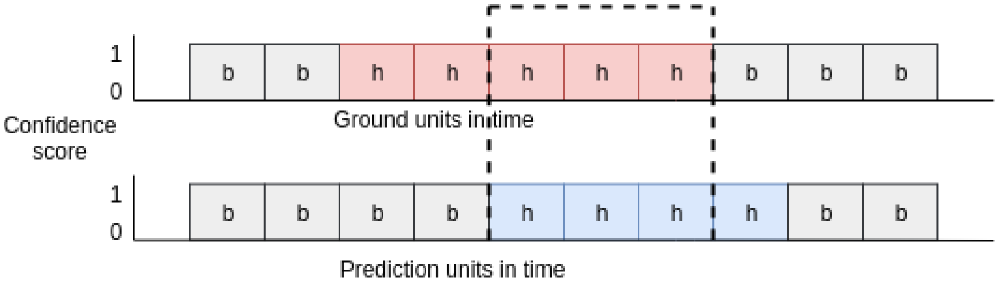


Figure 5.5. Unit-level prediction. **Top**: ground truth labels. **Bottom**: prediction results. Each rectangle represents consecutive 30 frames. Label “b” is a “non-hygiene” action and label “h” a is “hand-hygiene” action.

- Question 2: How much computational complexity is required for the standard-level hand-hygiene task? What models should we use? How deep? How much accuracy do we lose if we use classifiers with lower computational complexity? (Section 5.3.6)
- Question 3: Is RGB information sufficient or should we include motion information? (Section 5.3.7 and part of 5.3.9)
- Question 4: To coarsely recognize hand-hygiene temporal regions from untrimmed input video, what model should be used for the wall camera? (Section 5.3.9)
- Question 5: To coarsely recognize the single action of “dry hands”, can we use hand-crafted features or would a CNN perform better? (Section 5.3.10)

First in Section 5.3.1, we compare the performance of using wall camera, nose camera videos, and chest camera videos on solving the standard-level hand-hygiene task which recognizes 7 hand-hygiene actions. After determining that the chest and wall camera are the best two camera views, we explore how sophisticated a model needs to be to solve this task for both camera views in Section 5.3.5. This design comparison across well-known CNN structures leads us to select the models for both refinement classifiers. Finally in Section 5.3.8, we design simpler structures for the coarse classifiers for both these camera views.

5.3.1 Camera-View Comparison for Hand Hygiene

As indicated in previous Chapter 3.5, the ”nelson100” dataset consists of video data from a third-person camera and two egocentric cameras. As a result of the camera placement, each camera view has its advantage for recording certain types of actions. As demonstrated in Figure 5.6b, the chest camera view fails to capture the participant’s arm when the pose involves body motion. In addition, in Figure 5.6c, the wall camera view can only record half of the participant’s body when the participant walks away from the sink. The nose camera is designed to record the same video as the user’s eye view. However, due to the nature of our hand-hygiene task in a narrow space, the user’s gaze does not always align with the camera view. Thus, some of the nose camera videos do not actually show the hands; an example is shown in Figure 5.6a.

Our goal in this section is to better understand which camera view is most effective for each action. Therefore, we apply a single model for each camera; the input to each model the set of RGB images from a trimmed video, and the output is one of the 7 actions for the standard-level hand-hygiene task. Since, for hand-hygiene actions, the majority of the content is composed of hand-to-hand and hand-to-arm interactions and there are no salient objects to help distinguish each action type, we choose to use a single deep-learning based model to learn an efficient representation. In this chapter, we follow the idea of the two-stream network [35] to consider a 2D CNN structure that processes both RGB image and motion information for activity recognition. In particular, for this exploration, we apply ResNet152 [21] to be consistent with our previous work [86] where we considered only the view from the chest camera.

In the following subsections, we describe our experimental design (Section 5.3.2), the model training (Section 5.3.3), and finally the results of applying the model to our dataset (Section 5.3.4).

5.3.2 Experimental Design

To solve the standard-level task defined in Section 3.3.2, we manually labeled our hand-hygiene video data at a frame level with 7 action types to create ground truth. Since our

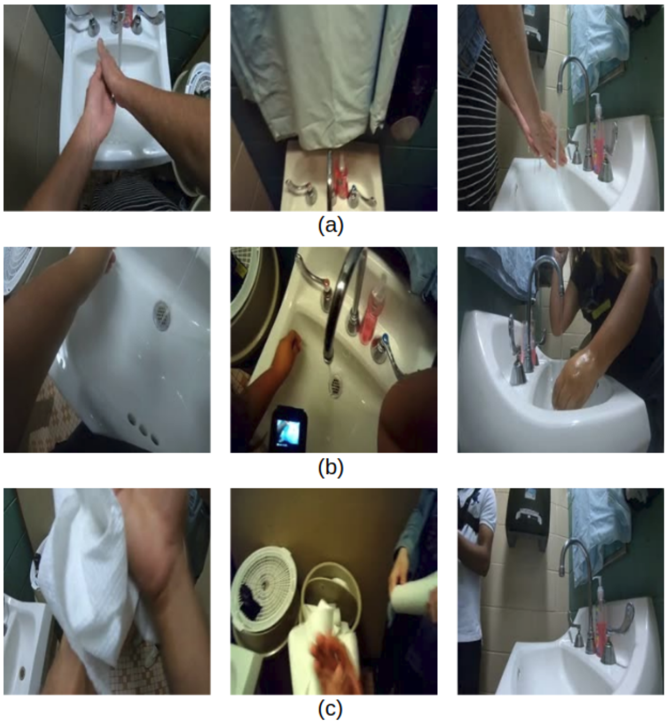


Figure 5.6. Images from chest camera (left), nose camera (middle), and wall camera (right): (**a**) “rub hands without water”; (**b**) “touch faucet with elbow”; (**c**) “dry hands”.

100 participants were each recorded twice, this provides us with around 200 videos for each camera view. To verify the system’s robustness against different subjects, the training, validation, and testing data were randomly selected among 100 participants with 66, 12, and 22 people, respectively. This partition and random selection was performed 5 times to create the 5 trials that we will refer to throughout this chapter. We further trim each video into clips based on the frame-level ground truth of actions types. As a result, each contains only one action type for its duration.

5.3.3 Model Training on RGB Images

For each camera view, we begin with a ResNet152 model that was pre-trained on ImageNet [93]. We fine-tune the model so its last fully-connected layer outputs one of our 7 action classes. As a result that our data were all recorded in two nearly identical environments and because we are interested in a pure comparison between the different camera views, we do not apply data augmentation techniques like image scaling or random crop for training. However, since the wall camera was placed on one side in the first room and the opposite side in the second room, we horizontally flip its frames to the same direction. This improves both training and testing efficiency.

The training hyperparameters for each camera view are set to the same for comparison, although they could be further optimized. However, our goal for this experiment was to compare the efficiency of each camera when processed by the same CNN architecture. Each model was trained for 250 epochs using a Stochastic Gradient Descent (SGD) optimizer with learning rate 0.001. The learning rate was decreased by a factor or 10 at 100 and 200 epochs. The batch size is 25, and each sample in a batch is a randomly selected video frame from a trimmed training video clip.

5.3.4 Model Evaluation on RGB Images

Testing is performed on each trimmed video clip. The trained model is applied on every frame of a test video clip, followed by a softmax function. The average score among all frames indicates the prediction result. The testing results of all 7 actions averaged over the 5 trials is shown in Table 5.1, for each of the three camera views.

Table 5.1. Classifier accuracy for all three camera views; seven actions

Action\Camera	Wall RGB	Chest RGB	Nose RGB	Number of clips
Faucet elbow	94.03%	85.07%	83.58%	67
Faucet hand	91.76%	94.59%	93.41%	425
Rub water	93.73%	92.68%	91.38%	383
Rub nowater	87.21%	94.06%	86.30%	219
Soap	93.19%	88.48%	96.34%	191
Dry hand	54.76%	90.48%	76.19%	210
Non-hygiene	91.36%	93.39%	92.03%	590
Average	88.01%	92.57%	90.12%	NaN

Comparing across the rows of the table, we can see that the chest view outperforms the other views for recognizing actions with detailed hand-hand interactions, such as “rub hands without water”, “touch faucet with hands”, and “dry hands”. However, we observe a small performance drop for the chest view on the actions “rub hands with water” and “apply soap” compared to the wall view, because the chest camera does not always capture the hand regions.

Compared to the chest camera view, the nose camera view achieves less accurate results for all action types except “apply soap”. This is because the nose camera is mounted above the chest camera, so it is easier for the nose camera view to capture the scene when the user applies soap.

On the other hand, the wall camera can also predict many of the action types within its viewing range, especially when an action contains the body motion from a participant. As the shown in the heatmaps in Figure 5.7a,b,and c, the chest camera can accurately capture human hands. But due to the limitation of the camera angle, the chest camera cannot capture the salient region of the arm as well as the wall camera can. Thus, the CNN model from the chest camera view predicts the “touch faucet with elbow” action by focusing on the sink region. This may be effective for the current action set where only one action contains significant body motion, but in general, the chest camera view will not be robust to body actions. As a result, the wall camera outperforms the chest camera by about 10% on the “touch faucet with elbow” action. Another drawback of the wall camera view is also obvious if we consider the “dry hands” action. For this, a participant is likely to move around the room while they wipe their hands with a paper towel. The failure to track the participant causes the wall camera’s low prediction accuracy of 54% on detecting “dry hands”.

The last row of Table 5.1 summarizes the average accuracy over all 7 actions for each camera view. Due to the disparity in the number of videos for each action type, the wall camera’s advantage is not reflected by the average accuracy. However, it is undeniable that the wall camera performs best for recognizing actions related to body motion. Overall, to answer Question 1 in the beginning of Section 5.3, we conclude that the chest camera view is effective at recognizing hand-hand related actions and the wall camera view can be applied to recognize body actions to monitor the presence of a participant within its viewing area.

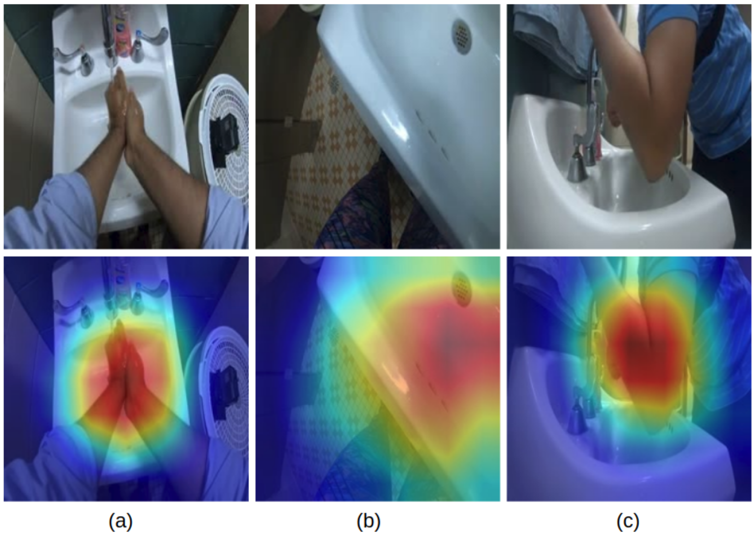


Figure 5.7. Grad cam [54] results of (**a**) chest cam: rub hand with water, (**b**) chest cam: touch faucet with elbow, (**c**) wall cam: touch faucet with elbow, when applied to the respective model.

5.3.5 Model Comparison for Refinement Classifiers

In the last subsection, we applied the ResNet152 model to explore the relative advantages of using the wall and chest camera views for a standard-level hand-hygiene task. ResNet152 is a relatively complex model; therefore, in the following subsections, we evaluate the performance of different CNN models on the standard-level hand-hygiene task (Section 5.3.6). In addition, we also evaluate whether adding optical flow improves performance for the chest camera (Section 5.3.7). Together, these experiments inform the design of the refinement classifiers in the second stage of our system.

5.3.6 Model Comparison for RGB Images

The models we consider are VGG19, VGG16, and VGG11, which are variants of the VGG network [20] with high to low structure complexity. Again, we apply pre-trained models and fine-tune the last fully-connected layer to output 7 action classes. For comparison purposes, the training settings of VGG19, VGG16, and VGG11 are exactly the same as ResNet152 from Section 5.3.3.

The testing results of data from trial 1 (only) are listed in Table 5.2. The model accuracy is evaluated as the number of true positive and true negative predicted video clips divided by the total number of video clips. As we can see, the overall detection accuracy from both wall camera and chest camera view have minor variations as the model complexity drops, and the final performance of VGG11 is similar to that of ResNet152. This is mainly because the scenario we currently consider is limited to a public bathroom environment with similar camera angles. Therefore, the less complex CNN architecture can still achieve reasonable detection accuracy. This decision may have to be revisited if the model is tested with different camera angles in different environments. However, for the current scenario, these results suggest that the answer for Question 2 in the beginning of Section 5.3 is that VGG11 with an input RGB image is adequate for the standard-level hand-hygiene task as a refinement classifier.

Table 5.2. Classifier accuracy for models of different complexities; seven actions

Model\Detection	Accuracy
Chest ResNet152	93.44%
Chest VGG19	93.68%
Chest VGG16	92.74%
Chest VGG11	94.38%
Wall ResNet152	86.65%
Wall VGG19	88.06%
Wall VGG16	88.52%
Wall VGG11	87.12%

5.3.7 RGB and Optical Flow Comparison

Based on the result from Section 5.3.3, we have demonstrated that chest camera RGB-image model has an advantage for analyzing the hand-hygiene actions that specifically concern the hands. Many previous works have demonstrated the importance of including motion information for egocentric activity recognition [32], [33] and third-person activity recognition [35]. Thus, we conducted an experiment to explore the degree to which incorporating motion information helps to interpret egocentric hand-hygiene videos. For this, we create optical flow images using the TV-L1 optical flow [90] implementation [36].

Training: Similar to the chest camera RGB model, the chest camera optical flow model still takes a pre-trained ResNet 152 network and fine-tunes the last fully-connected layer for our 7 action classes. The first convolutional layer is revised to take as input 10 frames of horizontal and vertical optical flow images. The model is trained with 350 epochs and learning rate 0.001. The learning rate was decreased by 10 at 200 and 300 epochs. The remaining hyperparameter settings are the same as above.

Testing: The testing step follows the same procedures as Section 5.3.3 on each trimmed video clip, except that the trained model processes every 10 optical flow frame pairs instead of a single RGB frame. The testing results of all 7 actions averaged across the 5 trials are listed in Table 5.3.

Table 5.3. Classifier accuracy for the chest camera with spatial and temporal models; Seven actions.

Action\Camera	Chest RGB	Chest flow
Faucet elbow	85.07%	83.58%
Faucet hand	94.59%	89.41%
Rub water	92.68%	92.43%
Rub nowater	94.06%	94.98%
Soap	88.48%	76.96%
Dry hand	90.48%	94.28%
Non-hygiene	93.39%	95.59%

We see from the table that for all 7 actions classes, using optical flow does not provide significant improvements for recognition. In addition, in our previous work [86], fusing the RGB and optical flow models did not show a meaningful boost in recognition accuracy. Therefore, to answer Question 3 in the beginning of Section 5.3, we conclude that the adding temporal motion information is not necessary for the standard-level hand-hygiene task.

5.3.8 Model design for coarse classifiers

In this section, we develop classifiers to recognize coarse hand-hygiene actions in the first system stage described in Section 5.1. The actions to be recognized in this stage are easier to distinguish than in the second stage. Thus, we can apply less complex CNN architectures in this stage. The designs for the coarse wall and chest classifiers appear in Sections 5.3.9 and 5.3.10, respectively.

5.3.9 Wall Camera Coarse Classifier

The wall camera (a third-person view camera) is placed on a flat platform in our experiment, so it captures the participant's actions near the sink from a close range. Due to the limitation of the camera angle, this camera view cannot capture hand actions in detail; however, we showed in Section 5.3.1, 5.3.2, 5.3.3, and 5.3.4 that it is useful at providing participant's body actions and location. We believe the wall camera view is suitable for coarsely localizing both the hand-hygiene period and the action of touching the faucet with an elbow. Thus, the goal of the wall coarse classifier is to predict 3 types of action classes: "touch faucet with elbow", hand-hygiene, and non-hygiene. The non-hygiene actions that happen before the first hand-hygiene action is identified are categorized as pre-hygiene actions, while those identified after are called post-hygiene.

Based on the observations in [33], [75], shallow CNN models with 2 convolutional layers and 2 fully connected layers are effective for recognizing hand actions in both egocentric and third-person camera views. However, simply applying an RGB-based model ignores potentially useful information about motion. Indeed, motion information [35] or multi-modality depth and skeleton [44] information has been shown to improve detection accuracy for action recognition. Therefore, we propose to use a simple CNN model for this coarse classifier to quickly process the untrimmed video, and we also explore whether the addition of motion information can improve accuracy. Designs based on these two considerations are described next. The performance of these design choices is then compared to finalize our design of the wall camera coarse classifier.

Model for RGB images: To ensure a low computational cost and fast processing speed for this coarse processor, we are inspired by the tiny image dataset CIFAR 10 [99]. Thus, we explore CNN structures that take as input a down-sampled image of size 32 × 32. The basic architecture of the model follows the design of the VGG networks with a 3 × 3 kernel size and max pooling kernel size of 2 × 2 and stride 2.

To explore how variations of the CNN architecture affect the detection performance, we evaluate three structures. The first structure consists of 5 groups of convolutional layer, max pooling layer, and RELU non-linearity followed with a fully connected layer and softmax as indicated in Figure 5.8. The second structure is modified from the first by changing the last group of convolutional layers with 512 output channels into a fully connected layer, which results in 4 groups of convolution layers followed by 2 fully connected layers. In the last setting, we attempt to further reduce the model complexity by replacing the 4th group of convolutional layers with depthwise separable convolutional layers [100].

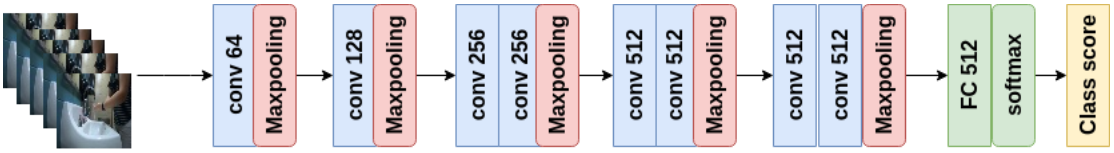


Figure 5.8. Coarse wall classifier: Model with RGB only.

Features to describe supplemental silhouette information: Since our wall camera is stationary, foreground–background segmentation is a powerful method to understand moving objects in the scene. Three methods we consider here are an adaptive background learning [101] method, the Fuzzy Choquet integral [102] and SuBSENSE [103]. We compute these using the implementation in [104].

In addition, we consider the motion history image (MHI), which is an efficient method to create a temporal template that indicates motion [105]. MHI encodes the motion density from past frames into the current frame. It is a robust representation of human gestures especially under static camera view. This feature has three parameters to set: the motion history duration τ, the decay parameter δ, and a threshold ξ [105]. In our experiment, these are set to $\tau = 5$, $\delta = 1$, and $\xi = 20$, respectively.

