2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

# Recovering Missed Detections in an Elevator Button Segmentation Task

Nicholas Verzic[1], Abhinav Chadaga[1], and Justin Hart[1]

***Abstract*— One obstacle that mobile service robots face is operating elevators. Reading elevator control panel buttons involves both an instance segmentation of buttons and labels and associating buttons with their respective metal labels in the elevator. Segmentation algorithms, however, can miss detections. This paper presents a segmentation model specifically designed to solve the problem of missed detections. This can be used to recover detections that the initial model misses. This work presents: 1) a new elevator button dataset containing both 108 images sampled from the internet and 292 images imaged from 24 buildings from the University of Texas at Austin campus and the surrounding neighborhood, along with their segmentation boundaries and associated labels; 2) a vision pipeline based on Mask-RCNN for solving the initial image segmentation and labeling task; and 3) a novel method for identifying missed detections, using a Mask-RCNN network trained on expected button locations. Results show that the missed detections model, specifically developed to recover buttons and labels that were missed by the initial pass, is accurate on up to $99.33\%$ of its predicted missed features on a synthetic missed-detection dataset and $97.14\%$ of its predictions for features missed on a non-synthetic dataset. In the case of the average accuracy of successful button and label detections of a specifically-trained "weak" initial detector at a standard IoU threshold of $0.5$, the missed detection model improves the detector's success rate from $80.38\%$ on the button recognition task with the initial segmentation model only to an average accuracy of $90.7\%$ with the missed detections model enabled. The overall accuracy of the best-performing pipeline implementing the missed detections model is $91.73\%$ and $98.27\%$ on our Internet subset and Campus subset of our dataset, respectively.**

## I. INTRODUCTION

The Living With Robots Laboratory develops service robots that operate in campus buildings at The University of Texas at Austin [1]. We would like for these robots to operate elevators without having to modify the elevator hardware. Part of this task is to enable the robot to read elevator button panels, as seen in Figure 1. The task of reading an elevator button panel can be decomposed into several subtasks. These are button and label detection, optical character recognition, and button-label association. Like other neural networks, instance segmentation networks can suffer from missed detections. A unique contribution of this paper is an additional model used to detect when the segmentation network has missed a detection.

In the United States, the design of elevator button panels is regulated by the Americans with Disabilities Act (ADA).[1] The ADA states that elevator buttons must be accompanied by separate labels that are placed to their left.[2] These labels should have either Braille[3] writing or icons that can be read by touch by people with visual impairments. Though not stated in the law, placing the labels in this fashion prevents the user from accidentally pressing a button reading by touch.

This layout of elevator buttons on ADA-compliant elevators can be leveraged for two important purposes in a computer vision pipeline. The first purpose is that the association between a button label and the corresponding button can be inferred by looking for the closest button to the right of a label. The associated label can be used to identify the function of the button; such as going to a floor or opening and closing the door. The second purpose is that this leads to a characteristic layout of elevator buttons on the panel. This layout can be used to infer where buttons that are not detected by the initial detection pass may be found. A false negative in the vision system may lead to problems finding elevator buttons. False negatives may be caused by wear on elevators, poor lighting conditions, or other factors. This paper presents a neural network based on Mask-RCNN [2] which is used to infer the positions of these missed buttons, enhancing system performance.

The presented system works as follows. The elevator button panel is segmented into buttons and labels using a Mask-RCNN [2] network implemented in Detectron2 [3]. Two versions of this network are implemented for use with this system: one based on the original Mask-RCNN using a ResNet101 backbone [4]; and the other based on an enhancement made by Li et al. [5], which proposes a modification to the feature pyramid network of Mask-RCNN that allows a plain, unmodified, vision transformer (ViT) [6] to be used as the backbone. After initial detections are made, the system runs a second network which is used to detect missing buttons based on the output of the first network. This second network is one of this paper's unique contributions. Buttons detected by either the initial button detection network or the missed button detection network are combined into a final set of detections. Scene Text Recognition (STR) using PARSeq [7] is then run over the image patches from the segments containing elevator labels to identify the floor or function corresponding to each label. Finally, buttons and labels are associated with each other by pairing each label with the button to its right.

