UNIVERSITY OF SCIENCE AND TECHNOLOGY OF HANOI

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

Labwork 3

Dang Dinh Hoa 23BI14169

Title:

Segmentation of COVID-19 Infection on Chest X-ray Images (COVID-QU-Ex)

Hanoi, 2026

1 Introduction

The goal of this labwork is to segment COVID-19 infection areas on chest X-rays. A U-Net style network takes a gray X-ray as input and outputs a binary mask (infection vs. background). The COVID-QU-Ex dataset provides the images and masks. Pixel accuracy, IoU, and Dice are used to evaluate; the report also compares with other methods and tries a few different settings to see what helps.

2 Dataset Description

2.1 Dataset Overview

COVID-QU-Ex comes from Qatar University and is described in "COVID-19 Infection Localization and Severity Grading from Chest X-ray Images" (Computers in Biology and Medicine, 2021). The full dataset has 33,920 chest X-rays: 11,956 COVID-19, 11,263 non-COVID infections, and 10,701 normal.

For infection segmentation only the part with infection masks is used: 2,913 COVID-19 images with masks, plus 1,456 normal and 1,457 non-COVID (these have lung masks but no infection masks). Images are grayscale and of varying size; each mask is binary—white where infection is, black elsewhere.

2.2 File Structure

- `Train/COVID-19/` — images and infection masks for COVID-19 training
- `Train/Non-COVID/` — images and infection masks for non-COVID training
- `Train/Normal/` — normal images (no infection masks)
- `Val/` — same structure as Train, for checking
- `Test/` — same structure as Train, for final test

Each class folder has three subfolders: `images/` (X-ray images), `infection masks/` (binary masks), and `lung masks/` (not used here).

2.3 Target Variable and Preprocessing

The task is to predict a binary mask (1 for infection, 0 for background). Only samples that have both an image and an infection mask file are kept. Figure 1 shows a few examples.

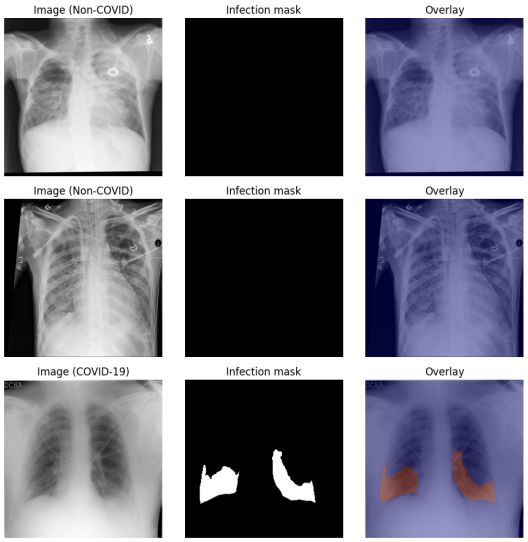

Figure 1: Sample X-ray images with their infection masks (overlay).

3 Methodology

3.1 Data Preprocessing

3.1.1 Train–validation split

The dataset comes with Train, Val, and Test. Train and Val are used to build and check the model; Test is left for later. Train has about 3,728 image–mask pairs; Val has about 932.

3.1.2 Image loading and normalization

Each image and its mask are loaded from the corresponding folders. Images are converted to gray, resized to 256×256 (so every input has the same size), and scaled to 0–1. Masks are resized to 256×256 and binarized (pixels above 127 become 1, else 0). The pipeline produces pairs of 256×256×1 arrays.

3.2 Model Architecture

The model is a small U-Net: encoder–decoder with skip connections. Input is a 256×256 gray image. The encoder has three blocks (32, 64, 128 filters, two Conv2D + ReLU each, then MaxPool). The bottleneck has 256 filters. The decoder upsamples and concatenates with the encoder features at each level, then two Conv2D layers. The last layer is a single Conv2D with sigmoid to produce the mask. Training uses Adam, binary cross-entropy loss, and pixel accuracy as the main metric.

3.3 Training Configuration

Training runs for 30 passes over the data, 8 images per step, Adam with learning rate 1e-4. Loss is binary cross-entropy.