Figure 5.9 shows the three different background detections along with the MHI for one frame. Compared with the fuzzy method and SuBSENSE method in Figure 5.9, the basic adaptive background learning method and motion history image appear to be more robust for isolating the human silhouette. Overall, the motion history image shows the best results of tracking the human silhouette with less background noise. Therefore, we select motion history image as an additional input for the wall camera coarse classifier.

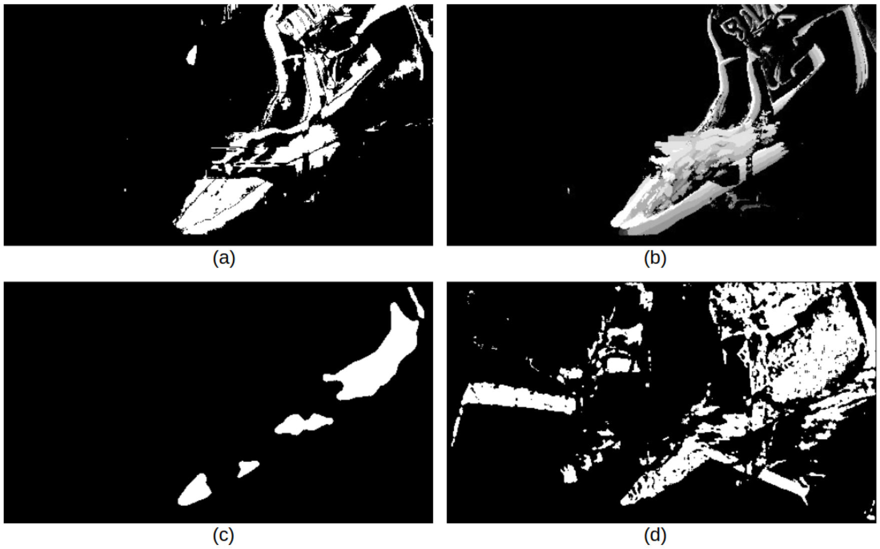


Figure 5.9. (**a**) Adaptive background learning [101]. (**b**) Motion history image. (**c**) Fuzzy Choquet integral [102]. (**d**) SuBSENSE [103].

Model for both RGB images and MHI: In addition to the three RGB-only models mentioned above, we also consider a model that takes RGB and the motion history image as indicated in Figure 5.10. MHI is computed using RGB images of size 224 × 224 and down-sampled to 32 × 32 size to be input to the model. We use the same architecture for both the MHI and the RGB image. The output feature maps from the last convolutional layer are concatenated with the feature maps from RGB model and passed into the final fully-connected layer.

Model training: All the training and testing are performed on the first data trial defined in Section 5.3.2. We applied the same hyperparameters settings for the model that uses only the RGB as well as the model that uses both RGB and MHI. All models are trained from scratch without using pre-trained weights. For fast convergence and to reduce over-

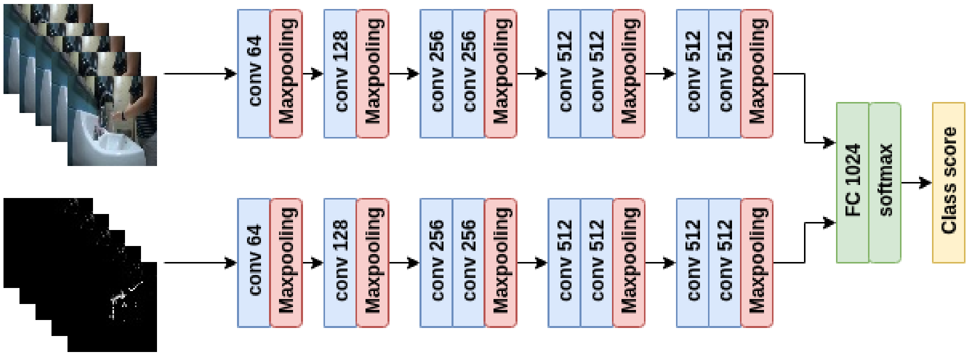


Figure 5.10. Coarse wall classifier: Model with RGB and motion history image (MHI).

fitting, batch normalization [106] is applied to the output of every convolutional layer before it enters the ReLU non-linearity. Each model is trained by 250 epochs using Stochastic Gradient Descent (SGD) optimizer, cross entropy loss, and learning rate 0.001. The learning rate was decayed by 10 at 100 and 200 epoch. Batch size is selected at 128 and each sample in the batch is a randomly selected video frame from a trimmed training video clip.

Testing: To compare the performance of these structures, we consider untrimmed testing videos and predict 3 action types: "faucet elbow", hand-hygiene, and non-hygiene. We evaluate the performance of each structure with two metrics: the frame-level accuracy and the unit-level accuracy. Frame-level accuracy is computed by the sum of true positive and true negative predictions over all frames in the video. Instead of predicting every single frame, the unit-level prediction starts by cutting the untrimmed video into non-overlapping units of consecutive 30 frames, which is 1 s in our video. Then, we make a prediction of an action class for each unit by averaging each frames prediction confidence score. The unit-level accuracy is also computed as the sum of true positive and true negative units over the total number of units in the video.

The prediction results of the three structures for RGB-only, as well as for the RGB + MHI model are shown in Table 5.4. As can be seen, among the three different structures that have RGB-only input, the structure with 5 groups of convolutional layers combined with 1 fully connected layer has better performance than others. Notably, the structure with depthwise separable convolution experienced a large performance drop. In addition, the structure with both RGB + MHI achieves better performance than using only the RGB modality. These two architectures are further compared in Table 5.5, which indicates that incorporating the MHI markedly improves performance for the “faucet elbow” action and the hand-hygiene action. Therefore, to answer Question 4 in the beginning of Section 5.3, we select the RGB + MHI to be the model for the wall coarse classifier.

Table 5.4. Coarse wall classifier: Performance of RGB and RGB + MHI structures. **conv:** convolutional layer. **dw conv:** depthwise separable convolution. **fc:** fully connected layer.

Structure\Evaluation	frame-level	unit-level
RGB, 5 groups conv, 1 fc	89.96%	90.41%
RGB, 4 groups conv, 2 fc	86.45%	87.13%
RGB, 4 groups dw conv, 2 fc	83.50%	84.35%
RGB + MHI, 5 groups conv, 1 fc	91.21%	92.42%

Table 5.5. Coarse wall classifier: Performance of RGB and RGB + MHI between 3 actions at unit-level, both RGB and RGB + MHI use 5 groups of convolutional layers and 1 fc layer setting.

Action\Structure	RGB	RGB + MHI
faucet elbow	75.00%	85.71%
hand-hygiene	89.52%	93.08%
non-hygiene	89.17%	87.64%

5.3.10 Chest-Camera Coarse Classifier

As described in Section 5.1, after applying the wall coarse classifier on the untrimmed hand-hygiene video, three temporal periods are identified. In the “post-hygiene” period, the participant is expected to dry their hands with a paper towel. However, the wall camera cannot predict accurately whether the “dry hands” action exists or not. Therefore, in this section, we design a system to process the videos from the chest camera, but only in the post-hygiene temporal region to detect the “dry hands” action. Since we only need to confirm if the “dry hands” action happens or not, we anticipate that a low-complexity model is sufficient for this task. A low-complexity model is also desired because the exact location of the “dry hands” action could be anywhere in the long post-hygiene region.

Several options are available for low-complexity models. For temporal segmentation of egocentric videos, Bolaños et al. [107] apply color descriptors to detect the action “in transit”. They computed the color histogram for each video frame and use the difference between histograms as a feature to describe changes in the camera angle. Moreover, Azad et al. [108] argued that hand-crafted features have an advantage over deep-learning methods when applied to small data sets. They achieve good performance by using Histogram of Oriented Gradients (HOG) [13] and Local Binary Pattern (LBP) [109] descriptors on a depth motion map to recognize hand gestures. Therefore, to recognize the “dry hand” action within the post-hygiene period, we propose to test both deep-learning features and the hand-crafted features of color histogram, HOG, and LBP.

Training: Similar to the experiments in the previous section, all training and testing are performed on the first data trial defined in Section 5.3.2. For the deep-learning model, we apply the same architectural structure of five groups of convolutional layers with a fully connected layer chosen for the wall coarse classifier in Section 5.3.9. The hyperparameters and training steps are also the same as mentioned previously. Since the model is designed to predict only “dry hand” and “non-hygiene” actions, we modify the last fully-connected layer to output a single confidence score using the sigmoid function. The model applies binary cross entropy as its loss function.

We compute the hand-crafted features on an RGB image with size 224 × 224. We compute the color histogram separately on 4 non-overlapping spatial regions of the image using the hue, saturation, value (HSV) color space. The histogram for the lightness (i.e., value) channel has 8 evenly-spaced bins while the histograms for the other two each have 3 evenly-spaced bins. Concatenating all the histograms yields a final color feature with 288 dimensions. We also extract HOG [13] and LBP [109] descriptors for comparison experiment. To classify the action “dry hands” from the post-hygiene period using these hand-crafted features, we train a Random Forest.

Testing: We perform frame-level prediction to compare both the deep-learning model and hand-crafted feature Random Forest classifiers for the two ground-truth actions labeled either “dry hands” or “non-hygiene”. We evaluate the methods using precision, recall, and accuracy at the frame level by considering “dry hands” as a positive sample and “non-hygiene” as a negative sample.

From the results in Table 5.6, we observe that the CNN model performs best among all methods, even though the recall is a few percent lower than the other methods. As discussed in Section 5.2, the goal of detecting the “dry hands” action is to verify that it took place. If a non-hygiene action is mistakenly classified as drying hands, the system will fail to correctly assess the entire hand-hygiene process. Therefore, recall is less important than precision. To answer the Question 5 in the beginning Section 5.3, we conclude that the low-complexity CNN model is the preferred model to recognize “dry hands” action in the post-hygiene period.

Table 5.6. Coarse chest classifier: Prediction result for the ”dry hand” action. Positive class: dry hand, negative class: non-hygiene.

Model\Evaluation	Precision	Recall	Accuracy
32x32 CNN	79.54%	61.31%	81.63%
Color histogram	54.32%	64.33%	70.49%
HOG	49.53%	65.38%	66.65%
LBP	50.51%	66.99%	67.58%

5.4 Performance of the Two-Stage Hand-Hygiene System

In the previous Section 5.3, we explore the individual designs for each of the coarse and fine classifiers. The overall two-stage system for recognizing and evaluating hand-hygiene activities within untrimmed video appears in Section 5.1. In this section, we evaluate the performance of the entire two-stage system for detecting actions within untrimmed hand-hygiene videos. We first explain the experimental protocol, then report our experimental results for the overall system performance.

5.4.1 Experimental Protocol

In the first stage of the system, the wall coarse classifier uses the CNN model with RGB and MHI as inputs, and the chest coarse classifier applies a CNN model for binary classification on the “dry hands” action. In the second system stage, both the CNN model for wall and chest camera view are applied with VGG11 network architecture.

To create a point of comparison, we consider a baseline system that applies the second-stage VGG11 networks to the entire chest camera video and wall camera video. Each classifier densely processes the entire untrimmed video using non-overlapped 30-frame units. The classifier for the wall camera is only responsible to detect the action “touch faucet with elbow” action, and the classifier for the chest camera is applied to detect all other actions.

Recall that our goal for the two-stage system was to achieve similar performance to the baseline system, but with less computation. The baseline system applies VGG11 on every 30 consecutive frame units throughout the entire video, while the two-stage system applies VGG11 only when a simpler classifier would be insufficient. Therefore, we expect that both systems will achieve very similar detection accuracy. If so, it demonstrates that our coarse classifier successfully localizes the crucial hand-hygiene temporal parts, reducing the overall system complexity without sacrificing performance.

The overall performance of the two systems is evaluated on the first trial defined in Section 5.3.2.

5.4.2 Results and Discussion

We evaluate the overall performance of this system in two parts. In the first part, we evaluate only the two actions that require an estimate of how long they last; these actions are “rub hands with water” and “rub hands with no water”. We measure their detection performance using the Jaccard index, applied for units of 30 consecutive frames. In addition, we measure the average prediction error by computing the absolute difference between the detected time duration and ground-truth duration, averaged across all test videos. In the second part of evaluation, we consider the remaining actions for which we only need to confirm whether or not they happened. We evaluate these simply using the accuracy across all test videos, computed by dividing the number of correct predictions divided by the total number of predictions.

Tables 5.7 and 5.8 show the evaluation of the rubbing actions. As we can observe, both the baseline and proposed two-stage system have similar performance in terms of both the Jaccard index and error in the estimated duration. This indicates that the first stage of the two-stage system could successfully localize the hand-hygiene period within the untrimmed video; the estimates of how long each action happens is consistent. Moreover, the ground-truth statistics for these two actions across the entire data set is shown in Table 5.9. Given the large average duration and standard deviation of these two actions, an estimation error of around 2 s is an reasonable result. To obtain further improvement, increasing the complexity of the CNN model and optimizing the hyperparameters might reduce the average error.

Table 5.7. Two-stage system: Average Jaccard Index

Action\Model	Baseline	Two-stage system
Rub without water	0.8036	0.8036
Rub with water	0.8299	0.8307

Table 5.8. Two-stage system: Average mis-prediction in seconds

Action\Model	Baseline	Two-stage system
Rub without water	2.39	2.39
Rub with water	2.23	2.20

Table 5.9. Statistics for the duration of each action in the ground truth

Action	Mean (secs)	Std. dev. (secs)
Rub without water	16.40	11.83
Rub with water	13.72	7.93

Table 5.10 demonstrates the accuracy across all test videos for the discrete actions of ”apply soap”, ”dry hands”, and ”touch faucet with elbow”. These three actions have nearly identical performance for both the two-stage system and the baseline system. Therefore, no performance is lost by applying the low-complexity model for temporal localization.

Overall, the two-stage system and the baseline system achieve similar performance for recognizing each action and for estimating the duration of rubbing. However, with the

Table 5.10. Two-stage system: Average detection accuracy

Action\Model	Baseline	Two-stage system
Soap	86.36%	86.36%
Dry hands	97.73%	93.18%
Faucet elbow	95.45%	95.45%

support of the low-complexity CNN models for localization, the two refinement classifiers in our two-stage system only process 67.8% of the frames in the untrimmed videos. This is in contrast to the baseline that must densely process 100% of the frames in the untrimmed video, regardless of the duration of the hand-hygiene activity. Further, it should be noted that the videos we collected in this project were specifically designed to analyze hand hygiene, and as such they contain very little time spent on non-hygiene actions. Specifically, the average non-hygiene actions occupy only 28.1% of the total video duration, with the remaining 71.9% containing hand-hygiene activities. In more typical situations, where the hand hygiene would take less time relative to the overall collection of activities, the computational savings achieved by the temporal localization in our two-stage system would increase dramatically.

5.5 Conclusion

In this chapter, we introduce the task of hand-hygiene action recognition from untrimmed video. We approach this problem by designing a system that performs hand-hygiene recognition at the standard level. To explore the efficiency of using different camera views on recognizing 7 hand-hygiene actions, we used the data in our ”nelson” dataset with three cameras and 100 participants. Using this dataset we are able to explore different deep-learning models on our hand-hygiene dataset with both egocentric and third person camera views. The results indicate both these camera views have their own unique advantages for recognizing certain action types. Thus, it is important to use both camera views for hand-hygiene action recognition.

Moreover, we also explore the realistic scenario in which we recognize hand-hygiene actions inside untrimmed video. We design a two-stage system to localize the target hand-hygiene regions and we apply deep-learning models from two camera views for the final recognition. In the first stage, a low-complexity CNN model is applied on the third-person view to segment the untrimmed video into three temporal periods. In the second stage, we assign these temporal periods to more complex CNN models trained for different camera views, so that each model only has to recognize the actions suited for that camera view. In the final evaluation, our two-stage system achieves similar performance to the baseline, which applies CNN models to densely process every second in the entire untrimmed video. We demonstrate that the two-stage system can efficiently filter out non-hygiene regions so that it only needs to apply complex CNN models to the crucial hand-hygiene temporal regions.

6. HAND-HYGIENE IN CROSS SCENARIO

In this chapter, we further extend hand-hygiene recognition from single scenario into multiple scenarios. In here, the term ”scenario” represent the variation in video data in background environment or camera view or both. The majority content in this chapter is also covered in our work [110]. In previous Chapter 4 and 5, we apply the dataset of ”nelson100”, which is recorded in two separate college bathroom. Even though the scenario changed, but college bathroom shares a common layout and there does not exist a significant difference between those two rooms. To study the feasibility of developing a hand-hygiene system which works in general situation, it is necessary for use to explore the hand-hygiene recognition problem in multiple scenarios.

Therefore, in this chapter's work, we take use of the ”class23” dataset introduced in Chapter 3.6 to address the following questions with experimental approach:

- Question 1: In same scenario, what is the general approach to recognize hand-hygiene actions? Is temporal information necessary? (Section 6.1.1 to 6.1.5)
- Question 2: In cross scenario, does the system designed in the same scenario still work? What is the main issue to prevent it from working? (Section 6.1.6 to 6.1.8)
- Question 3: In cross scenario, how to design a system to utilize multi-modality information for hand-hygiene recognition? (Section 6.2 and 6.3)

6.1 Preliminary exploration on Hand Hygiene with RGB

In this section, we describe experiments on exploring hand-hygiene action recognition with RGB image or video, which is the most commonly used input source for action recognition. The exploration is motivated by our new dataset ”class23”, which includes hand-hygiene video collected under 3 different ”scenarios” as we mentioned in ”class23” dataset in Chapter 3.6. Under ideal experimental setting, the model to recognize hand-hygiene actions are suppose to be developed and deployed onto video data from the same scenario, namely ”room1 camera1”, ”room2 camera1”, and ”room2 camera2” in our dataset. In real life, however, both the camera setting and room layout for hand-hygiene can not always be the same.

We would like to take use of the 3 separately recorded scenarios of hand-hygiene to explore the possibility of cross-scenario hand-hygiene recognition, which our model trains on data from one scenario, and tests on data from another scenario. Especially, our experiments focus on using RGB image or video as model input, since it is the most commonly used modality in many computer vision tasks. We take an experimental approaches to address the following questions:

- In same scenario, how well does RGB modality perform on hand-hygiene action recognition? (Section 6.1.1, 6.1.2, and 6.1.3)
- In same scenario, how well does RGB modality perform on hand-hygiene action detection? (Section 6.1.4 and 6.1.5)
- In cross scenario, does RGB modality still have good performance? What is main reason for RGB modality to fail? (Section 6.1.6, 6.1.7, and 6.1.8)

6.1.1 Same scenario hand-hygiene action recognition

In this section, we explore the performance of using RGB modality for hand-hygiene action recognition on the same scenarios of our ”class23” dataset. Action recognition is a task to take a trimmed video clip which includes only one action, and make a prediction on its action category class. We have explored hand-hygiene action recognition on a uniform college bathroom scenario from our previous work [86] [98]. With our new dataset, we would like to extend this exploration onto 3 different food laboratory scenarios to recognize the four hand-hygiene actions introduced in Chapter 3.6.2. To analyze the efficiency of both spatial and temporal information in hand-hygiene recognition, we compare the performance of CNN with spatial RGB information only with spatio-temporal modeling which combines CNN with Long Short-Term Memory (LSTM) [22] and Temporal Relational Network (TRN) [25].