We also present a dataset of $400$ images; including a subset sampled from internet searches and another subset

Department of [1]Computer Science, The University of Texas at Austin, 2501 Wichita St, Austin, TX 78712-1757. {nicholasverzic, abhinav.chadaga}@utexas.edu, hart@cs.utexas.edu

[1]Title 42 of the US Code.

[2]https://www.ada-compliance.com/ada-compliance/ada-elevators.html

[3]A writing system for people with vision impairments. Symbols are represented by sets of raised bumps, which can be read by touch.

photographed inside elevators in buildings on the UT Austin Campus and in the nearby neighborhoods. The images have been carefully annotated and are used to both train and test the system presented in this paper.

The unique contributions in this paper are: 1) A system for semantically labeling elevator button panels; 2) An approach to detecting segments missed by the initial segmentation network through a neural network used to detect missed detections; and 3) The Living With Robots Elevator Button Dataset, containing images of elevator button panels and their corresponding segments and semantic label. The presented computer vision pipeline is tested and compared against the current state-of-the-art system for this task [8]–[10].

## II. RELATED WORK

Operating elevators is an important problem for mobile robot navigation, as elevator operation enables robots to traverse between multiple floors in a building. Following human operators into elevators was a part of the 2014 RoboCup@Home rules [11]. Rosenthal et al. [12] enabled the CMU CoBot to navigate elevators by asking people using the elevator for assistance. Another system, GRACE, performed this task by asking people to operate an elevator as part of the AAAI Mobile Robot Competition [13]. The Savioke Relay+ has a proprietary button operation module that was introduced in 2021.[4]

Klingbeil et al. [14] present a system to read elevator buttons using the STAIR vision library [15] to generate bounding boxes. Similar to the system in this paper, their paper implements a method to recover missed detections. Their detector leverages the fact that elevator controls are typically laid out in a grid to implement an Expectation Maximization algorithm over possible button locations. Liu et al. [16] present a system that tests several semantic segmentation networks to identify buttons. The loss function for training each of the networks is designed to put a special emphasis on missed detections. This improves the performance of their detection network, but — unlike our system — is not designed to recover missed detections after the initial detection pass.

Semantic segmentation labels all pixels in an image corresponding to a class. Instance segmentation, generates separate pixel-level masks for each instance of an object. Long et al. [17] propose a Fully Convolutional Network (FCN) to solve the problem of semantic segmentation. FCNs replace the last few fully connected layers of a CNN with upsampling layers and 1x1 convolutional layers to classify each pixel. Mask-RCNN [2] is a framework for instance segmentation. A recent enhancement on Mask-RCNN is the incorporation of a ViT [6] into the pipeline [5]. This replaces the ResNet101 convolutional neural network, which was previously used as the feature extractor or "backbone." Initial versions of the vision transformer necessitated pre-training on large amounts of labeled data like in ImageNet [18]. However, He et al. [19] propose a self-supervised pre-training method for vision transformers using masked autoencoders (MAE). Their routine involves training a transformer encoder to reconstruct images from which a large set of patches were removed. He et al. [19] demonstrate stronger representation learning when compared to traditional supervised learning methods for both ViTs and CNNs based on the results for tasks such as classification.

When tackling the problem of recovering missed detections in instance segmentation, two approaches have been previously presented. Wang et al. [20] address the task of minimizing missed segmentation instances by training two adversarial networks, one specialized in recognizing false negatives and one designed to identify false positives. Both networks are applied over the image to be segmented before the resulting segments are averaged together to determine appropriate detections. The authors apply this to small object segmentation in infrared footage. By contrast, the method presented in this paper works over plain-color images and uses a the newer method of a Mask-RCNN network with a ViT backbone. Another approach to recovering missed detections is presented by Wang et al. [21] for use in autonomous vehicles. Compiling all the information on an initial network's missed detections with Global Pyramid Networks, their system utilizes both semantic segmentation and query-based instance segmentation to propose and confirm potential missed detections. This works in their case because it enhances detections over the underlying heatmap; whereas the approach in the present work is tested in conditions where the feature is entirely missed and may not be present at all in the heatmap output from the initial segmentation network.

