1. **Overview**

   Due to the recent progress of cost-effective depth sensors, many tasks in smart the hospital have been automated through AI-assisted solutions. One of these tasks, automating hand-hygiene compliance, can be used to prevent hospital acquired infections. In this paper, convolutional networks (CNNs) are used to classify a top-view depth image as “using dispenser” or “not using dispenser.” One model is trained per sensor.

2. **Related Work**

   Yeung et al. [1] used CNNs to detect if a dispenser was used. To compare their proposed approach with another model, they developed a pose estimation model on RGB-based images to provide information on the person performing the action. They first segment and detect humans in each frame using a background subtraction-based method, and then detect the hand of each human using a CNN-based hand detector trained on a large hands dataset. The pose-estimation model performed better when CNN is fed the entire image. However, the CNN-based classifier performed better than the pose estimate model when the CNN was fed in cropped regions contained the dispenser.

   Haque et al [2] built upon previous work on automating hand-dispenser detection through depth images by focusing on tracking individuals. For each individual captured in the depth map, the model classified whether or not the individual used the dispenser or not. The model combines many computer vision techniques, such as pedestrian detection, tracking across cameras, and hand hygiene activity classification.

3. **Data**

   During the preprocessing, images are transformed to highlight the important regions, such as the dispenser and the individuals near the container. As a result, the first step in the preprocessing stage is to reduce the noise in the images. In the case of depth images, noise usually come in the form of pixels having a value of zero. In depth images, pixels of value of zero are interpreted as being close to the camera. These virtual highly-elevated (relative to the ground) objects will distract the model from paying attention to important, real objects that are close to the camera relative to the floor, such as a person or a dispenser. To get rid of these virtual objects, all pixels with a value of zero are replaced by the average pixel value of the entire image. In addition, 4X4 median filter is used to further remove an pixels with a value of zero.

4. **Methodology**

   Inspired by the work of Yeung et al. [1] and Haque [2], the method used to classify a depth as “using dispenser” or “not using” can be broken into steps:

   a. Background/Foreground Segmentation Approximation:

      Basic thresholding is used for background segmentation. There are two types that Python package library, skimage, uses to implement thresholding: histogram-based and local.

      In histogram-based methods, pixels with similar intensities are bucketed, creating a distribution of intensities. This distribution is used to create a binary mask that will attempt to segment the background from the foreground.

      Local methods classify an individual pixel by looking at its neighboring pixels’ intensities. As a result, local methods tend to require more computation time than histogram-based methods.

      One type of histogram-based method is called Otus’s thresholding. Otus’s method involves calculating an optimal threshold by maximizing the variance between two classes of pixels [3]. As a result, the method works well with an image that is dominated by two peaks of pixel intensities in a histogram of pixel intensities.

      Because the depth images that are collected are top-view, there will be two classes of intensities, where one class will be representing those pixels that are close to the floor and the other class will be representing those pixels that are closer to the sensor, namely the individual and the dispenser. As a result, otsu thresholding is used to mask out the floor and highlight objects that are at higher elevations, such as a person and the dispenser. You can see the final processed images in the file, viewing_processed_images.ipynb.

   b. CNN-Hand Hygiene Detection:

      Three convolutional layers are used, where each layer is followed immediately by a pooling layer. The output of the layer is flattened to be feed into a one layer feed forward network. The output of the feed forward network is a binary classification of whether the hand hygiene action is occurring, and we optimize a logistic loss function using stochastic gradient descent.

5. Results

Sensor	Accuracy on Dev Set
02	0.8233173076923077
04	0.9622641509433962
06	0.7219827586206896
08	0.8968023255813954
10*	No one_labeled images
11	0.8646449704142012
15	0.967032967032967
21*	No one_labeled images
22*	No one_labeled_images
23	0.8520220588235294
24*	No one_labeled_images
39	0.9085526315789474
52	0.7809244791666666
59	0.860625
62	0.8016666666666666
63	0.49270833333333336
72	0.647159090909091

6. Discussion

During training, when the model only looks at one type of images, and then, when all images of particular image is exhausted, the model is fed the other type of images, the model will only accurately predict the labels of the class that it most recently seen. As a result, when feeding the data into the model, there are two paths, one path contains all the filenames of the zero-labeled images and the other path contains all the filenames of the one-labeled images. When preparing the dataset, I weave one class of images with the other class (ie [image1_zero, image2_one, image3_zero, and so one]. This resulted in better classification accuracies for all models.

7. Future Work

Optical flow can be used to segment moving pixels from static pixels by analyzing consecutive frames in the video. In the data, for almost all the sensors, there batches of images that were taken one immediately after the other, as shown in the timestamp that is used to name corresponding image's npz file. Optical flow can be applied to these group of images to detect the objects that are moving within a single batch of images. This information may help to track individuals and determine which individuals are doing the action of using the dispenser. In addition, I will produce images where everything is masked (pixel value equal to zero) except for the objects that are moving added these images to my training dataset.

8. Hardware Specifications

The hardware specifications of the machine used to train the models are the following: a) MacBookPro (Retina, 13-inch, Early 2015), b) Processor: 3.1 GHz Intel Core i7 c) Memory: 16 GB 1867 MHz DDR3

9. References

[1] Yeung et al. Vision-Based Hand Hygiene Monitoring in Hospitals, 2016.

[2] Haque et al. Towards Vision-Based Smart Hospitals: A system for Tracking and Monitoring Hand Hygiene Compiance, 2017.

[3] Dhawan et al. Implementation of Hand Detection based Techniques for Human Computer Interaction, 2013