In Section 6.1.2, we compare the action recognition performance between spatial only model and spatio-temporal model on four hand-hygiene actions. The experiment indicates that the spatial only model is capable to achieve hand-hygiene recognition in same scenario.

In Section 6.1.3, we continue to use the spatial only model on the combination of hand-hygiene and non-hygiene actions to further explore its performance.

6.1.2 Four-task hand-hygiene action recognition

For each of our three scenarios, namely ”room1 camera1”, ”room2 camera1”, and ”room2 camera2”, they all include the 4 hand-hygiene actions: ”touch faucet with hand”, ”rub hands with water”, ”rub hands without water” and ”apply soap”. In this section, we will focus on the action recognition tasks on these hand-hygiene actions. We begin with a 2D CNN ResNet50 model. Due to the limitation of our data collection, we initialize the network with the pre-trained weights from ImageNet [19] and fine-tune it using our data. The final fully-connected layer is replaced with an output of 4 action classes.

Training: For comparison purposes, the training procedures for each scenario are identical. Specifically, the ResNet50 model is trained with 250 epochs with batch size 32 and an initial learning rate 0.001, which decreased by 10 at 100 and 200 epochs. We apply a Stochastic Gradient Decent (SGD) optimizer and cross-entropy loss function. In a training batch of images, each image is selected by random sampling from the input training trimmed video. To avoid over-fitting, we apply data augmentation using multi-scale crop and random horizontal flip duration training as introduced in [111].

Testing: Testing is performed on every trimmed video clip of the same scenario's test dataset. The trained ResNet50 model is applied on every frame of the test video clip, followed by a softmax function. The prediction result for a video clip is the average score among all its frames. The testing results for all three scenarios is shown in the first row of Table 6.1. Across all three scenarios, the performance of applying spatial RGB information only model achieved over 90% accuracy. This indicates that spatial RGB based model is capable for standard-level hand-hygiene action recognition under different food laboratory scenarios.

Besides using spatial 2D CNN with RGB, we also experiment with adding temporal modeling for hand-hygiene recognition. As it has been proved in many works [94] [112], the 2D CNN combines with temporal model create better performance than 2D CNN solely. There-

Table 6.1. Model accuracy for all three scenarios, four hand-hygiene actions and non-hygiene actions. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2. **ALL**: Accuracy of total correct video prediction among all three scenarios. **H**: involve only hand-hygiene actions. **H+N**: involve both hand-hygiene and non-hygiene actions.

Model\Scene	R1C1	R2C1	R2C2	ALL
ResNet50 (H)	91.67%	97.06%	98.11%	95.24%
ResNet50 + LSTM (H)	91.76%	94.59%	93.41%	95.24%
ResNet50 + TRN (H)	93.73%	92.68%	91.38%	95.92%
ResNet50 (H+N)	92.50%	91.67%	95.38%	93.26%

fore, we experiment the efficiency of temporal models by combining them with our previously tested 2D CNN ResNet50. The temporal models we selected are Long Short-Term Memory (LSTM) and Temporal Relational Network (TRN). LSTM is one the most commonly used temporal modeling structures not only in Nature Language Processing (NLP), but also in computer vision. And reference to the work [113], TRN is a high performance structure with expensive number of parameters and computational cost. For temporal modeling, it requires multiple frames input with fixed length to involve temporal information. We are aware that there exists many popular choices for input selection. For example, Temporal Segment Network (TSN) [36] proposes to cut a video input into N segments and apply random sampling at each segment to get fixed length input. Other methods include uniform sampling or select fixed length consecutive frames at the beginning of video are also feasible for input selection. For hand-hygiene action recognition, one of the crucial tasks is to detect how does a rub hands action last. Therefore, it is important to select consecutive input video frames to our temporal modeling. For a video $V = \{f_1, f_2, \cdots, f_n\}$, which f_{i} is the ith frame in V. In training stage, we randomly select a start frame f_s and consecutively sample k frames until f_{s+k-1} as an input for temporal modeling. In our experiment, we select $k = 10$ for both LSTM and TRN model. The 2D CNN ResNet50 is used as a feature extractor by removing the last fully-connected layer and connected to the temporal model. LSTM model includes 10 time steps and generates the prediction result at the last step. For TRN model, we applied the multiscale TRN that builds 2-frames, 3-frames, ..., 10 frames relations with a selection of 3 relations for efficient training and testing as mentioned in the original work [25]. The training and testing setting are the same for both LSTM and TRN for a fair comparison.

Training: The ResNet50 model is trained with 250 epochs with batch size 8 and an initial learning rate 0.001, which decreased by 10 at 100 and 200 epochs. We select to use Stochastic Gradient Decent (SGD) optimizer and cross-entropy loss function. In a training batch of images, each image is selected by random sampling from the input training trimmed video.

Testing: Testing is performed on every trimmed video clip of the same scenario's test dataset. The trained 2D CNN temporal model is applied on every sliding window on test video clip. Each window has fixed size of 10 frame and step size of 1 frame. The prediction result for a video clip is the average score among all the softmax-ed score of each window.

The testing results for all three scenarios is shown in the second and third row of Table 6.1. Comparing the results of using spatial 2D CNN and 2D CNN combined with temporal modeling, we find that adding additional temporal modeling only has small impact on the final prediction accuracy for same scenario hand-hygiene action recognition. Therefore, we conclude that 2D CNN with spatial RGB information is suitable for hang-hygiene action recognition.

6.1.3 Hand-hygiene and non-hygiene action recognition

In the last sub-section, we compared the performance of different model on hand-hygiene action recognition and confirmed that 2D CNN with spatial RGB information as a suitable selection. In this section, we extend our exploration to includes not only hand-hygiene actions, but also non-hygiene actions. During daily hand-hygiene, it is unavoidable for a participant to perform a variety of unexpected actions other than hand-hygiene. For example, in our class23 data collection, students doing hand washing in the same room could potentially talk to each other, and walk past and occlude the camera. Even when a student is doing hand-hygiene, actions like ”swing hands” could still happen; however, this action has no contribution one way or another with respect to hand-hygiene quality. Our dataset are collected in two separate room, and each room has its own non-hygiene action types. As we described in Chapter 3.6.2, data collected in room1 has non-hygiene actions of ”dry hands with paper towel”, ”grab paper towel”, ”occlusion”, and ”swing hands”. In contrast, data from room2 only has non-hygiene actions of ”occlusion” and ”swing hands”. We applied 2D CNN ResNet50 model with spatial RGB image from Section 6.1.2 on each scenario’s data collection. The test result is indicated in the last row of Table 6.1. Compared with the overall accuracy of 95.24% when apply ResNet50 on 4 hand-hygiene actions only, the overall accuracy on all actions dropped around 2%. But we conclude 2D CNN ResNet50 with spatial RGB image still perform adequately on recognizing hand-hygiene actions when mixing with non-hygiene actions.

6.1.4 Same scenario hand-hygiene action detection

In this sub-section, we continue our exploration of same scenario hand-hygiene on a more difficult task, action detection. Instead of trimmed video clips that contain one action from begin to end, action detection processes untrimmed videos which include a mix of multiple actions. Therefore, the goal is not only to predict an action class label, but also predict the temporal location of each action. A typical approach in the literature [7][6][68] is to create temporal proposals or apply an end-to-end training structure to solve this problem. However, in hand-hygiene, we focus on the action in terms of one second. Instead of generating multi-scale temporal proposals, it is more efficient to apply fixed size sliding window and recognize actions in each window individually. Also, hand-hygiene video for a person is usually last 1 to 2 minutes long. Compare to large untrimmed video which might last half an hour, hand-hygiene untrimmed video has a short duration. Therefore, we propose to implement action detection by using action recognition model and densely processing the entire untrimmed hand-hygiene video. To evaluate the performance, we propose to use three different metrics: frame-wise accuracy, window-wise accuracy, and task-wise accuracy, as indicated in Section 6.1.5.

6.1.5 Method and Evaluation

We apply the 2D CNN ResNet50 with spatial RGB information introduced in Section 6.1.3 as the action recognition model. The model densely processes every frame of the input untrimmed video and predicts an action class category for every frame.

Metrics: For evaluation, we propose to use three different metrics: frame-wise accuracy (F-acc), window-wise accuracy (W-acc), and task-wise accuracy (T-acc). Frame-wise accuracy directly compares the frame level prediction result with ground truth label. Window-wise accuracy cuts the untrimmed video into non-overlapping fixed size windows, and averages all the frame prediction results in a window as the window's prediction result. Finally, the window-wise accuracy is computed by comparing the window level prediction with ground-truth window level label. This metric is designed to evaluate the ”seconds duration” in detecting hand-hygiene actions. Since our video was recorded under 30 FPS, the window size is set to 30 frames.

Finally, the task-wise accuracy is particular designed for hand-hygiene actions. Since the purpose of doing hand-hygiene recognition is to report the participant's hand-hygiene quality, it is necessary to have a metric directly connects to that purpose. The task-wise accuracy includes four standards: (1) mis-detection seconds for ”rub hands with water”. (2) mis-detection seconds for ”rub hands without water”. (3) existence of ”touch faucet with hand” after the last hand to hand (rub hands with or without water) action. (4) existence of ”apply soap” in between of a ”rub hands with water” and a ”rub hands without water” action. During the evaluation of the task-based accuracy, we applied a tolerance towards the ”transit” actions within our untrimmed videos. The transit action usually occurs in the middle of two hand-hygiene actions, for example, after rubbing hands in water, a person will move his/her hands toward the soap. This intermediate actions are the transit actions. The action usually last for few frames and has no crucial influence to hand-hygiene quality. Thus, we tolerate these actions by not counting them into our evaluation.

Experiment results: The test is performed on every untrimmed video clip of the same scenario's test dataset. The experimental results for all three metrics are shown in Table 6.2 and 6.3. The results indicate our strategy of using an action recognition model combined with a sliding window produces reasonable action detection accuracy of about 80%. A visualization of action detection result on two sample untrimmed videos are shown in Figure 6.1. The performance of spatial only action recognition model with fixed size sliding windows predicts majority of actions to the correct action type and temporal location. The only defect is some minor mis-predictions on transit action.

Table 6.2. Frame-wise accuracy (F-acc) and window-wise accuracy (W-acc) for all three scenarios, action detection. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2.

Metric\Scene	R1C1	R2C1	R2C2
F-acc	85.75%	74.75%	82.76%
W-acc	86.28%	79.22%	85.41%

Table 6.3. Task-wise accuracy (T-acc) for all three scenarios, action detection. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2.

Metric\Scene	R1C1	R2C1	R2C2
Rub water	2.00s	1.70s	2.13s
Rub nowater	1.44s	1.20s	2.40s
Faucet hand	78%	80%	93%
Soap	100%	100%	93%

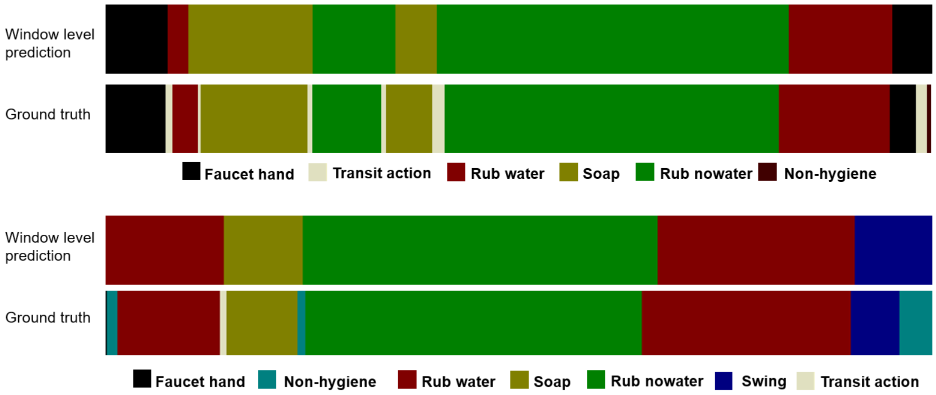


Figure 6.1. Visualization result from hand-hygiene action detection

6.1.6 Cross scenario hand-hygiene action recognition

After discussing hand-hygiene recognition in same scenario, we continue to explore hand-hygiene action recognition under cross scenarios. For researcher in the area of computer vision, deep learning is one the most popular strategies to solve various tasks. And majority of deep learning methods require large-scale dataset to build a robust model. In our research of hand-hygiene, the final model is expected to be deployed onto different laboratories or food factories scenarios. And it is unlikely we will be able to collect video data from all these facilities for training usage. Therefore, our goal is to explore the performance of constructing a CNN model on one scenario, and inference it on a different scenario. In this section, we explore the action recognition performance by applying the same models from Section 6.1.1 and inference them on cross scenarios for the four hand-hygiene action recognition. Based on the experiment results, we also analyze the main problem for RGB information to fail on cross scene hand-hygiene recognition task.

6.1.7 Four hand-hygiene action recognition

In this section, we apply the models introduced in Section 6.1.2 onto cross scenario recognition. For example, the model trained on scenario ”room1 camera1” will be tested on ”room2 camera1” and ”room2 camera2” test sets. All the training and testing setting are exactly the same as Section 6.1.2. Compare with the performance of testing on the same scenario, the testing on cross scenario experience a huge performance drop among all models, including both spatial only and spatio-temporal model.

6.1.8 Dataset bias in hand-hygiene action recognition

In this section, we focus on exploring the reason for cross scenario hand-hygiene to fail under RGB modality. As introduced by [114], a data collected for a particular task inevitably describes only part of the task; this is termed the dataset bias problem. Especially, the capture bias is related to how the data are captured under different camera view, illumination conditions, and background scenes. As our hand-hygiene data is collected under three differ-

Table 6.4. Model accuracy for all three scenarios cross recognition, four hand-hygiene actions. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2. **CNN**: ResNet50. **LSTM**: ResNet50 with LSTM. **TRN**: ResNet50 with TRN. **H**: involve only hand-hygiene actions.

Model\Scene	R1C1	R2C1	R2C2
R1C1 CNN (H)	91.67%	5.88%	30.19%
R2C1 CNN (H)	33.33%	97.06%	43.40%
R2C2 CNN (H)	56.67%	73.53%	98.11%
R1C1 LSTM (H)	91.76%	11.76%	43.40%
R2C1 LSTM (H)	28.33%	94.59%	47.17%
R2C2 LSTM (H)	35.00%	55.88%	93.41%
R1C1 TRN (H)	93.73%	38.24%	54.72%
R2C1 TRN (H)	58.33%	92.68%	45.28%
R2C2 TRN (H)	50.00%	61.76%	91.38%

ent scenarios, we believe dataset bias is a major cause of our poor performance in the cross scenario recognition.

To prove the existence of this issue, we use Grad-CAM [54] on the *conv_5x* layer of ResNet50 to generate a saliency map. These saliency maps help visualize the discriminative area that a CNN model uses to make a prediction. The top and bottom rows of Figure 6.2 show two different scenarios depicting the same ”rub hands with water” action. These images are tested by their corresponding 2D CNN ResNet50 with spatial RGB with correct prediction results. From the Grad-CAM, we observe the discriminative region for room1 camera1 not only covered “hands” and “waterflow”, but also include “sanitizer”, “water spout” and “faucet”, which are irrelevant objects to “rub water”. In contrast, room2 camera1’s discriminative region mostly covers the ”waterflow” and ”hands” region. Our hypothesis is that in certain room scenario, the discriminative region learned by CNN through weak supervision might involve irrelevant object. Thus, the irrelevant spatial object becomes a visual cue which contributes to ”hands related” action recognition. This bias on data capturing limits the model’s capability to do cross scene recognition.

To further demonstrate our hypothesis, we construct a hidden patch experiment. It has been addressed in several previous works [115][116][117], hide patch on a certain spatial area of in training could force a CNN model to extent its attention onto other discriminative areas.

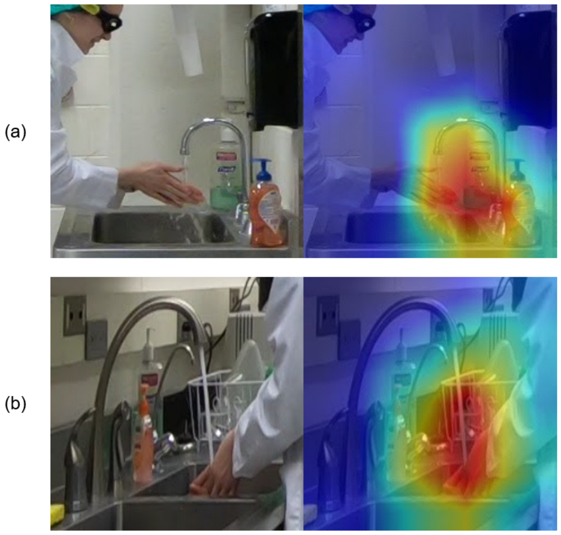
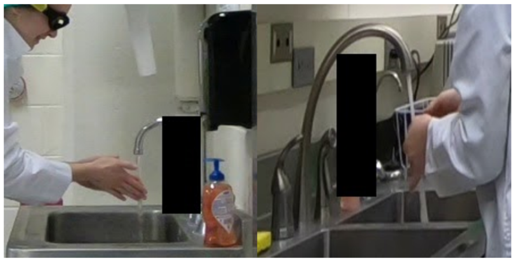


Figure 6.2. Saliency map for hand-hygiene; Row (a): rub hands with water in room1 camera1; Row (b): rub hands with water in room2 camera1.

Moreover, if hide patch applies onto images only for testing, the change in performance could reflect the importance of the hidden area to the target task. To reveal the “contribution” of irrelevant object to hand related action, we plan to cover an irrelevant object for hide patch experiment in testing stage. Our experiment makes comparison between scenarios ”room1 camera1” and ”room2 camera1” on the target action ”rub hands with water”. We plan to cover an irrelevant object ”alcohol sanitizer” in both scenarios’ test dataset of target action. All the ”alcohol sanitizer” in image are covered with black color patch, as shown in Figure 6.3. We test the 2D CNN ResNet50 models of each scenario on their own hide-images. Our expectation is that the detection accuracy on hide-image will decrease compare to original image, which indicates the irrelevant object indeed contributes to hand related actions.

We apply the same 2D CNN ResNet50 with spatial RGB as introduced in Section 6.1.2. The training and testing steps are also remains the same. In Table 6.5, we show the average prediction results on both scenarios’ test ”rub hands with water” videos. To distinct the change in recognition result before and after apply hide path, we directly apply the raw prediction score from CNN without softmax function. As we observe, the confidence score on ”rub hands with water” action for ”room1 camera1” model dropped 2.32 after apply hide patch to ”alcohol sanitizer”. In contrast, the confidence score on ”rub hands with



(a) (b)

Figure 6.3. Hide patch image (a): room1 camera1; Row (b): room2 camera1.

water" action for "room2 camera1" model dropped only 0.55. Thus, model train on "room1 camera1" significantly relies on the irrelevant object "alcohol sanitizer" to recognize "rub hands with water". And model trained on "room2 camera1" is barely affected that object.