Once buttons and labels have been segmented, the symbol on the label must be read. Zhu et al. [8] treat reading elevator button labels as an OCR task rather than a classification task. They use a Recurrent Neural Network with attention masks to serially process the tokens in an elevator button label, implemented as a branch of Faster-RCNN [22]. Our system uses PARSeq [7], a ViT-based approach that treats the problem as one of Scene Text Recognition, or interpreting text in natural settings.

Training and testing these systems requires datasets of elevator button panels. Klingbeil et al. [14] collected 150 different images of elevators from 60 distinct elevators featuring 686 buttons. Because their work is based in the United States, their dataset also follows ADA guidelines. Unfortunately, their dataset is not publicly available. Yang et al. [23] collected a dataset of 260,560 images to perform their research, though their dataset is not publicly available. Liu et al. [16] published a much larger — publicly available — dataset of 3,718 images with 35,100 buttons, however many of the elevators in this dataset are not ADA compliant. To support research on this problem, our dataset of 400 elevator button panel images is publicly available on the Texas Robotics Dataverse [24].

## III. ELEVATOR BUTTON DATASET

The Living With Robots Elevator Button Dataset — detailed in Table I — contains a total of 400 images of

[4]https://www.relayrobotics.com/

elevator button panels, all adhering to ADA guidelines. It contains two subsets of images; with 108 sampled from the internet and 292 photographed in 24 buildings around the UT Austin campus. Each image has been carefully annotated at the pixel-level and labeled using the VGG Image Annotation tool [25], [26] with two classes: "button" and "label." Segments are also labeled with either floor numbers or symbols representing operations like "door open."

### A. Campus Sub-dataset

The Campus sub-dataset comprises 292 images taken of 25 different elevators across 24 buildings. These images are taken in buildings on and around The University of Texas at Austin campus. All pictures are taken facing the elevator panel's wall roughly straight-on, while the camera itself is positioned in each of nine locations in a 3x3 grid layout relative to the panel: to the top left, top middle, top right, middle left, center, middle right, bottom left, bottom middle, and bottom right. A subset of these also includes versions of each image with the elevator door closed or open, varying the lighting conditions. Examples of images from the Campus sub-dataset can be seen in Figure 1.

### B. Internet Sub-dataset

The Internet sub-dataset includes 108 images sourced from the Internet; featuring user-shared photos with irregular or uncommon panel characteristics. Images in this dataset vary widely in terms of resolution, clarity, button/label shape, and angle of the image. Examples of images from the Internet sub-dataset can be seen in Figure 1.

| Sub-Dataset | Train | Val. | Test | Total |
|---|---|---|---|---|
| Campus | 192 | 12 | 88 | 292 |
| Internet | 76 | 10 | 22 | 108 |
| Combined | 268 | 22 | 110 | 400 |

TABLE I
BREAKDOWN OF THE LIVING WITH ROBOTS ELEVATOR BUTTON DATASET, INCLUDING TRAIN/VALIDATION/TEST SPLITS USED IN THIS PAPER.

## IV. MISSED DETECTIONS MODEL IN A BUTTON AND LABEL DETECTION PIPELINE

A network performing image segmentation on an elevator button panel can miss buttons or labels. To remediate this, this work presents a missed detections model that is trained to fill in button and label detections missed by the first network. Its input is the class map output from initial segmentation network. It outputs a new map that may contain previously unidentified segments to be appended the original detections. The network is trained by taking the class labels from the training data and removing them — one at a time — from the input, while retaining them in the output. Because this network has no particular requirements other than the segments returned by the initial segmentation model, it could be added to any elevator button detection pipeline.

The entire system for identifying, segmenting, labeling, and reading elevator buttons works as follows. The system input is an RGB image of an elevator panel. First, the system performs a segmentation pass of the elevator buttons and labels using a Mask-RCNN network using either a ResNet101 or ViT backbone. The class map output by the detection network is input to the missed detections network, which outputs labels and buttons that were undetected by the first network. The two class maps are then added together; combining the detections of the first and second networks. PARSeq is then used to read labels next to the buttons, and buttons are associated with their corresponding labels by associating a label to the nearest button to its right. An overview of the system can be seen in Figure 2.