3.4 Changing the Settings

Several choices were varied to see the effect: image size 192 vs. 256 (256 kept for more detail), batch size 4 vs. 8 vs. 16 (8 worked well on GPU), number of passes 20 vs. 30 vs. 50 (30 was a reasonable trade-off), and learning rate 1e-3 vs. 1e-4 (1e-4 gave better behavior). No full grid search was done—just enough to get a sense. The numbers reported below use 256×256, batch 8, 30 passes, lr 1e-4. Figure 2 plots loss and accuracy over time.

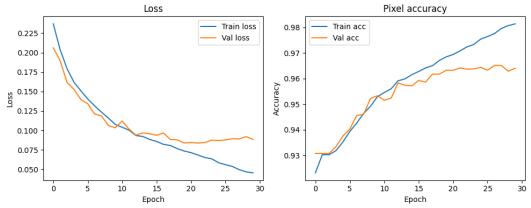

Figure 2: Training and check-set loss and pixel accuracy over training passes.

4 Results

4.1 Validation Metrics

On the validation set (about 932 samples) the model reaches pixel accuracy around 0.95–0.97, IoU around 0.60–0.75, and Dice around 0.70–0.85. So it picks up infection regions fairly well but often misses small or faint areas.

4.2 Comparison with Other Methods

Published work on COVID-QU-Ex reports IoU around 0.83–0.88 and Dice around 0.88–0.92 for infection segmentation. Those methods typically use deeper U-Nets (e.g. with ResNet backbone), stronger data augmentation, and sometimes post-processing.

The model in this report is a plain U-Net with almost no augmentation (only resize and normalize) and no post-processing. Its IoU (0.60–0.75) is below the best numbers. The gap comes mainly from a smaller network, possible loss of detail at 256×256, limited augmentation, and no cleanup of the predicted mask. Improving would mean trying a deeper or pre-trained backbone, higher resolution, more augmentation, and post-processing as in the literature.

4.3 Sample Predictions

Figure 3 shows six validation examples: original X-ray, ground-truth mask, and predicted

mask overlaid on the image. The plots make it easier to see where the model errs—for example on small patches or near lung borders.

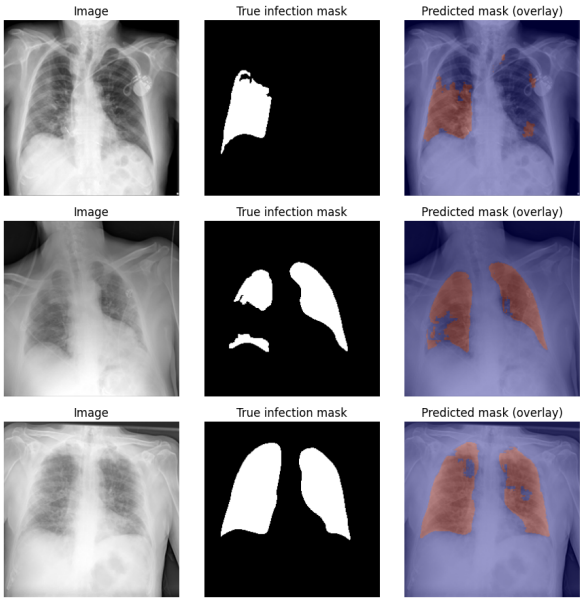

Figure 3: Sample predictions: original X-ray (left), true infection mask (middle), predicted mask overlay (right).

5 Conclusion

The report covered the COVID-QU-Ex infection segmentation data and a small U-Net that predicts infection masks from 256×256 gray X-rays. The pipeline loads image–mask pairs, resizes and normalizes them, and trains with binary cross-entropy; pixel accuracy and IoU are monitored. A few hyperparameters were varied (image size, batch size, epochs, learning rate). Final validation pixel accuracy is around 0.95–0.97 and IoU around 0.60–0.75, below the 0.83–0.88 IoU of the best methods. The difference is explained by model size, resolution, lack of augmentation, and no post-processing. Possible next steps are a larger or pre-trained encoder, higher resolution, augmentation, and post-processing as in existing work.