Table 6.5. Prediction score (no softmax) on testset, rub hands with water videos; **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2; **origin**: test on regular image. **hide**: test on image with hide patch. **Act1**: touch faucet with hand. **Act2**: rub hands with water. **Act3**: rub hands without water. **Act4**: apply soap.

Model\Action	Act1	Act2	Act3	Act4
R1C1 origin	-1.93	**4.71**	0.02	-2.86
R1C1 hide	-1.24	**2.39**	-1.24	0.08
R2C1 origin	-1.98	**4.99**	-0.05	-3.10
R2C1 hide	-1.67	**4.44**	0.25	-3.24

This reveals that under weakly supervised learning, where each action is only given a class label, the model's learning stage is "uncontrollable". Depends on the scenario layout of data collection, there exist different kinds of irrelevant objects surround hand washing sink. Because RGB modelity is our only input resource, these irrelevant objects will always exist in our input data. The model could identify irrelevant object as discriminative visual cue to recognize an action. Even though in Table 6.5, which tests on the same scenario, both models are still capable to make correct prediction base on the highest confidence score, this "model doesn't pay attention on the correct discriminative part" issue could impact its capability to do cross-scenario action recognition.

6.2 Multi-modalities for cross scenario hand-hygiene recognition

In this section, we propose to use multi-modalities as input sources to solve hand-hygiene recognition problems in cross scenarios. The concept of using multi-modalities in action recognition has been explored in many previous works [35][118], where optical flow images and depth information are applied to combine with RGB information for action recognition.

First, we take the concept from transfer learning to explain our problem on cross scenario hand-hygiene recognition. Through this exploration, we confirm that the nature of solving this problem relies on focusing a common hand-hygiene actions set which shared by all scenarios and developing robust feature representation which build similarity refer to all scenarios.

Second, we propose to apply optical flow, segmentation masks, and skeleton joints as modalities as multi-modalities for cross scenario action recognition. We compare and explore each of these modalities on their capability of distinguishing certain type of hand-hygiene actions.

6.2.1 Idea from transfer learning

To solve the problem of cross scenario hand-hygiene recognition, we will need to analyze the necessary conditions. In this section, we introduce the idea from ”transfer learning” area to address our own task of hand-hygiene recognition. Reference to the work from Pan *et al.* [57], we can define our hand-hygiene recognition in terms of domain D and task T. A domain contains a feature space χ and a marginal probability distribution $P(X)$, where our video data collection $X = \{x_1, x_2, \cdots, x_n\} \in \chi$. A task contains a label space γ, which contains our ground truth label, and an objective predictive function f, which refers to our action recognition model. Since our goal to do cross scenario recognition, there exists at least 2 groups of ”domains” and ”tasks”, which name as ”source” and ”target”.

By the definition of ”transfer learning”, or ”domain adaptation”, the cross scene recognition can be interpreted as a transfer from ”source” to ”target”. Given a source domain D_s and a source task T_s, a target domain D_t and a target task T_t, the goal is to improve target task’s objective predictive function f_t with D_s and T_s. And the constraints are $D_s \neq D_t$ and

$T_s = T_t$. If we consider our cross scene hand-hygiene, it is always guaranteed that different camera angle, room layout, and illumination condition will always allow $D_s \neq D_t$. But for T_s and T_t, there will usually exist a different action set collection in non-hygiene actions. For example, data collected in roomA has ”talking” action, and data in roomB doesn’t. For our hand-hygiene, we summarized two strategies to deal with this action set issue. The first strategy is to assign all different non-hygiene actions into a uniform action class ”non-hygiene”. The second one is to classify an action as ”non-hygiene” if it gets rejected by all hand-hygiene classes. Due to the uncountable various of non-hygiene actions types, we select the second strategy to define both T_s and T_t in hand-hygiene.

Moreover, as a hand-hygiene action recognition task, we also need to find a reliable method which capable of using the knowledge of D_s and T_s to support the learning of f_t. There exist previous works on ”domain adaptation” for action recognition in cross views. Kong *et al.* [119] constructed view shared feature with auto-encoder to recognize cross-view actions. For the same task, Liu *et al.* [120] built view-invariant feature through sparse feature representation and distribution adaptation. From these cases, we summarize the key to transfer the existing knowledge in source domain to build objective function in target domain can be achieved by constructing robust feature representation.

In conclusion, we summarize that the two major components on solving cross scenario hand-hygiene recognition are focusing on common hand-hygiene action set and building robust feature representation.

6.2.2 Explore multi-modalities in hand-hygiene recognition

In action recognition, images or videos are the most commonly used input resource which represented by RGB color model. Beside RGB video/image, researchers also attempts to using other representations, which we refer as modalities, to help recognition. Simonyan *et al.* [35] applied stack of optical flow images in combining with RGB image to build a two-stream network model. Hu *et al.* [118] used RGB-D input, which contains depth map beside RGB video, to recognize human actions. Moreover, Yan *et al.* [37] took only skeleton joints information with Graph Convolutional Neural Network for action recognition. As we discussed in Section 6.1.8, the irrelevant objects in RGB images is the major obstacle to prevent cross scenario hand-hygiene recognition. Modalities such as optical flow and human skeleton joints are capable of maintaining motion or human only information, which effective removes the disturbance of irrelevant objects and robust against different scenarios. Furthermore, human action has grown into a large research area. Large scaled human action datasets such as UCF101[71] and NTU RGB-D[121] provide solid training data for deep learning models. With the support of these previous work, it is convenient for us to apply deep learning models pre-trained on large scale human action dataset and generate robust modality feature on our hand-hygiene data directly. For example, we can directly apply OpenPose[122] method to generate human skeleton joints information directly on our hand-hygiene data without the need for additional training.

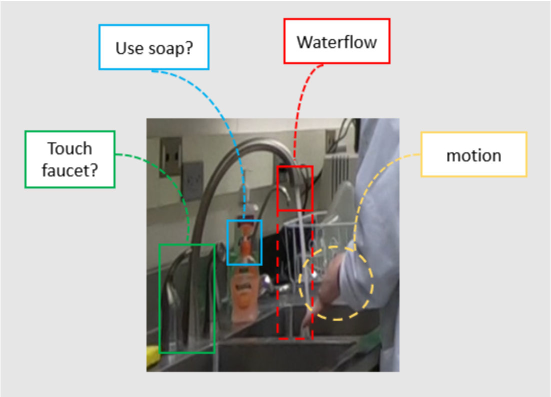


Figure 6.4. Human object interaction, **Red rectangle**: water spout. **Blue rectangle**: soap head. **Green rectangle**: faucet. **Yellow circle**: hand motion.

Moreover, inspired by the research topic Human-Object Interaction (HOI)[50][51], which could be considered as a sub-task to action recognition, the spatial region where participant interacts object could have significant contributions toward cross scenario hand-hygiene recognition. The task of HOI is to localize both human and object locations and predict an interaction class category for them. Reference to Chao *et al*, the baseline framework of HOI consists object, human, and interaction streams. Especially for object and human streams, a "hard attention" is applied by cropping region of interest (ROI) at object and human location to force CNN models to extract only information on these regions. As shown in Figure 6.4, the hand-hygiene human object interaction ties with ”faucet”, ”soap head”, and ”water spout” regions. Basically, hands appear in one of these regions could trigger a corresponding hand-hygiene action. The motion of hands could reveal action cues by itself as well. To acquire the locations of ”faucet”, ”soap head”, and ”water spout”, one could construct an object detector, such as YOLO [81]. In this work, our prime goal is to explore cross scenario hand-hygiene recognition. For exploration purpose, we apply human labeling bounding box for the rest of this paper.

For the rest of this section, we introduce to use multi-modalities we selected to recognize cross scenario hand-hygiene actions. The modalities we select are RGB modality, optical flow, segmentation masks, and human skeleton joints. We explore what is each modality’s capability on recognizing certain hand-hygiene action types, and which region of interest from the input video should each modality applies to. To better describe each modality’s capability, we further categories our four hand-hygiene actions into two parent category, hand to hand action and hand to object action. The hand to hand action includes ”rub hands with water” and ”rub hands without water” actions. The hand to object action includes ”touch faucet with hand” and ”apply soap”. Through our exploration, we conclude to use skeleton joints modality for non-hygiene action rejection, optical flow modality to categorize hand to hand and hand to object action, hand mask modality to distinguish ”touch faucet with hand” and ”soap” actions in hand to object category, and RGB modality with adversarial learning to recognize ”rub hands with water” and ”rub hands without water” actions in hand to hand category.

6.2.3 Optical flow

Optical flow is one of the widely used modalities in action recognition [33][32][35], which is capable at describing motions on image pixel level. For our hand-hygiene actions recorded with a static third person camera view, optical flow can track the moving body parts of the participant, which majorly focus on the forearm, as well as rejecting static objects information such as the appearance of soap and sanitizer. Thus, we propose to use optical flow on recognizing cross scenario ”hand to hand” action and ”hand to object” action as we defined in the beginning of Section 6.2.2.

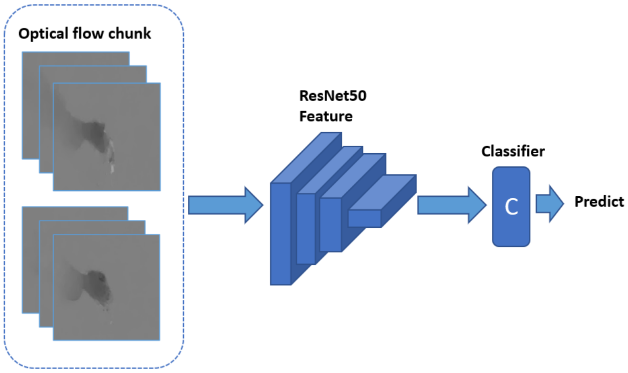


Figure 6.5. Optical flow modality model structure

All three different scenarios optical flow are pre-computed with TV-L1[90], rescaled to $[0, 255]$, and saved as images. Refer to the temporal network structure mentioned in [35][111], we select ResNet50 with pre-trained weight on ImageNet[93] as our model to take 10 stack of optical flow image pairs as input as shown in Figure 6.5. The first convolutional layer is edited to take 20 channels input. The pretrained weight on that conv layer is averaged by its original 3 channels and repeated 20 times to match the edition.

Training: The ResNet50 model is trained with 350 epochs with batch size 32 and an initial learning rate 0.001, which decreased by 10 at 200 and 300 epochs. We select to use Stochastic Gradient Decent (SGD) optimizer. For the loss function, we compare the binary cross-entropy loss with cross-entropy loss, and decide to use cross-entropy loss for better performance. To avoid over-fitting, data augmentation multi-scale crop and random horizontal flip duration training stage as introduced in [111].

Testing: Each model is tested on all 3 scenarios’ trimmed video clip datasets. The model is applied on every sliding window on test video clip. Each window has fixed size of 10 frame and step size of 1 frame. The prediction result for a video clip is the average score among all the softmax-ed score of each window.

The testing results of using optical flow are shown in Table 6.6. We observe the result model trained on one scenario and inference to any other scenario achieve good result of over

Table 6.6. Model accuracy for all three scenarios cross recognition, two actions. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2. **flow**: ResNet50 with optical flow input.

Model\Scene	R1C1	R2C1	R2C2
R1C1 flow	98.33%	94.12%	98.11%
R2C1 flow	95.00%	100.00%	96.23%
R2C2 flow	96.67%	94.12%	100.00%

90% accuracy. To conclude, optical flow based model is capable at cross scenario hand to hand and hand to object recognition without pre-knowledge of target data.

6.2.4 Mask modality

In this section, we continue to explore another modality in cross scenario hand-hygiene recognition, the mask modality. The mask modality is based on the result from image segmentation task, which segments an image by labeling each pixel with a class category [123]. There exist previous work [33] apply segmentation masks as clue of hands to recognize egocentric activities. For hand-hygiene actions, the majority of actions are performed by human body parts and its interaction with fixed location objects. With image segmentation masks, we could discard redundant appearance information and maintains the shape and silhouette which is robust cross scenarios. Therefore, we apply both the mask information of human and objects to explore on cross scenario hand-hygiene recognition.

As human mask, we compare two different mask types, hand mask and person part mask. The hand mask is generate by the method in [124], which applies RefineNet[125] pretrained on EYTH (EgoYouTubeHands) dataset for hand segmentation. The result hand mask image is a binary image. For person part mask, we use the method of [126], which is a RefineNet pretrained on PASCAL Person-part[127] dataset to segment 6 human body parts. The result hand mask image is a RGB image.

To explore the capability of mask modality on cross scenario recognition, we compare two different tasks.Task 1 : Action recognition of ”hand to hand”, ”touch faucet with hand”, ”apply soap”. Task 2: Action recognition of ”touch faucet with hand” and ”apply soap”.

6.2.5 Task 1: recognition of 3 actions

In this task, we explore the capability of both hand mask and person part modalities at recognizing ”hand to hand”, ”touch faucet with hand’, and ”apply soap” actions. As shown in Figure 6.6, in order to include both object and human masks information in the input source, we put ”faucet and water spout” bounding box mask, hand or person part mask, and ”soap head” bounding box mask into different images and combine them as channels to create a mask image. And to include temporal information as comparison to optical flow modality, we stack 10 of the mask images as input to CNN network. This results into 30 and 50 channels input source for hand and person part modalities. For model selection, we choose to apply ResNet50 model with pretrained weight. The first convolutional layer is edited to match the corresponding modality’s channel size. The usage of pretrained weight is the same as Section 6.2.3.

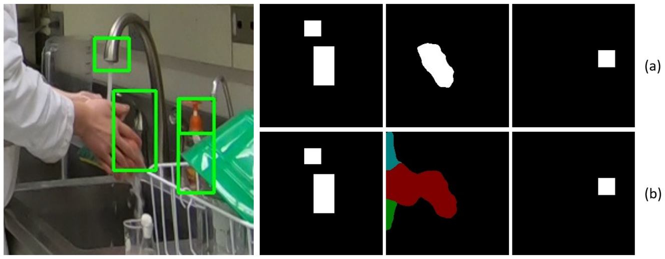


Figure 6.6. Object and human mask, images from left to right (a) faucet and water spout mask, hand mask, soap mask (b) faucet and water spout mask, person part mask, soap mask.

For training, the ResNet50 model is trained with 250 epochs with an initial learning rate 0.0001, which decreased by 10 at 100 and 200 epochs. The rest training and testing setting is the same as Section 6.2.3. The test result of hand and person part modalities among all three scenarios are summarized in Table 6.7. We observe that both modalities demonstrate their capabilities to recognize all three actions under same scenario for over 90% accuracy. However, when applying cross scene recognition, there exists performance drop.

A more detailed version of modality performance on cross scenario recognition among three actions is shown in Table 6.8. In this Table, we observe the person part mask has a better performance than hand mask at hand to hand action. However, for the performance of ”touch faucet with hand” and ”apply soap” actions, hand mask is more effective. Since we have demonstrated optical flow modality in Section 6.2.3 with high performance at recognizing hand to hand action, our expectation is to use other modalities to better recognize ”touch faucet with hand” and ”apply soap” action. So far, even though hand mask has better performance than person part mask at recognizing these two actions, the performance could still be improved.

Table 6.7. Model accuracy for all three scenarios cross recognition, three actions. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2. **hand**: ResNet50 with hand mask modality. **person**: ResNet50 with person mask modality.

Model\Scene	R1C1	R2C1	R2C2
R1C1 hand	93.33%	88.24%	84.91%
R1C1 person	100%	85.29%	94.34%
R2C1 hand	73.33%	100.00%	64.15%
R2C1 person	75.00%	100.00%	67.92%
R2C2 hand	63.33%	97.06%	96.23%
R2C2 person	63.33%	79.41%	98.11%

Table 6.8. Modality average accuracy for cross scenarios only, three actions. **Hand**: average performance of hand mask modality. **Person**: average performance of person mask modality. **F(C)**: touch faucet with hand action, tested on other scenarios beside the training scenario. **HH(C)**: hand to hand action, tested on other scenarios beside the training scenario. **S(C)**: apply soap action, tested on other scenarios beside the training scenario.

Modality\Action	F(C)	HH(C)	S(C)
Hand	50.00%	90.56%	72.91%
Person	42.42%	97.78%	43.75%

6.2.6 Task 2: recognition of 2 actions

In this task, we remove ”hand to hand” from our action set and only focusing on recognizing hand to object actions, which are ”touch faucet with hand” and ”apply soap”. It has been proved in Table 6.8 that hand mask is better at recognizing hand to object actions than person part. Thus, we select hand mask modality for this task. Moreover, based on our discussion about HOI in Section 6.2.2, the ”touch faucet with hand” and ”apply soap” actions occur in the ROI of ”faucet” and ”soap head”. Instead of using the whole image, we crop the ROI of ”faucet” and ”soap head” on the hand mask as input source to our model.

For hand mask ROI images on ”faucet” and ”soap head”, we compare different settings and decide to resize these images to fixed size 224 x 224 and stack with 10 frames to include temporal information. As shown in Figure 6.7, each stack of hand mask ROI image input into a VGG11[20] backbone with unshared weights and concatenate their features before the last fully-connected layer for final prediction. The model is trained with 350 epochs with an initial learning rate 0.0001, which decreased by 10 at 200 and 300 epochs. The rest training and testing setting is the same as Section 6.2.3.

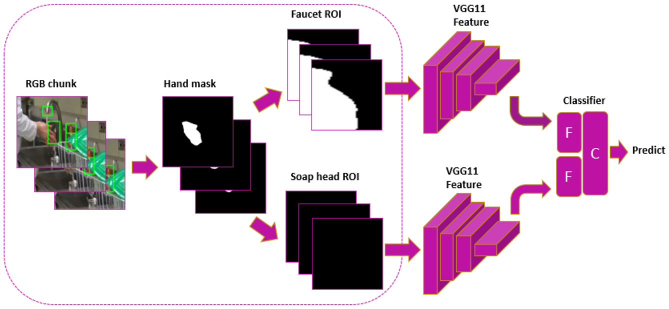


Figure 6.7. Hand mask modality model structure

The performance of cropped ROI hand mask for each action on cross scenario recognition is shown in Table 6.9. The average detection accuracy of each action is computed from cross scenario recognition only, where a model is tested on all other scenarios except its training scenario. The hand mask with ROI targeting on objects is giving good performance on both actions, especially with model trained on room1 camera1. Due to the lack of depth information and camera angle constraint, it is difficult to distinguish when hands and object are spatially overlapped but not touched. Thus, there exist a performance drop especially with model trained on room2 camera1. To conclude, the usage of hand mask modality with cropped ROI is capable at recognizing ”touch faucet with hand” and ”apply soap” actions in cross scenario, even without any pre-knowledge of target data.