### A. Initial Button and Label Detection

The buttons and labels on elevator button panels generally have different appearances. This system treats button and label detection as a two-class instance segmentation problem. This work uses the popular Mask-RCNN architecture. Systems are trained and tested using both ResNet101 and ViT backbones.

He et al. [19] propose an effective pre-training process for ViT in which a ViT encoder is trained as an autoencoder to recreate the masked-out portions of images. They find this pre-training process to be more effective than traditional supervised pre-training methods for both ViT and CNN-based models. Thus, to train the ViT-based Mask-RCNN models, the fine-tuning is initialized from these higher-quality pre-trained weights. The ResNet101-based model is initialized from pre-trained weights on ImageNet. All configurations are fine-tuned for 50 epochs with an initial learning rate of 0.00025 on a training set sampled from the Living With Robots Elevator Dataset.

Vision Transformers work very well on the segmentation task presented in this paper. To demonstrate the effectiveness of the missed detections solution, we present a less accurate ViT segmentation model (denoted as "Weak" in the evaluation). This less-accurate model misses more detections, and therefore gives a better demonstration of the missed detection network's performance. The weaker ViT segmentation model is trained on only one-third of the dataset.

All ViT-based models presented in this work are trained using the AdamW optimizer. The ResNet101-based model is trained using SGD with a momentum of 0.9. Models are evaluated every two epochs on the combined validation set and the model producing the lowest validation loss is selected for the final pipeline. All models are trained on an Nvidia A100 system with Xeon Gold 6342 CPUs and 512 terabytes of RAM.

### B. Missed Button and Label Recovery Detection

To train the missed button and label detection network, the system generates random permutations of class maps with missing buttons or labels from each elevator panel in the dataset. A base class map sets pixels that belong to the label class to [0, 255, 0] and button class to [0, 0, 255].