Table 6.9. Two action average accuracy for cross scenarios only **Faucet hand**: average performance of ”touch faucet with hand action”. **Soap**: average performance of ”apply soap” action. **R1C1(C)**: Model trained on room1 camera1, test on other scenarios beside room1 camera1. **R2C1(C)**: Model trained on room2 camera1, test on other scenarios beside room2 camera1. **R2C2(C)**: Model trained on room2 camera2, test on other scenarios beside room2 camera2.

Action\scene	R1C1(C)	R1C2(C)	R2C2(C)
Faucet hand	100.00%	100.00%	68.00%
Soap	90.91%	28.57%	75.00%

6.2.7 Coordinate modality

In this section, we explore the usage of coordinate modality in cross scenario hand-hygiene recognition. As introduced in previous sections, modalities such as optical flow and mask segmentation are capable of keeping shape and silhouette information. In contrast, coordinate base modality further discards silhouette and maintains semantic location information. It has been demonstrated in previous works[128][32] that coordinates information can be applied onto action recognition, either use human skeleton joints as feature or use object coordinate to localize region of interest. For our cross scenario hand-hygiene recognition, we experiment the performance of coordinate information to recognize three actions: ”hand to hand” action, ”touch faucet with hand” action, and ”apply soap” action.

6.2.8 Coordinate generation

Our coordinate modality includes two parts: object coordinates and human skeleton joints. For object coordinates, we focus on the location of three objects, namely ”faucet”, ”water spout”, and ”soap head”. The coordinates are computed as center coordinates of each object’s bounding box, which is detected from a object detector. In our exploration, we apply human labeling of object location to replace the object detector. For skeleton joints, we are interested in the upper body skeleton joints of a participants, namely ”shoulder”, ”elbow”, ”wrist”, and ”hands”. We reference to the methods [122][129][130] for the upper body skeleton detection. The skeleton joints detection was applied on the original 1080 image which whole human can be clearly viewed. Using OpenPose method, 18 human body joints and 21 hand joints for all the people in the image are detected, and we keep only joints from person around the sink area defined in Section 3.6.2. Among all 18 body joints, we keep ”shoulder”, ”elbow”, and ”wrist” of both left and right side. The 21 hand joints location is averaged to generate the coordinates of hand.

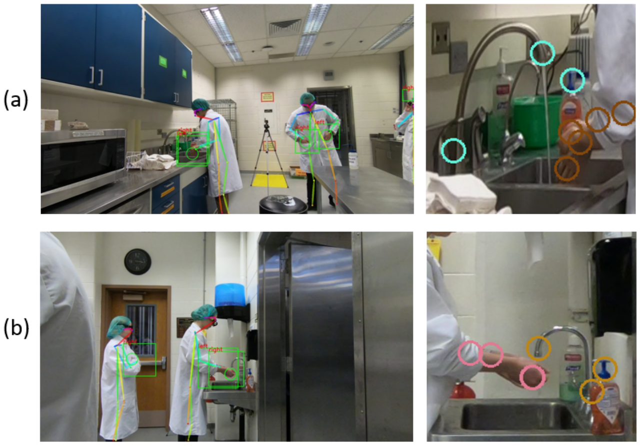


Figure 6.8. Coordinate detection results, left: OpenPose, right: object + upperbody coordinates; Row (a) room2 camera1. Row (b) room1 camera1.

6.2.9 Method and Evaluation

We apply the CNN based method of Li *et al.*[39] which stacks coordinates by time as a input matrix to recognize action. For cross scenario hand-hygiene action, we apply the 4 human upper body joints of left and right side plus 3 object coordinates, which builds 11 coordinates from each video frame. The time stack length is chosen as 16 frames to match the down samplings in the CNN. For training, we apply the same setting as the work[39] of NTU RGB+D dataset. And the model is applied on every sliding window of each test video clip. Each window has fixed size of 16 frame and step size of 1 frame. The prediction result for a video clip is the average score among all the softmax-ed score of each window. The detection result are shown in Table 6.10. The performance of coordinate modality is reasonable when train and test on the same scenario. However, when doing cross scenario recognition, its performance is worse than mask modality shown in Table 6.7. The performance drop of coordinate information is majorly caused by the inconsistency at skeleton joint detection. From Table 6.11, the scenario room1 camera1 is able to detect almost all the upperbody joints. However, the scenario room2 camera1 and room2 camera2 always have issue on detecting one side of wrist and hand. This missing wrist and hand issue constraints the coordinate modality to build robust feature against different scenarios.

Table 6.10. Model accuracy for all three scenarios cross recognition, three actions. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2. **coord**: coordinates modality.

Model\Scene	R1C1	R2C1	R2C2
R1C1 coord	98.33%	70.59%	92.45%
R2C1 coord	60.00%	94.12%	79.25%
R2C2 coord	71.67%	64.71%	96.23%

Table 6.11. Upperbody skeleton detection rate for all three scenarios. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2. **L**: left side. **R**: right side. **S**: shoulder. **E**: elbow. **W**: wrist. **H**: hand.

(a) Left side detect rate

Model\Scene	L+S	L+E	L+W	L+H
R1C1	98.44%	97.01%	91.21%	91.21%
R2C1	98.83%	98.57%	97.91%	97.91%
R2C2	98.73%	83.89%	52.73%	52.73%

(b) Right side detect rate

Model\Scene	R+S	R+E	R+W	R+H
R1C1	99.41%	99.32%	99.04%	99.04%
R2C1	95.83%	94.04%	56.37%	56.37%
R2C2	99.29%	99.17%	99.06%	99.06%

Despite coordinate modality does not provide a competitive performance over other modalities on recognizing ”hand to hand”, ”touch faucet with hand” and ”apply soap” actions. It is still an useful modality which can be applied to reject non-hygiene actions. As introduced in Section 6.2.1, we plan to not create an action class category ”non-hygiene” and reject it with other methods. Human skeleton joints are capable of providing semantic information of human body points. As we observe from Table 6.11, among all three scenarios, the detection rates of hand and wrist are over 90%, at least in one of the left or right side. Therefore, many of non-hygiene actions can be rejected by checking if hand and wrist joints present in the sink ROI. To conclude, the coordinate modality is capable for rejecting non-hygiene actions with unsupervised based method among all scenarios.

6.2.10 RGB modality

In this section, we continue to explore the usage of RGB modality in cross scenario hand-hygiene recognition. As mentioned in Section 6.1.8, the RGB modality has its own disadvantage on cross scenario recognition by involving redundant object appearance. However, RGB modality also contains special appearance information which is not covered by any of the rest modalities. For example, the appearance ”waterflow” is crucial for actions ”rub hands with water” and ”rub hands without water”. Thus, we experiment the capability of RGB modality on recognizing the two actions in hand to hand action category, which are ”rub hands with water” and ”rub hands without water”. First, we compare the different selection of region of interest for RGB modality in cross scene recognition and evaluate the performance. Second, we further improve the result of RGB modality on recognizing ”rub hands with water” and ”rub hands without water” actions with offline data augmentation and domain adaptation methods.

6.2.11 Region of interest selection for RGB modality

Based on the idea of ”hard attention” to crop relevant objects and human in Human-Object Interaction (HOI), and our demonstration that RGB modality might include redundant object appearance, we select the relative regions of interest (ROI) to recognize ”rub hands with water” and ”rub hands without water” actions. Our ROI fall into two candidates: hand ROI and waterflow ROI. As shown in Figure, both of these candidates are cropped with a fixed size patch and resized into 224 x 224 image. The hand ROI is achieved through skeleton recognition in Section 6.2.7 of the leading hand joint, the closest hand to water spout, and fix crop of 80 x 80 image patch. The missing hand joint is replace by a black image. The waterflow ROI is achieved by the human labeling bounding box of water spout and extend it to the bottom of the image. Our comparison sets are selected as hands and waterflow ROI or waterflow ROI only.

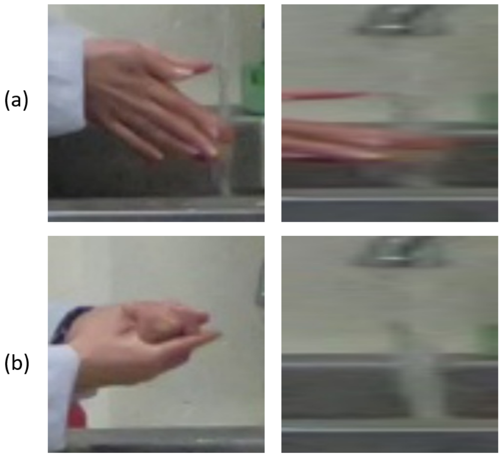


Figure 6.9. ROI candidates; Row left: hand ROI. Row right: waterflow ROI.

We select ResNet50 model for both comparison sets, and input the stack 10 frames for temporal information. For the set of two ROI images, we expand two branch of ResNet backbones with unshared weights and concatenate their features before the last fully-connected layer for final prediction. Model in both sets are trained with 350 epochs with an initial learning rate 0.0001, and decreased by 10 times at 200 and 300 epochs. The rest training and testing setting is the same as Section 6.2.3.

Table 6.12. Model accuracy for all three scenarios cross recognition, two actions. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2. **H**: hand ROI. **W**: waterflow ROI.

Model\Scene	R1C1	R2C1	R2C2
R1C1 H + W	96.15%	62.96%	62.16%
R1C1 W	100.00%	88.89%	62.16%
R2C1 H + W	73.07%	100.00%	62.16%
R2C1 W	73.07%	100.00%	62.16%
R2C2 H + W	92.31%	66.67%	100.00%
R2C2 W	88.46%	100.00%	100.00%

The result is indicated in Table 6.12. As we observe, the performance of using waterflow ROI only is better than using both hand and waterflow ROI, especially on scenarios room1 camera1 and room2 camera2. The main reason is because that the participant's hand might show up in any place of the scenario, which brings uncertainty to the information included in the background of hand ROI as well as increase the miss detection rate of hands. Therefore, we will select only to use the waterflow ROI for RGB modality.

6.2.12 Data augmentation and adversarial learning

Data augmentation With waterflow ROI applied, there exists three situations for "rub water with hands" and "rub water without hands" actions: (1). Hands cut waterflow. (2). Hands away from waterflow. (3). Hands overlap waterflow, where (1) belongs to "rub water with hands" action and (2) (3) belong to "rub water without hands" action. In our data collection, however, room1 camera1 and room2 camera2 doesn't include any of the situation (3), which is an crucial sample type to distinguish the two actions. As shown in Figure 6.10, the overlapped hand and waterflow forces CNN model to recognize the bottom area of water as discriminative region, and increases feature robustness. Therefore, we observe in Table 6.12 that the model trained on room2 camera2 with waterflow has a highest prediction accuracy at recognize cross scenario actions over models trained at the other two scenarios.

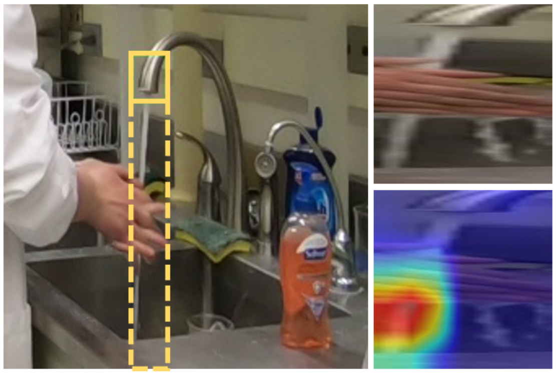


Figure 6.10. Hands overlap waterflow; (a) Left: water spout ROI extends to acquire waterflow (b) Right top: resized waterflow ROI. (c) Right bottom: waterflow ROI attention.

To compensate this situation for room1 camera1 and room2 camera1, we use offline data augmentation to create samples. The sample video is created by manually selecting few consecutive frames in those two scenarios video where hands are overlapped with waterflow. And these frames are repeated to reach 60 frames to create a video clip for situation ”hands overlap with waterflow”. Consider the balance of two action classes during training, we created 8 and 3 ”hands overlap with waterflow” video samples and added into room1 camera1 and room2 camera1 training sets. As indicated in Table 6.13, the offline data augmentation improved the recognition accuracy for room1 camera1 and room2 camera1 models on room2 camera2 data.

Domain adaptation Moreover, we apply adversarial learning based domain adaptation method [61] to improve CNN model robustness across scenarios. As we explained in Section 6.2.1, each of our three scenarios could be considered as a individual domain, and the nature of cross scenario recognition is to build a model based on source domain data and label, and apply to target domain. Despite we narrow the discriminative regions on waterflow ROI to distinguish ”rub water with hands” and ”rub water without hands” actions, the dataset bias problem still exists among different domains. To overcome this problem, we use a discriminator in training stage to help mapping target domain model's feature onto source domain. A source domain CNN model is trained separately in the beginning with its labeled data. For target domain CNN model, the discriminator and target domain CNN are trained alternatively follow two steps. As indicated in Figure 6.11, step one starts by training the discriminator with mixed input of CNN feature extracted from data in both domains with their corresponding CNN models. The goal of discriminator is to recognize the domain of each input feature vector. Step two, the target domain CNN extracts feature from target domain data and input into discriminator. To constraint the target domain CNN's feature to match source domain, this predicted domain labels is expect to match source domain label (inverted label).

Moreover, we target at actions of ”rub hands with water” and ”rub hands without water” from all four hand-hygiene action types, it is expected that we could adapt source domain to target domain only for the action set of these two actions only. However, our assumption is that the target domain training data doesn't have ground truth label provided, and we

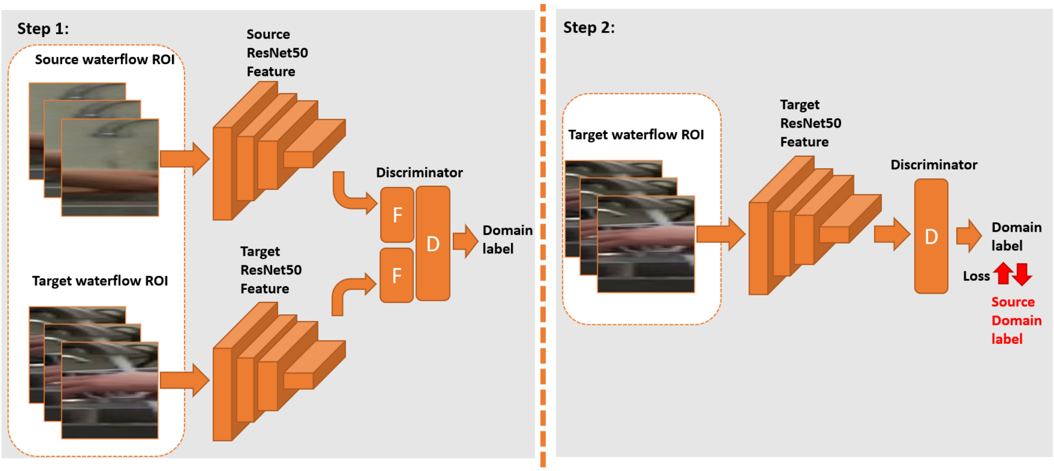


Figure 6.11. Target domain model training procedures.

couldn't directly select the hand to hand action set out of all action clips. To solve this problem, we apply the optical flow model trained in Section 6.2.3 and make prediction on the training data of target domain to assign "hand to hand" and "hand to object" class label, and use data with "hand to hand" label as training data in target domain. With this additional procedure, the unnecessary hand to object actions in target domain get filtered out and it reduces the level of difficulty to do domain adaptation.

To accomplish the domain adaptation, we choose the same ResNet50 with input of stack 10 frames as CNN model for both source and target domain. The discriminator is a multilayer perceptron (MLP) with 3 layers. Both the target domain CNN and discriminator are trained with cross entropy loss in combine with Adam optimizer with 0.00001 and 0.001 learning rate respectively for 350 epoches. Since we have three scenarios as individual domain, each source domain is tied with two target domain models. In testing, we will need to have the pre-knowledge of which target domain the test data belongs to. The rest testing setting is the same as Section 6.2.3.

As results in Table 6.13, the adversarial learning further improve the performance over most of the cross scenarios. especially for model trained on room2 camera1. Moreover, there also exist some cross scenario results which are not good as expected, such as room1 camera1 model inference on room2 camera2. The reasons behind this issue is due to both the data collection and algorithm. The data we collected in room1 camera1 scenario is naturally short

Table 6.13. Model accuracy for all three scenarios cross recognition, two actions. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2. **W**: waterflow ROI. **D**: offline data augmentation. **A**: adversarial. **I**: ideal.

Model\Scene	R1C1	R2C1	R2C2
R1C1 W	100.00%	88.89%	62.16%
R1C1 W + D	100.00%	96.30%	70.27%
R1C1 W + D + A	100.00%	92.59%	56.76%
R1C1 W + D + A(I)	100.00%	88.89%	81.08%
R2C1 W	73.07%	100.00%	62.16%
R2C1 W + D	84.62%	100.00%	78.38%
R2C1 W + D + A	100.00%	100.00%	83.78%
R2C1 W + D + A(I)	100.00%	100.00%	94.59%
R2C2 W	88.46%	100.00%	100.00%
R2C2 W + D + A	88.46%	100.00%	100.00%
R2C2 W + D + A(I)	100.00%	100.00%	100.00%

of some types of action variation. Even though we attempt to compensate this issue with data augmentation with manually created data, it is still not as satisfied as real data. Also, the mistake in optical flow target domain data labeling might also influence the domain adaption performance.

To further demonstrate the efficiency of our method, we provide the results from ideal situation as well. The ideal situation assumes we have a perfect method, instead of our current optical flow model, to label and select all the hand to hand data in target domain for training. As we observe, the results under ideal situation is further improved, especially for model train on room2 camera2, which achieved 100% accuracy on all cross scenario recognition.

Therefore, we conclude that RGB modality with adversarial learning is capable at recognizing ”rub hands with water” and ”rub hands without water” actions.

6.3 Cross scenario hand-hygiene recognition system

In previous Section 6.2, we explored the capability of multiple modalities on recognizing hand-hygiene actions under cross scenarios and proved that each modality is expertise at recognizing certain set of actions. In this section, we combine all these different modalities for hand-hygiene action detection task in cross scenarios. In Section 6.3.1, we introduce our idea to design the system refer to K-class pattern recognition problem with K neural networks. Our system is designed as a collaboration between multiple modalities' CNN models to hierarchically detect hand-hygiene actions. In Section 6.3.2, we show our final performance of multi-modality system on cross scenario action detection.

6.3.1 System design

As we explored in previous sections, each of the modalities we chose has its own advantage on recognizing certain action types and the model architecture design for each modality is also varied. It becomes a question of how to collaborate all these modalities for a action detection task? We refer to the work of Ou *et al.*[131] which leads to the K-class pattern classification problem. Basically, our hand-hygiene recognition problem can be considered as a branch of K-class pattern classification problem. As mentioned in [131], one solution to solve this problem is to apply K neural networks using OAA (one-against-all). This solution has advantages to allow each network to have its own non-interfered feature space and model architecture. The solution matches our desire for each modality to solve a partial task with its own model design without interfere with other features. As we mentioned before, each modality is only capable at recognizing certain action types. The OAA strategy which requires a model to learn all other actions as one action category becomes unrealistic for our application. Therefore, we use a hierarchical design to collaborate all the modalities.