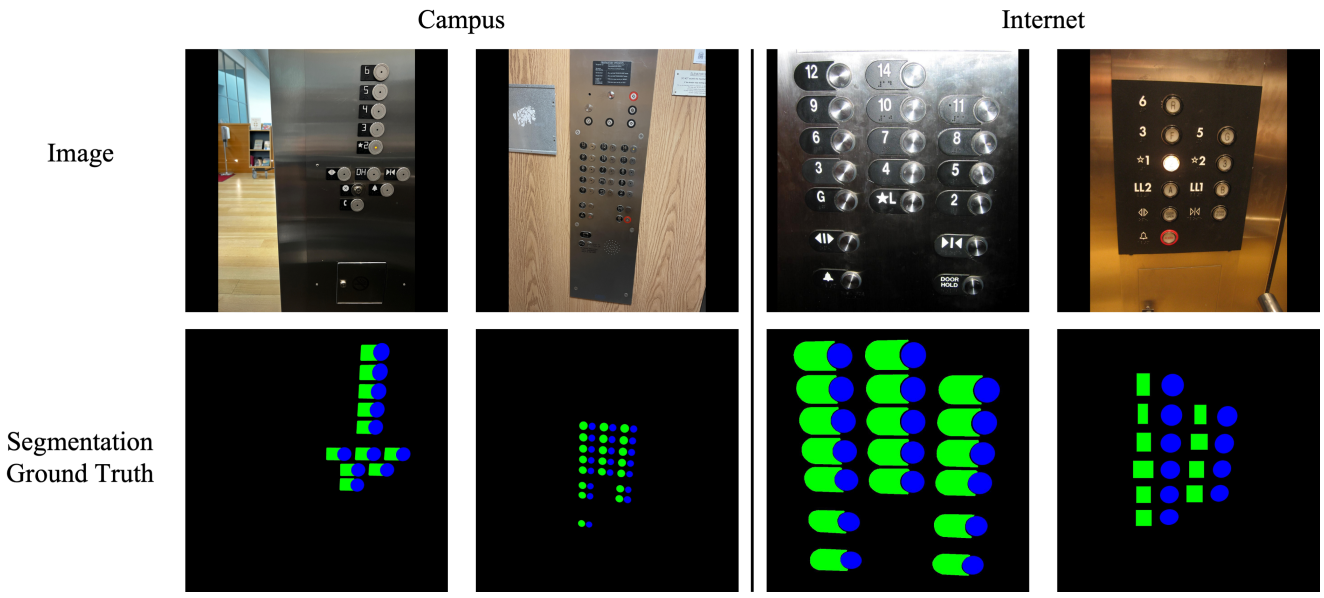


Fig. 1. Example images and segmentation ground truth masks from the Living With Robots Elevator Button Dataset.

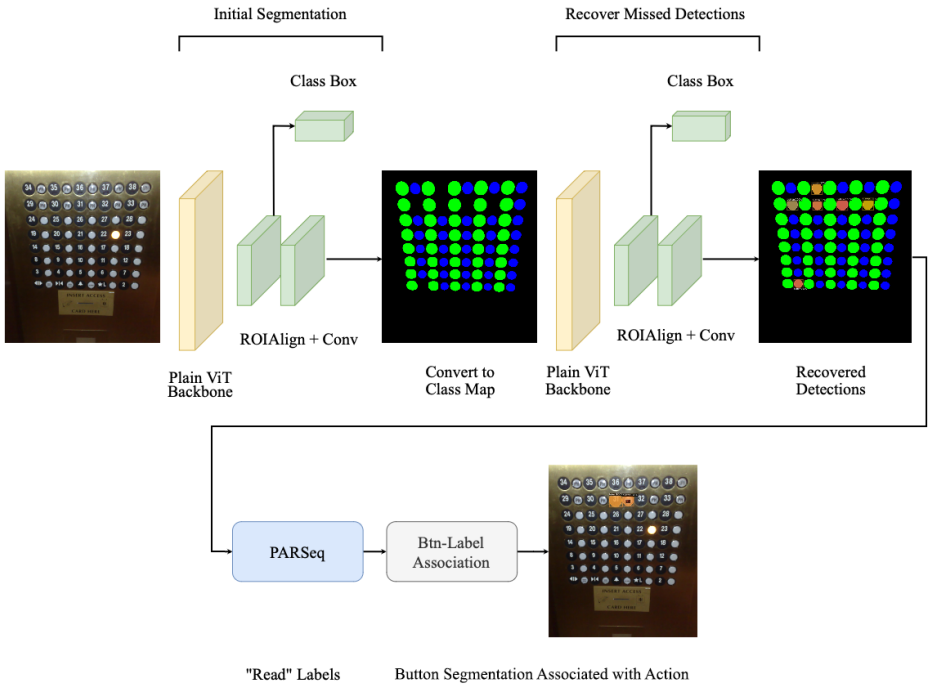


Fig. 2. An overview of the button and label detection system.

An example class map is shown in Figure 1. By generating class maps as 3D RGB tensors, the system is able to start from pre-trained weights. Permutations of missing features from this base class map are generated for each image using Algorithm 1.

Ground truth labels for the dataset generated by Algorithm 1 are the masks of the missing buttons and labels for each image. The system prevents buttons and labels of the same pair from being dropped out together, while assuring that every button-label pair has the mask of either its button or label dropped at least once. A base class map with no missing features is also included in testing.

Training follows the same procedure as the initial segmentation task by training two models - one with a ViT backbone and one with the ResNet101 backbone. Models are trained for 50 epochs using the AdamW optimizer and SGD with a momentum of 0.9 respectively. Both models are trained with a base learning rate of 3e-5.

**Algorithm 1** Training algorithm for the missed button and label detection network.

```
N ← Number of button-label pairs
for k ∈ {1, ··· , ⌈0.4 ∗ N⌉} do
    P ← {all button-label pairs}
    C ← ∅
    while |P| > 0 do
        if k ≤ |P| then
            Sample and remove k pairs from P
            Add selected pairs to C
        else
            Randomly sample k − |P| pairs from C
            Combine with remaining pairs in P
        end if
        for p ∈ C do
            Randomly select either button or label from pair
            Set selected pixels in base class map to 0
        end for
        Save image
    end while
end for
```

Note: Sampling is done randomly and without replacement

### C. Improved Button and Label Detection

The output of the initial detection model and the "recovered" buttons and labels from the missing button and label detection model are combined by adding the two sets of detections to yield the full set of button and label detections. This combined set of detections is the final output for button and label detection.