As shown in Figure 6.12, we apply densely processing strategy to iterate through the target untrimmed video. Because all of our modality models take a video chunk of 10 frames as input, we iterate every frame i by selecting frame i to frame i + 9 as a video chunk and predict on the video chunk as the result for frame i. Each chunk of video clip is first input into the skeleton joint modality model, which applies unsupervised method to reject non-hygiene actions which upperbody joints does not present in the sink ROI. Then, the remaining hand-hygiene action clip is input into optical flow modality to distinguish hand to hand and hand to object actions. Clips of these two categories is further delivered into

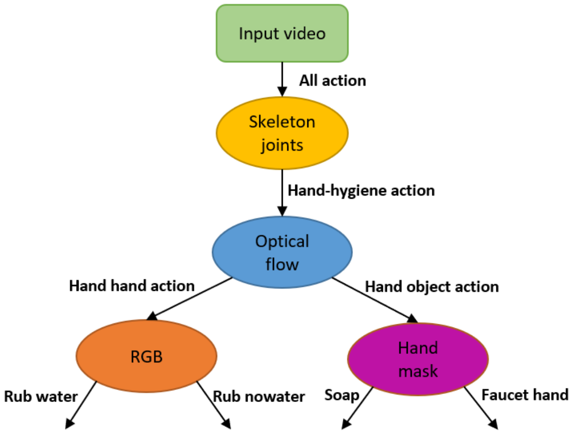


Figure 6.12. Hierarchical action recognition with multi-modalities.

RGB and hand mask modality models respectively, and predict final action label as one of the four types hand-hygiene actions. The rest of test setting details and evaluation is the same as Section 6.1.6.

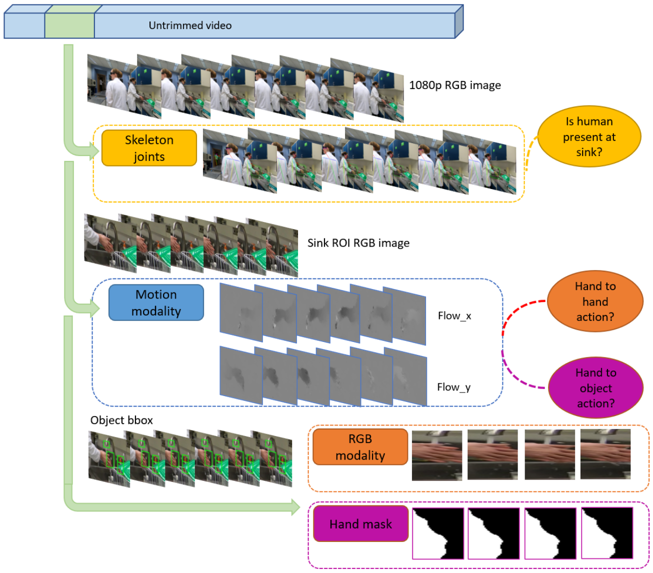


Figure 6.13. Cross scenario action detection.

6.3.2 Metric and Evaluation

For action detection in cross-scenario hand-hygiene, we follow the same densely processing strategy to acquire video chunk from untrimmed video and the video chunk is input into our hierarchical multi-modalities models to recognize its class category. The detailed test setting is the same as Section 6.1.6. For comparison, we select the action detection method in Section 6.1.6 as the baseline method. Moreover, to demonstrate the full potential of our method, we add the ideal situation introduced in Section 6.2.12 to build the system and show its result for comparison as well. The final result in shown in Table 6.14 with metrics frame-wise accuracy (F-acc) and window-wise accuracy (W-acc). On average, our multi-modality system outperforms the baseline system by 36.42% and 38.46% on cross scenario with respect to frame-wise accuracy and window-wise accuracy. This proves that our system design is effective against cross scenario hand-hygiene recognition. Moreover, the ideal situation further improves the performance of our multi-modality system by 3.82% and 4.19% on cross scenario by average with respect to frame-wise accuracy and window-wise accuracy, especially for the cross scenario between room1 camera1 and room2 camera2. Figure 6.14 and 6.15 show two different untrimmed video cross scenario action detection from current multi-modality and ideal systems. As we observe, ideal system is better at distinguish ”rub hands with water” and ”rub hands without water actions” due to the ideal target domain hand to hand labeling.

Moreover, we also show the task-wise evaluation, as introduced in Section 6.1.5, in Table 6.15 and 6.16 with respect to ideal labeling and optical flow labeling in target domain. In Table 6.15(b) and 6.16(b), we observe that the multi-modalities trained on room2 camera 1 scenario achieves a reasonable performance in cross scenario action detection. Especially in ”rub hands with water” and ”rub hands without water” actions, many cross scenario detection achieve mis-match seconds around 2 or 3 seconds, which is close to some results in same scenario detection. In contrast, the multi-modalities system trained with room1 camera1 data, as Table 6.15(a) and 6.16(a), is experiencing a performance worse than other models. This indicates that some certain scenarios has its natural disadvantage for its layout and data collection, which could result into a less good performance at cross scenario detection.

Table 6.14. Frame-wise accuracy (F-acc) and window-wise accuracy (W-acc) for all three scenarios, action detection. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2.

(a) Baseline system

Metric\Model	R1C1	R2C1	R2C2
R1C1 F-acc	85.75%	10.93%	12.29%
R1C1 W-acc	86.28%	10.33%	12.25%
R2C1 F-acc	9.69%	74.75%	34.14%
R2C1 W-acc	9.54%	79.22%	33.38%
R2C2 F-acc	37.99%	52.70%	82.76%
R2C2 W-acc	38.94%	55.50%	85.41%

(b) Multi-modality system

Metric\Model	R1C1	R2C1	R2C2
R1C1 F-acc	74.92%	67.62%	54.56%
R1C1 W-acc	76.69%	67.74%	56.68%
R2C1 F-acc	58.11%	75.14%	68.68%
R2C1 W-acc	63.07%	76.14%	71.45%
R2C2 F-acc	53.18%	74.08%	82.31%
R2C2 W-acc	57.20%	74.56%	86.08%

(c) Ideal multi-modality system

Metric\Model	R1C1	R2C1	R2C2
R1C1 F-acc	74.92%	67.08%	66.00%
R1C1 W-acc	76.69%	66.90%	69.91%
R2C1 F-acc	58.47%	75.14%	69.42%
R2C1 W-acc	63.81%	76.14%	72.81%
R2C2 F-acc	62.84%	75.32%	82.31%
R2C2 W-acc	66.71%	75.72%	86.08%

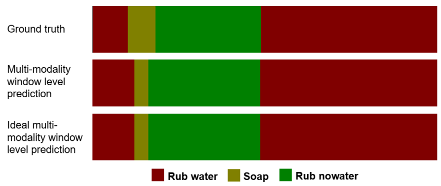


Figure 6.14. Visualization result from cross scenario hand-hygiene action detection.

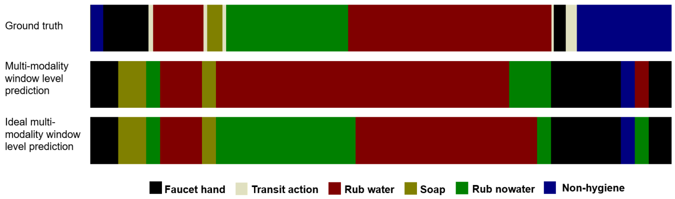


Figure 6.15. Visualization result from cross scenario hand-hygiene action detection.

Table 6.15. Task-wise accuracy (T-acc) for all three scenarios, action detection, ideal labeling. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2. **RW**: rub hands with water. **RNW**: rub hands without water. **Faucet**: touch faucet with hand. **Soap**: apply soap.

(a) Ideal labeling: R1C1 model

Scene\Task	RW	RNW	Faucet	Soap
R1C1	0.89s	4.11s	67%	89%
R2C1	4.00s	6.50s	70%	80%
R2C2	4.60s	6.80s	80%	93%

(b) Ideal labeling: R2C1 model

Scene\Task	RW	RNW	Faucet	Soap
R1C1	2.00s	2.89s	67%	100%
R2C1	2.30s	2.40s	60%	90%
R2C2	3.60s	3.67s	27%	93%

(c) Ideal labeling: R2C2 model

Scene\Task	RW	RNW	Faucet	Soap
R1C1	2.00s	3.89s	89%	89%
R2C1	1.90s	3.20s	60%	80%
R2C2	1.60s	2.20s	33%	100%

6.4 Conclusion

In this chapter, we introduce hand-hygiene video systems to evaluate hand-hygiene actions with different application requirements. To support our exploration in hand-hygiene actions, we collect a dataset from 23 students in a food class with static third person camera view. Our first application targets at hand-hygiene recognition in same scenario. We compare the performance of utilizing spatial information only and spatio-temporal information for the tasks of action recognition and action detection. The results indicate that using

Table 6.16. Task-wise accuracy (T-acc) for all three scenarios, action detection, ideal labeling. **R1C1**: room1 camera1. **R2C1**: room2 camera1. **R2C2**: room2 camera2. **RW**: rub hands with water. **RNW**: rub hands without water. **Faucet**: touch faucet with hand. **Soap**: apply soap.

(a) Flow labeling: R1C1 model

Scene\Task	RW	RNW	Faucet	Soap
R1C1	0.89s	4.11s	67%	89%
R2C1	3.90s	6.30s	70%	80%
R2C2	6.87s	8.93s	80%	87%

(b) Flow labeling: R2C1 model

Scene\Task	RW	RNW	Faucet	Soap
R1C1	2.67s	3.44s	67%	100%
R2C1	2.30s	2.40s	60%	90%
R2C2	4.00s	3.80s	27%	93%

(c) Flow labeling: R2C2 model

Scene\Task	RW	RNW	Faucet	Soap
R1C1	6.33s	6.11s	89%	100%
R2C1	2.10s	3.60s	60%	70%
R2C2	1.60s	2.20s	33%	100%

spatial information only is effective enough to process hand-hygiene actions within the same scenario.

Moreover, we also explore the dataset bias problem in hand-hygiene actions. Because of the variance in camera angle, illumination, and background scene, hand-hygiene data collected in different facilities doesn't share same appearance. A model trained on one data collection might failure at recognizing the same type of actions from videos of another data collection. As a demonstration, we build a hide-patch experiment which intends to cover irrelevant object in the background scene in hand-hygiene actions and observe the performance change from hand-hygiene model. The results prove that the appearance in background could be mistakenly considered as discriminative cues towards hand-hygiene actions. This is one of the main reasons to prevent hand-hygiene actions to be recognized in different data collections.

Furthermore, we propose a multi-modalities system to recognize hand-hygiene actions in cross scenarios. We experiment the capabilities of optical flow, hand segmentation mask, and human skeleton joints' on recognizing certain hand-hygiene action types. And we combine all these separate modalities as K separate classifiers to recognize hand-hygiene actions in a hierarchical way. Our result indicate this multi-modalities system outperforms the single RGB modality method on detecting cross scenario hand-hygiene actions.

In the future, we plan to explore and find a more robust camera view for hand-hygiene action recognition. In our previous work [86][98], the egocentric camera views of chest and nose, they share a common advantage on capturing subtle-motion level details in all the food handling actions. Therefore, if a food handling task involves subtle motions or requires to capture the texture details on objects, egocentric camera view is a good selection. However, the shortcomings of egocentric camera view, such as target out of camera view and uncomfortable for long time wearing, is also obvious. In contrast, static third camera view offers a burden-free data collection experience to the participants and capable at capturing many of the hand-hygiene actions. However, it is unavoidable to loss the subtle information on hands with the side third person camera view as we introduced in Figure 3.17 and 3.18. Based on the above, we would like to find a new camera view which mounts the camera equipment in the front of the sink area. The camera view is expected to capture all the action details as an egocentric camera without interfering the subject from focusing on their actions.

7. ADDITIONAL EXPLORATION OF ACTION RECOGNITION ON VIDEO QUALITY

Besides the previous exploration of hand-hygiene actions with different approach methods, I also worked on an additional exploration of action recognition with video quality variation. The content in this chapter is an extended version of my previous work [132]. Executing video analytics tasks using a large camera network is a challenging problem in the field of video processing. Video compression is a necessary step to reduce video data size before transmission. However, the performance of video analytics tasks generally degrade as video quality drops.

In this chapter, we explore the impact of compression on detection accuracy in activity recognition. We explore this using different sets of activities, and show that each activity is affected by compression differently and the impact of compression depends on the "neighboring" activities from which this activity is to be distinguished. Moreover, we propose a video analytics system corresponding to the task of activity recognition using compressed videos. We use feature descriptors to predict activity recognition task success or failure under different QP values. With this prediction result, the system then selects an optimal compression rate for each input video. And therefore, an acceptable detection accuracy and video data bitrate can be achieved. The chapter is organized as following: Section 7.1 provides the data selection of this work. Section 7.2 explores the impact of compression on activity recognition. Section 7.3 describes the design details of the video analytics system. Section 7.4 demonstrates the system can perform better than one that uses the same QP to compress all videos and Section 7.5 concludes the paper.

7.1 Data selection

Compare to the particular hand-hygiene action task in previous chapters, the work in this chapter focuses on the action recognition task in general category. Thus, we need to select appropriate data to explore this topic. The dataset we used is UCF101 [71] dataset, which contains 101 action class categories from 5 types, which are "Human-Object Interaction",

Table 7.1. Activity names of five different sets

Activity Set A	***Activity Set B***	***Activity Set C***	***Activity Set D***	***Activity Set E***
ApplyEyeMakeup	ApplyEyeMakeup	ApplyEyeMakeup	Archery	ApplyEyeMakeup
ApplyLipstick	ApplyLipstick	ApplyLipstick	BabyCrawling	BreastStroke
Archery	Archery	Archery	BandMarching	Fencing
BabyCrawling	BabyCrawling	BabyCrawling	HorseRace	Haircut
BalanceBeam	BalanceBeam	BalanceBeam	JugglingBalls	IceDancing
BandMarching	BandMarching	BandMarching	MoppingFloor	MilitaryParade
JugglingBalls	Rowing	JugglingBalls	PlayingSitar	PlayingDhol
Basketball	Basketball	PlayingCello	Punch	SalsaSpin
Kayaking	Kayaking	PlayingSitar	Rowing	TaiChi
BenchPress	BenchPress	Rowing	YoYo	WalkingWithDog

”Body-Motion Only”, ”Human-Human Interaction”, ”Playing Musical Instruments”, and ”Sports”. To explore the impact of the set of activities, we use different subsets of 10 activities chosen among the entire collection of 101 activities. We create five different activity sets which are listed in Table 7.1. There are small variations between the activities chosen for **Sets A**, **B**, **C** and **D**, while in **Set E**, most of the activities are completely different from the other sets.

7.2 Impact of compression

To investigate how video compression influences activity recognition, we encode each video using Mencoder [133] into five different compressed versions in H.264 format, each using among a constant QP (Quantization Parameter) from the list {20,26,32,38,44}. A large QP value generates a low bitrate, which results in low video quality. In our designed method, we train only one SVM for each activity class and use it on videos from that class for all QP values. As it is indicated in Figure 7.1, the general trend of the fixed QP curves shows decreasing performance as the bitrate drops, especially below 100 kbit/s. And it is very different across each test set. In addition, there exist a few surprises. For instance, on the fixed QP curve of **Set C**, the performance degrades as bitrate increases in the middle range of the curve.

To understand more about these observations, we analyze the impact of video compression on each activity class in **Set A**. The impact of compression for each activity is evaluated as the average of all test videos' confidence score from SVM. The results are demonstrated in Figure 7.2 and Figure 7.3. Figure 7.2 represents detection results from each individual activity class in **Activity set A**. It is obvious as the video quality degrades (QP increases), most actions' detection confidence score decrease, but each to a different degree. In Figure 7.2, the confidence score for the activity **"ApplyLipstick"** drops 34% as the QP increases from 20 to 44. However, the confidence score for activity **"BalanceBeam"** drops only 4%. Moreover, the activity **"JugglingBalls"** increases its confidence score as the compression increases from QP value 20 to QP value 44. This implies that compression actually makes this activity class easier to identify, and accounts for the decrease in accuracy across the entire set shown in Figure 7.1 as the QP increases from 38 to 44. Therefore, from Figure 7.2, we see that for the task of activity recognition, the impact of compression depends heavily on the specific activity class.

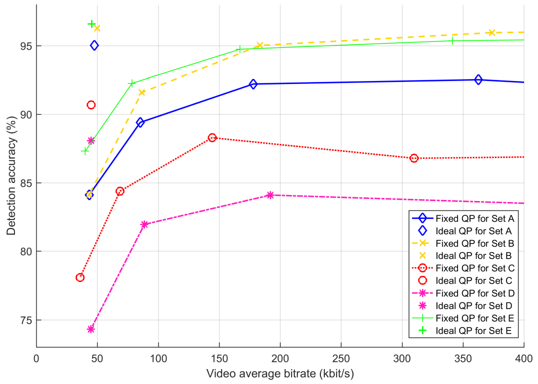


Figure 7.1. Activity Set A to E ideal QP points and Fixed QP curves

We also observe that different combinations of activity classes influences detection accuracy. **Set A**, **Set B**, **Set C** and **Set D** all include the activity class **"BabyCrawling"**. In Figure 7.3, all of them start with the same confidence score around 81%, but as QP increases to 44, the confidence score of class **"BabyCrawling"** in **Set A**, **Set B**, **Set C** and **Set D** decreases: 26.3%, 22.3 %, 16.7% and 33.3%. This indicates that the impact of compression on the detection performance of each activity depends on the set of other activities.

As these results show, when considering the impact of compression on activity recognition, it is important to consider the impact both on each individual activity and due to different collections of activities. Therefore, in this paper, we propose a system that predicts the optimal amount of compression for each individual video.

Consider a system that could compress each individual video i using an ideal quantizer, QP_{i}^{*}, that corresponds to the largest QP in our list that produces the correct detection result for that video. While such a selection may not be possible in practice, performance of such a system can demonstrate whether overall performance could be improved if the amount of compression could be optimally chosen for each input video.

To demonstrate the power of such a system, we define the concept of ideal QP point. The ideal QP point represents the result where all the test videos are compressed to the lowest quality to be detectable by **"Improved Dense Trajectory" (IDT)** activity recognition

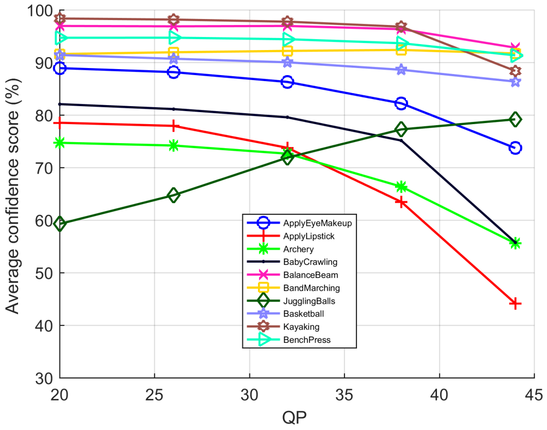


Figure 7.2. Set A confidence score versus QP

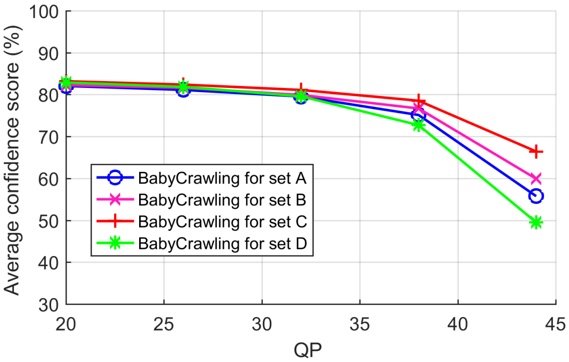


Figure 7.3. "BabyCrawling" confidence score versus QP

algorithm. The ideal points for each set are also shown in Figure 7.1. Compared to the fixed QP curves, ideal points demonstrate promising performance both in bitrate saving and accuracy improvement. These ideal points are obtainable only with perfect knowledge. Therefore they provide the upper bounds on the performance of our system.

7.3 Prediction system

In this section, we present our prediction system whose goal is to predict each input video's performance under different compression QP values and select the optimal QP value for each video. The proposed system is illustrated in Figure 7.4. The main components are: Feature extraction, Hierarchical K-means, Random Forest and Compression rate selection. The system starts with feature extraction from all compressed versions of the input video, and then applies the visual word assignment pipeline [134] to assign words to each descriptor. The resulting histogram represents the video. After that, the histogram is input to the trained Random Forest to receive a classification result whether the detection performance is **"success"** or **"failure"** for a given QP value. The final step is to collect the classification results from the previous step and select the optimal QP. The following sections will describe each component in detail.

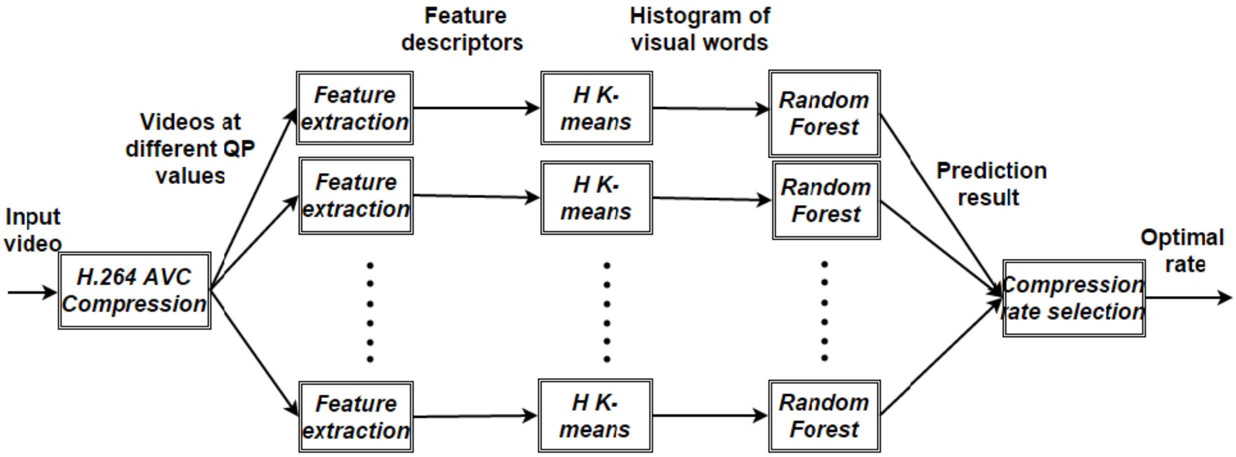


Figure 7.4. Prediction system pipeline

Feature Extraction Texture features normally include representative information about the video. In order to find an appropriate feature for the prediction system, we evaluated four different types of features in this paper. According to [15], densely sampled features have the best performance on complex datasets. Therefore, all the features evaluated in this paper are densely sampled. We selected HOG (Histogram of oriented gradients) [134], HOF (Histogram of oriented flow) [134], MBH (Motion Boundary Histograms) [134][14] and SIFT (Scale-invariant feature transform) [135][136] to test our prediction system.

Visual word assignment Densely sampled features extracted from different videos normally have a different amount of descriptors due to the video length. In our system, we choose to use hierarchical k-means [134] to assign each video with an equal length histogram of visual words.

Random Forest A Random Forest [137] includes a collection of decision trees, where the growth of trees and the split of nodes both depend on random selection. An input vector proceeds through each tree to receive a decision vote. After collecting all the votes, the forest selects the class that receives the most votes as the final decision. In our prediction system, for a given set of activities, we design five Random Forests, one for each QP value considered. Each Random Forest predicts success or failure of the activity recognition task for a given input video compressed using that QP.

To train each Random Forest for this classification problem, we require feature inputs and the correspond labels. As feature input for one video, it is the histogram of visual words from the last step. For feature labels, we use the confidence score of the SVM predictor in the activity recognition task. The score indicates the probability that a video belongs to its ground truth activity class. In our system, if the score is higher than 0.5, we denote it as **"success"**. Otherwise, we denote it as **"failure"**.

From the perspective of training our system, we need to train both the activity recognition **IDT** algorithm and our prediction system. We split the 25 groups of videos inside each activity class into two parts. Group 01 to 12 is used for testing the **IDT** algorithm and 13 to 25 for training the **IDT** algorithm. Furthermore, to avoid training the Random Forests on the same data used to train the **IDT**, we randomly splits groups of 01 to 12 into two parts equally, one part for training the Random Forests, another part for testing them. The

Table 7.2. Random Forest training and testing samples for Set A and E

	success	*failure*
Number of Set A Training samples	1365	230
Number of Set A Testing samples	1437	168
Number of Set E Training samples	1464	136
Number of Set E Testing samples	1499	116

prediction result for each video is a 1x5 vector indicating **"failure"** or **"success"**, where each element in the vector corresponds to one QP value.

Compression rate selection After each Random Forest predicts whether a correct decision will be made at each QP considered, we select the estimated $\hat{QP}_{\mathrm{i}}$ to be the largest QP that yields a **"success"** prediction.

Examining the fixed QP curves in Figure 7.1, we notice that the detection accuracy drops gradually at high bitrate and sharply at low bitrate. Therefore, it is usually good to conservatively trade a small amount of bitrate for a relatively large detection accuracy increment. Therefore, in some extreme cases, when all QPs lead to a **"failure"** prediction, we conservatively compress these videos using QP 20, which corresponds to the highest video quality and bitrate.

7.4 Experimental Result

In this section, we construct an evaluation method whose goal is to test the performance of our prediction system. The prediction system's pipeline has already been discussed in Section 7.3.

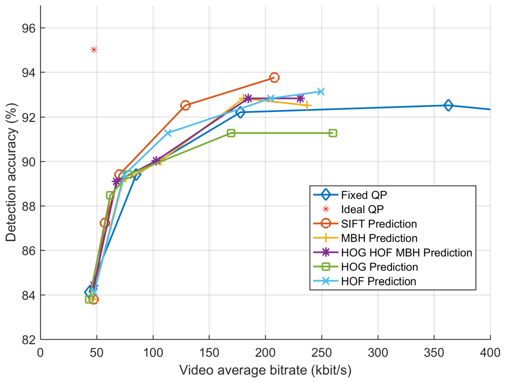


Figure 7.5. Prediction result from Set A

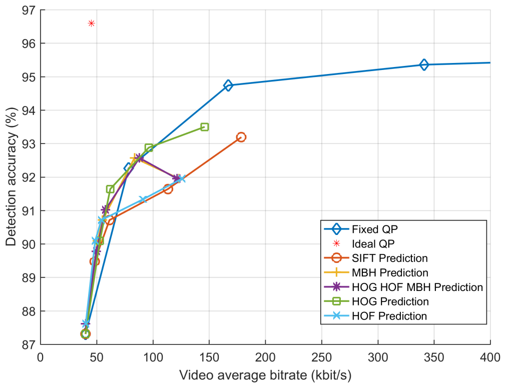


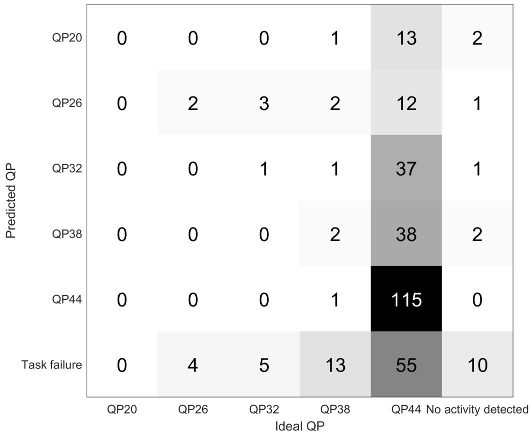
Figure 7.6. Prediction result from Set E

In addition to the fixed QP curve and ideal QP point as we discussed Section 7.2, we tested our prediction system with each of the different features mentioned in Section 7.3. However, as shown in Table 7.2, we can see that the number of overall samples among all QP values for **"success"** and **"failure"** are quite imbalanced. The **"failure"** cases only occupy a small proportion of the total samples. This is a crucial factor that limits our prediction accuracy [138]. To reduce the impact of this imbalance, we pre-assigned class weights to the Random Forest during training to increase the importance of the minority class. As the assigned class weight varies, we plot each feature's prediction result as a curve.

Figure 7.5 and 7.6 examine the performance of an entire activity recognition system that incorporates our prediction results, for activities **Set A** and **Set E** respectively. For **Set A**, most of our features have a better performance than the fixed QP curve, both in accuracy and compression rate. For **Set E**, our prediction has a better performance compared with the fixed QP curve only when the bitrate is less than 100 kbit/s, even though the accuracy from the prediction is still significantly lower than the ideal point.

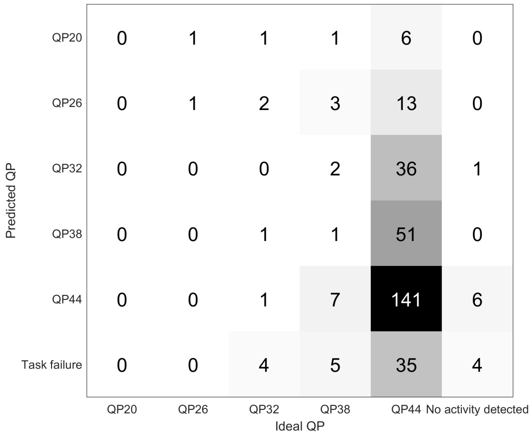
To interpret the result in Figure 7.5 and 7.6, we present confusion matrices of the ideal QP value (horizontal axis) and predicted QP value (vertical axis). In Figure 7.7 and 7.8, respectively. Both matrices reflect the prediction point of highest accuracy. For **Set A**, this is the right-most point on the SIFT curve. For **Set E**, this is the right-most point on the HOG curve.

The most important distinction we can find is the difference in the ratio of **“Task failure”** between these two sets. Since, our prediction system is limited by the imbalanced data, there exists some videos’ prediction results that are mistakenly predicted as all **“failure”** across all QP values and marked as **“Task failure”**, even though an ideal QP exist. As we described in section 4, it is worthwhile to conservatively compress these videos with QP 20 to increase the chance that the activity is correctly detected while only sacrificing a small amount of bitrate. As we noticed, the percentage of **“Task failure”** videos for **Set A** and **Set E** are 27.10 % and 14.86 %.



Predicted QP \ Ideal QP	QP20	QP26	QP32	QP38	QP44	No activity detected
QP20	0	0	0	1	13	2
QP26	0	2	3	2	12	1
QP32	0	0	1	1	37	1
QP38	0	0	0	2	38	2
QP44	0	0	0	1	115	0
Task failure	0	4	5	13	55	10

Figure 7.7. Confusion matrix for SIFT from Set A



Predicted QP \ Ideal QP	QP20	QP26	QP32	QP38	QP44	No activity detected
QP20	0	1	1	1	6	0
QP26	0	1	2	3	13	0
QP32	0	0	0	2	36	1
QP38	0	0	1	1	51	0
QP44	0	0	1	7	141	6
Task failure	0	0	4	5	35	4

Figure 7.8. Confusion matrix for HOG from Set E

Moreover, videos that have higher predicted QPs than their ideal QPs are considered to be aggressively predicted. These aggressively predicted videos appear in the lower triangular region below the diagonal in the matrices of Figure 7.7 and 7.8. Comparing these matrices, the percentage of aggressively predicted videos for **Set A** and **Set E** are 0.31 % and 2.78 %. These videos will certainly fail at the activity recognition. Therefore, it is understandable for **Set E** to have a worse performance than **Set A**.

We also compare the performance of each feature in our prediction system. For example, in Figure 7.5, SIFT feature has a better detection accuracy compared with other features over all bitrates. Therefore, SIFT has the best performance for **Set A**. However, after checking the performance of each feature in all sets, we notice that the feature HOF (Histogram of oriented flow) has the best performance in three out of five of these activity sets. Thus while more exploration is needed, the HOF may be the most reliable feature for our prediction system.

7.5 Conclusion

In this chapter, we proposed a system to predict each video's optimal compression rate for the task of activity recognition. We explored the effect of compression on the performance of activity recognition using different sets of activities. We also defined the concept of an ideal QP point and fixed QP curve to help evaluate the performance of our prediction system. The ideal QP point provides a great deal of promise that significant gains might be possible in both detection accuracy and bitrate. Through our experiments, we were able to generate an acceptable prediction result for some of the combinations of activities. For other combinations, we analyzed the potential factors which limit the performance of our system. There still exists potential space to explore in this field.

8. CONCLUSIONS AND FUTURE WORK

This chapter summarizes the projects in the thesis and provides future insights of food handling in video analytics. In Section 8.1, we summarize this thesis in two different perspectives. In Section 8.2, we propose our directions of the future work.

8.1 Conclusions

8.1.1 Human machine trade-off

In this thesis, we propose to build a video monitoring system to assist food handling, which involves both human and machine. In general, a fully automated system relies on the machine to take the majority of the responsibility and relieve a human from their work. However, in food handling, a human is also responsible to put their effort to support and ensure the machine to get the video monitoring job done.

Food handling, especially hand-hygiene in food handling, is a type of activity which requires its participants to have willingness to collaborate. In other words, it is the participant's responsibility, not the machine's, to clearly demonstrate each step of his/her action in front of the video monitor to pass the quality check. In Chapter 4 and 5, we compared the performance of three camera views with the same CNN model architecture. Egocentric camera views such as chest and nose camera always experience with different level of hands or objects out of camera view issues. The reason for issues is that different participant has difference in body shape and personal habits. It is difficult to set a uniform standard to apply on every individual to clearly capture all their hand-hygiene steps. Therefore, it is important for participants to ensure their actions are under video monitoring rather than designing an omnipotent algorithm to overcome problematic input data.

8.1.2 Camera view influence

Food handling as a sequence of actions can be addressed with different video monitoring tasks. For example, as we defined in Chapter 3.3, the hand-hygiene actions include three levels of tasks. According to the task's difficult level, one could select to use the most appropriate camera view.

In this thesis, we explore three different types of camera views, as defined in Chapter 3.2, for food handling. For the egocentric camera views of chest and nose, they share a common advantage on capturing subtle-motion level details in all the food handling actions. Therefore, if a food handling task involves subtle motions or requires to capture the texture details on objects, egocentric camera view is a good selection. However, the shortcoming of egocentric camera view is also obvious. Because the camera is equipped on human subject, the video recording quality heavily depends on the behavior of that subject. An effortless participant could record the video input in inappropriate manners which could cause video monitoring system fails at detecting and reporting the actual food handling quality. Moreover, the equipment to tie a camera onto a participant could cause an uncomfortable experience. Especially for produce washing, which requires a labor to hold the same pose for more than half an hour, the camera equipment becomes a burden to the subject.

In contrast, static third person camera view which mounts on a flat platform frees the subject from carrying the camera equipment. Based on our results in Chapter 5.3.4, even though the wall camera is less accurate on certain subtle motion actions, it is still a reasonable camera choice for the ”standard level” hand-hygiene task.

8.2 Future work

In the future, we plan to improve our current hand-hygiene system to address the conclusions in Section 8.1.1 and 8.1.2. First, to better engage a subject to perform food handling without disturb video monitoring system, our video monitoring system is expected to remind a subject to repeat its actions until all the actions can be clearly captured by the system. Second, we would explore a new camera view which mounts the camera equipment in the front of the sink area. This camera view is expected to capture all the action details as an egocentric camera without interfering the subject from focusing on their actions.

REFERENCES

[1] L. McIntyre, L. Vallaster, L. Wilcott, S. B. Henderson, and T. Kosatsky, "Evaluation of food safety knowledge, attitudes and self-reported hand washing practices in foodsafe trained and untrained food handlers in British Columbia, Canada," *Food Control*, vol. 30, no. 1, pp. 150–156, 2013.

[2] B. Michaels, C. Keller, M. Blevins, G. Paoli, T. Ruthman, E. Todd, and C. J. Griffith, "Prevention of food worker transmission of foodborne pathogens: Risk assessment and evaluation of effective hygiene intervention strategies," *Food Service Technology*, vol. 4, no. 1, pp. 31–49, 2004.

[3] F. G. P. S. Challenge, "WHO guidelines on hand hygiene in health care: A summary," *World Health Organization, Geneva, Switzerland*, 2009.

[4] A. Betancourt, P. Morerio, C. S. Regazzoni, and M. Rauterberg, "The evolution of first person vision methods: A survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 5, pp. 744–760, 2015.

[5] B. Ghanem, J. C. Niebles, C. Snoek, F. C. Heilbron, H. Alwassel, V. Escorcia, R. Khrisna, S. Buch, and C. D. Dao, "The activitynet large-scale activity recognition challenge 2018 summary," *arXiv preprint arXiv:1808.03766*, 2018.

[6] X. Dai, B. Singh, G. Zhang, L. S. Davis, and Y. Q. Chen, "Temporal context network for activity localization in videos," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 5727–5736.

[7] Y. Zhao, Y. Xiong, L. Wang, Z. Wu, X. Tang, and D. Lin, "Temporal action detection with structured segment networks," *IEEE International Conference on Computer Vision (ICCV), Oct*, vol. 2, 2017.

[8] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Computing Surveys (CSUR)*, vol. 43, no. 3, p. 16, 2011.

[9] Y. Kong and Y. Fu, "Human action recognition and prediction: A survey," *arXiv preprint arXiv:1806.11230*, 2018.

[10] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[11] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," 2008.

[12] C. Schuldt, I. Laptev, and B. Caputo, “Recognizing human actions: A local svm approach,” in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, IEEE, vol. 3, 2004, pp. 32–36.

[13] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR*, vol. 1, 2005, pp. 886–893.

[14] N. Dalal, B. Triggs, and C. Schmid, “Human detection using oriented histograms of flow and appearance,” in *European Conference on Computer Vision*, Springer, 2006, pp. 428–441.

[15] H. Wang, M. M. Ullah, A. Klaser, I. Laptev, and C. Schmid, “Evaluation of local spatio-temporal features for action recognition,” in *BMVC 2009-British Machine Vision Conference*, BMVA Press, 2009, pp. 124–1.

[16] H. Wang and C. Schmid, “Action recognition with improved trajectories,” in *IEEE International Conference on Computer Vision*, 2013, pp. 3551–3558.

[17] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, “Dense trajectories and motion boundary descriptors for action recognition,” *International journal of computer vision*, vol. 103, no. 1, pp. 60–79, 2013.

[18] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, “Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations,” in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 609–616.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[20] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *In ICLR*, 2015.

[21] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[22] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[23] M. Schuster and K. K. Paliwal, “Bidirectional recurrent neural networks,” *IEEE transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[24] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, “Convolutional lstm network: A machine learning approach for precipitation nowcasting,” *arXiv preprint arXiv:1506.04214*, 2015.

[25] B. Zhou, A. Andonian, A. Oliva, and A. Torralba, “Temporal relational reasoning in videos,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 803–818.

[26] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, “Learning spatiotemporal features with 3d convolutional networks,” in *IEEE International Conference on Computer Vision*, 2015, pp. 4489–4497.

[27] J. Carreira and A. Zisserman, “Quo vadis, action recognition? a new model and the kinetics dataset,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[28] A. Diba, M. Fayyaz, V. Sharma, A. H. Karami, M. M. Arzani, R. Yousefzadeh, and L. Van Gool, “Temporal 3D convnets: New architecture and transfer learning for video classification,” *arXiv preprint arXiv:1711.08200*, 2017.

[29] Z. Qiu, T. Yao, and T. Mei, “Learning spatio-temporal representation with pseudo-3D residual networks,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5533–5541.

[30] H. Pirsiavash and D. Ramanan, “Detecting activities of daily living in first-person camera views,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 2847–2854.

[31] Y. Li, Z. Ye, and J. M. Rehg, “Delving into egocentric actions,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 287–295.

[32] M. Ma, H. Fan, and K. M. Kitani, “Going deeper into first-person activity recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1894–1903.

[33] S. Singh, C. Arora, and C. Jawahar, “First person action recognition using deep learned descriptors,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2620–2628.

[34] A. Fathi, X. Ren, and J. M. Rehg, “Learning to recognize objects in egocentric activities,” in *IEEE Conference On Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 3281–3288.

[35] K. Simonyan and A. Zisserman, “Two-stream convolutional networks for action recognition in videos,” in *Advances in Neural Information Processing Systems*, 2014, pp. 568–576.

[36] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Val Gool, “Temporal segment networks: Towards good practices for deep action recognition,” in *ECCV*, 2016.

[37] S. Yan, Y. Xiong, and D. Lin, “Spatial temporal graph convolutional networks for skeleton-based action recognition,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[38] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, “An end-to-end spatio-temporal attention model for human action recognition from skeleton data,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, 2017.

[39] C. Li, Q. Zhong, D. Xie, and S. Pu, “Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation,” *arXiv preprint arXiv:1804.06055*, 2018.

[40] L. Meng, B. Zhao, B. Chang, G. Huang, W. Sun, F. Tung, and L. Sigal, “Interpretable spatio-temporal attention for video action recognition,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[41] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, “Openpose: Realtime multi-person 2d pose estimation using part affinity fields,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 1, pp. 172–186, 2019.

[42] G. Chen, D. Clarke, M. Giuliani, A. Gaschler, D. Wu, D. Weikersdorfer, and A. Knoll, “Multi-modality gesture detection and recognition with un-supervision, randomization and discrimination,” in *European Conference on Computer Vision*, Springer, 2014, pp. 608–622.

[43] S. Escalera, X. Baró, J. Gonzalez, M. A. Bautista, M. Madadi, M. Reyes, V. Ponce-López, H. J. Escalante, J. Shotton, and I. Guyon, “Chalearn looking at people challenge 2014: Dataset and results,” in *European Conference on Computer Vision*, Springer, 2014, pp. 459–473.

[44] N. Neverova, C. Wolf, G. W. Taylor, and F. Nebout, “Multi-scale deep learning for gesture detection and localization,” in *European Conference on Computer Vision*, Springer, 2014, pp. 474–490.

[45] L. Pigou, A. Van Den Oord, S. Dieleman, M. Van Herreweghe, and J. Dambre, "Beyond temporal pooling: Recurrence and temporal convolutions for gesture recognition in video," *International Journal of Computer Vision*, vol. 126, no. 2-4, pp. 430–439, 2018.

[46] D. Wu, L. Pigou, P.-J. Kindermans, N. D.-H. Le, L. Shao, J. Dambre, and J.-M. Odobez, "Deep dynamic neural networks for multimodal gesture segmentation and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 8, pp. 1583–1597, 2016.

[47] N. Granger and M. A. el Yacoubi, "Comparing hybrid nn-hmm and rnn for temporal modeling in gesture recognition," in *International Conference on Neural Information Processing*, Springer, 2017, pp. 147–156.

[48] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: Realtime multiperson 2d pose estimation using part affinity fields," *arXiv preprint arXiv:1812.08008*, 2018.

[49] A. Mathis, P. Mamidanna, K. M. Cury, T. Abe, V. N. Murthy, M. W. Mathis, and M. Bethge, "Deeplabcut: Markerless pose estimation of user-defined body parts with deep learning," *Nature neuroscience*, vol. 21, no. 9, p. 1281, 2018.

[50] Y.-W. Chao, Y. Liu, X. Liu, H. Zeng, and J. Deng, "Learning to detect human-object interactions," in *2018 ieee winter conference on applications of computer vision (wacv)*, IEEE, 2018, pp. 381–389.

[51] H.-S. Fang, J. Cao, Y.-W. Tai, and C. Lu, "Pairwise body-part attention for recognizing human-object interactions," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 51–67.

[52] C. Gao, Y. Zou, and J.-B. Huang, "Ican: Instance-centric attention network for human-object interaction detection," *arXiv preprint arXiv:1808.10437*, 2018.

[53] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[54] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 618–626.

[55] J. Park, S. Woo, J.-Y. Lee, and I. S. Kweon, "Bam: Bottleneck attention module," *arXiv preprint arXiv:1807.06514*, 2018.

[56] K. Li, Z. Wu, K.-C. Peng, J. Ernst, and Y. Fu, “Tell me where to look: Guided attention inference network,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9215–9223.

[57] S. J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[58] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, “Domain-adversarial training of neural networks,” *The journal of machine learning research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[59] M. Ghifary, W. B. Kleijn, and M. Zhang, “Domain adaptive neural networks for object recognition,” in *Pacific Rim international conference on artificial intelligence*, Springer, 2014, pp. 898–904.

[60] B. Sun and K. Saenko, “Deep coral: Correlation alignment for deep domain adaptation,” in *European conference on computer vision*, Springer, 2016, pp. 443–450.

[61] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, “Adversarial discriminative domain adaptation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7167–7176.

[62] J. Munro and D. Damen, “Multi-modal domain adaptation for fine-grained action recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 122–132.

[63] M.-H. Chen, Z. Kira, G. AlRegib, J. Yoo, R. Chen, and J. Zheng, “Temporal attentive alignment for large-scale video domain adaptation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6321–6330.

[64] H. Huang, Q. Huang, and P. Krahenbuhl, “Domain transfer through deep activation matching,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 590–605.

[65] Y.-W. Chao, S. Vijayanarasimhan, B. Seybold, D. A. Ross, J. Deng, and R. Sukthankar, “Rethinking the faster r-cnn architecture for temporal action localization,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1130–1139.

[66] J. Gao, Z. Yang, K. Chen, C. Sun, and R. Nevatia, “Turn tap: Temporal unit regression network for temporal action proposals,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 3628–3636.

[67] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 156–165.

[68] Y. A. Farha and J. Gall, "Ms-tcn: Multi-stage temporal convolutional network for action segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3575–3584.

[69] Y. J. Lee, J. Ghosh, and K. Grauman, "Discovering important people and objects for egocentric video summarization," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 1346–1353.

[70] E. H. Spriggs, F. De La Torre, and M. Hebert, "Temporal segmentation and activity classification from first-person sensing," in *IEEE Computer Society Conference On Computer Vision and Pattern Recognition Workshops, CVPR*, 2009, pp. 17–24.

[71] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[72] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2011.

[73] Z. Luo, B. Peng, D.-A. Huang, A. Alahi, and L. Fei-Fei, "Unsupervised learning of long-term motion dynamics for videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2203–2212.

[74] F. Mueller, D. Mehta, O. Sotnychenko, S. Sridhar, D. Casas, and C. Theobalt, "Real-time hand tracking under occlusion from an egocentric rgb-d sensor," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1284–1293.

[75] S. Yeung, A. Alahi, A. Haque, B. Peng, Z. Luo, A. Singh, T. Platchek, A. Milstein, and F.-F. Li, "Vision-based hand hygiene monitoring in hospitals.," in *American Medical Informatics Association*, 2016.

[76] A. Haque, M. Guo, A. Alahi, S. Yeung, Z. Luo, A. Rege, J. Jopling, L. Downing, W. Beninati, A. Singh, *et al.*, "Towards vision-based smart hospitals: A system for tracking and monitoring hand hygiene compliance," *arXiv preprint arXiv:1708.00163*, 2017.

[77] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele, "A database for fine grained activity detection of cooking activities," in *2012 IEEE conference on computer vision and pattern recognition*, IEEE, 2012, pp. 1194–1201.

[78] S. Stein and S. J. McKenna, “Combining embedded accelerometers with computer vision for recognizing food preparation activities,” in *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, 2013, pp. 729–738.

[79] H. Kuehne, A. Arslan, and T. Serre, “The language of actions: Recovering the syntax and semantics of goal-directed human activities,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 780–787.

[80] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” *arXiv preprint arXiv:1506.01497*, 2015.

[81] J. Redmon, S. Divvala, R. Girshick, and 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.

[82] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” in *European conference on computer vision*, Springer, 2016, pp. 21–37.

[83] T.-J. Fu, S.-H. Tai, and H.-T. Chen, “Attentive and adversarial learning for video summarization,” in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, 2019, pp. 1579–1587.

[84] M. Otani, Y. Nakashima, E. Rahtu, and J. Heikkila, “Rethinking the evaluation of video summaries,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7596–7604.

[85] J. Meng, S. Wang, H. Wang, J. Yuan, and Y.-P. Tan, “Video summarization via multiview representative selection,” in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 1189–1198.

[86] C. Zhong, A. R. Reibman, H. M. Cordoba, and A. J. Deering, “Hand-hygiene activity recognition in egocentric video,” in *2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP)*, IEEE, 2019, pp. 1–6.

[87] B. Ghanem, J. Niebles, C. Snoek, F. C. Heilbron, H. Alwassel, V. Escorcia, R. Krishna, S. B, and C. D. Dao, “The activitynet large-scale activity recognition challenge 2018 summary,” *CoRR*, vol. abs/1808.03766, 2018. arXiv: 1808.03766. [Online]. Available: http://arxiv.org/abs/1808.03766.

[88] Y. Poleg, C. Arora, and S. Peleg, “Temporal segmentation of egocentric videos,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2537–2544.

[89] G. Farnebäck, “Two-frame motion estimation based on polynomial expansion,” in *Scandinavian conference on Image analysis*, Springer, 2003, pp. 363–370.

[90] C. Zach, T. Pock, and H. Bischof, “A duality based approach for realtime tv-l 1 optical flow,” in *Joint pattern recognition symposium*, Springer, 2007, pp. 214–223.

[91] L. Wang, Y. Xiong, Z. Wang, and Y. Qiao, “Towards good practices for very deep two-stream convnets,” *Computing Research Repository*, vol. abs/1507.02159, 2015. [Online]. Available: http://arxiv.org/abs/1507.02159.

[92] Y. Zhu, *Pytorch implementation of popular two-stream frameworks for video action recognition*, https://github.com/bryanyzhu/two-stream-pytorch, 2017.

[93] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *Computer Vision and Pattern Recognition*, IEEE, 2009, pp. 248–255.

[94] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, “Long-term recurrent convolutional networks for visual recognition and description,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2625–2634.

[95] S. Sharma, R. Kiros, and R. Salakhutdinov, “Action recognition using visual attention,” *arXiv preprint arXiv:1511.04119*, 2015.

[96] H. Sak, A. Senior, and F. Beaufays, “Long short-term memory recurrent neural network architectures for large scale acoustic modeling,” in *Fifteenth Annual Conference of The International Speech Communication Association*, 2014.

[97] C. Li and K. M. Kitani, “Pixel-level hand detection in ego-centric videos,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3570–3577.

[98] C. Zhong, A. R. Reibman, H. A. Mina, and A. J. Deering, “Multi-view hand-hygiene recognition for food safety,” *Journal of Imaging*, vol. 6, no. 11, p. 120, 2020.

[99] A. Krizhevsky, G. Hinton, *et al.*, “Learning multiple layers of features from tiny images,” 2009.

[100] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *arXiv preprint arXiv:1704.04861*, 2017.

[101] C. Zhang, S.-C. Chen, M.-L. Shyu, and S. Peeta, “Adaptive background learning for vehicle detection and spatio-temporal tracking,” in *Fourth International Conference on Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint*, IEEE, vol. 2, 2003, pp. 797–801.

[102] F. El Baf, T. Bouwmans, and B. Vachon, “Fuzzy integral for moving object detection,” in *2008 IEEE International Conference on Fuzzy Systems (IEEE World Congress on Computational Intelligence)*, IEEE, 2008, pp. 1729–1736.

[103] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, “Subsense: A universal change detection method with local adaptive sensitivity,” *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 359–373, 2014.

[104] A. Sobral, “BGSLibrary: An opencv c++ background subtraction library,” in *IX Workshop de Visão Computacional (WVC’2013)*, Rio de Janeiro, Brazil, Jun. 2013. [Online]. Available: https://github.com/andrewssobral/bgslibrary.

[105] M. A. R. Ahad, J. K. Tan, H. Kim, and S. Ishikawa, “Motion history image: Its variants and applications,” *Machine Vision and Applications*, vol. 23, no. 2, pp. 255–281, 2012.

[106] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *arXiv preprint arXiv:1502.03167*, 2015.

[107] M. Bolaños, M. Garolera, and P. Radeva, “Video segmentation of life-logging videos,” in *International Conference on Articulated Motion and Deformable Objects*, Springer, 2014, pp. 1–9.

[108] R. Azad, M. Asadi-Aghbolaghi, S. Kasaei, and S. Escalera, “Dynamic 3d hand gesture recognition by learning weighted depth motion maps,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 6, pp. 1729–1740, 2018.

[109] D. Huang, C. Shan, M. Ardabilian, Y. Wang, and L. Chen, “Local binary patterns and its application to facial image analysis: A survey,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, no. 6, pp. 765–781, 2011.

[110] C. Zhong, A. R. Reibman, H. M. Cordoba, and A. J. Deering, “Robust hand-hygiene video processing in cross-dataset video for food safety,” 2021.

[111] L. Wang, Y. Xiong, Z. Wang, and Y. Qiao, “Towards good practices for very deep two-stream convnets,” *arXiv preprint arXiv:1507.02159*, 2015.

[112] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4694–4702.

[113] O. Köpüklü, F. Herzog, and G. Rigoll, "Comparative analysis of cnn-based spatiotemporal reasoning in videos," *arXiv preprint arXiv:1909.05165*, 2019.

[114] T. Tommasi, N. Patricia, B. Caputo, and T. Tuytelaars, "A deeper look at dataset bias," in *Domain adaptation in computer vision applications*, Springer, 2017, pp. 37–55.

[115] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European conference on computer vision*, Springer, 2014, pp. 818–833.

[116] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, and S. Yan, "Object region mining with adversarial erasing: A simple classification to semantic segmentation approach," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1568–1576.

[117] K. K. Singh and Y. J. Lee, "Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization," in *2017 IEEE international conference on computer vision (ICCV)*, IEEE, 2017, pp. 3544–3553.

[118] J.-F. Hu, W.-S. Zheng, J. Pan, J. Lai, and J. Zhang, "Deep bilinear learning for rgb-d action recognition," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 335–351.

[119] Y. Kong, Z. Ding, J. Li, and Y. Fu, "Deeply learned view-invariant features for cross-view action recognition," *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 3028–3037, 2017.

[120] Y. Liu, Z. Lu, J. Li, and T. Yang, "Hierarchically learned view-invariant representations for cross-view action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 8, pp. 2416–2430, 2018.

[121] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "Ntu rgb+ d: A large scale dataset for 3d human activity analysis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1010–1019.

[122] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *CVPR*, 2017.

[123] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *arXiv preprint arXiv:2001.05566*, 2020.

[124] A. Urooj and A. Borji, "Analysis of hand segmentation in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4710–4719.

[125] G. Lin, A. Milan, C. Shen, and I. Reid, "Refinenet: Multi-path refinement networks with identity mappings for high-resolution semantic segmentation. arxiv 2016," *arXiv preprint arXiv:1611.06612*,

[126] V. Nekrasov, C. Shen, and I. Reid, "Light-weight refinenet for real-time semantic segmentation," *arXiv preprint arXiv:1810.03272*, 2018.

[127] X. Chen, R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, and A. Yuille, "Detect what you can: Detecting and representing objects using holistic models and body parts," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1971–1978.

[128] K. Su, X. Liu, and E. Shlizerman, "Predict & cluster: Unsupervised skeleton based action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9631–9640.

[129] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand keypoint detection in single images using multiview bootstrapping," in *CVPR*, 2017.

[130] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *CVPR*, 2016.

[131] G. Ou and Y. L. Murphey, "Multi-class pattern classification using neural networks," *Pattern Recognition*, vol. 40, no. 1, pp. 4–18, 2007.

[132] C. Zhong and A. R. Reibman, "Prediction system for activity recognition with compressed video," *Electronic Imaging*, vol. 2018, no. 2, pp. 254–1, 2018.

[133] T. M. Team, *Mencoder*, https://http://www.mplayerhq.hu/design7/news.html, 2000-2009.

[134] J. Uijlings, I. C. Duta, E. Sangineto, and N. Sebe, "Video classification with densely extracted hog/hof/mbh features: An evaluation of the accuracy/computational efficiency trade-off," *International Journal of Multimedia Information Retrieval*, vol. 4, no. 1, pp. 33–44, 2015.

[135] J. R. Uijlings, A. W. Smeulders, and R. J. Scha, “Real-time visual concept classification,” *IEEE Transactions on Multimedia*, vol. 12, no. 7, pp. 665–681, 2010.

[136] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[137] L. Breiman, “Random forests,” *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[138] C. Chen, A. Liaw, L. Breiman, *et al.*, “Using random forest to learn imbalanced data,” *University of California, Berkeley*, vol. 110, no. 1-12, p. 24, 2004.