### D. Optical Character Recognition

This system employs PARSeq [7], a state-of-the-art Scene Text Recognition model, which has been fine-tuned with the Living With Robots Elevator Button Dataset over image crops that belong to the "label" class. These images are reshaped into 64 x 64 images before being used to retrain the model. Labels featuring non-alphanumeric symbols such as the bell for "alarm" are mapped to their own characters in the model. Labels containing strings of text, like "CALL" instead of the phone symbol, are simply read as-is.

### E. Button / Label Association

Elevators in the United States typically follow ADA guidelines. The system utilizes the fact that labels should be located to the left of their accompanying button to associate the two together. For each label, the system locates the button that is closest to its right. In the event that no such button to the right of the target label exists, the system chooses the globally closest button; which is typically correct.

## V. Evaluation

The missed detections model and the entire pipeline are evaluated over three versions of the Living With Robots Elevator Button dataset: Campus, Internet, and Combined (which combines the prior two). The Combined dataset contains a total of 400 images divided into Campus and Internet sub-datasets, each of which are further split into training, validation, and testing sets as described in Table I. The system is trained and validated on the combined version and evaluated on the test portion of each version since the Campus and Internet subsets have different characteristics. The PARSeq model is trained on 3258 samples, validated during training on 343 samples, and tested on 1430 samples. These splits correspond to the image sets for the overall system.

Several metrics are used to evaluate the system's performance.

### A. Intersection Over Union

Intersection Over Union (IoU) quantifies the overlap between ground truth features and predicted features, evaluating the quality of a segmentation mask. It is calculated for a ground truth and predicted feature pair as follows:

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{1}$$

### B. Precision

Precision (or true positive rate) is the rate at which predictions are accurate. It measures the portion of predictions that are considered accurate by IoU. It is calculated as:

$$Precision = \frac{\text{No. True Positives}}{\text{No. True Positives + No. False Negatives}} \tag{2}$$

### C. Average Accuracy

Average accuracy measures the proportion of true positive detections to total detections. It is defined as follows:

$$AAC = \frac{\text{No. correctly identified features}}{\text{Total no. of features}} \tag{3}$$

The system presented in this paper performs semantic instance segmentation to classify buttons and labels. The existing state-of-the-art implementation of an elevator panel feature detector is an OCR-RCNN methodology proposed by [8] which both does not discriminate between buttons and labels and provides only bounding boxes around compact features. Thus, the detector presented in this paper preforms a more difficult task. To compare the effectiveness of the present system and OCR-RCNN via a metric both can measure, the evaluation presents recognition performance using average accuracy. Because what constitutes a positive sample is entirely dependent on the IoU threshold, the systems are evaluated and compared at progressively increasing IoU thresholds. Doing so allows for a comparison of the tightness of the bounding boxes between the two systems. This evaluation counts both labels and buttons. OCR-RCNN does not discriminate between the two classes. When calculating the total number of segments correctly identified by the present system, the evaluation does not count misclassifications by our system that meet the IoU threshold with a ground truth

feature as a positive sample. That is to say, if our system classifies a "button" as a "label" or vice-versa, it is counted as as inaccurate; even though it would be counted as accurate in the OCR-RCNN output (which is incapable of distinguishing between the two). Consequently, the evaluation holds the system presented in this paper to a higher standard than OCR-RCNN. Since OCR-RCNN only returns bounding boxes, when comparing IoU, the comparison is between the ground truth bounding box and the predicted bounding box (even though our system returns segments). The evaluation uses the pre-trained model from OCR-RCNN's github[5], and it is worth noting that that model was trained on far more images than the present system. All things considered, in order to facilitate a comparison between the two systems on the same metrics, the one presented in this paper is completing a far more difficult task than the competing state-of-the-art pipeline.

### D. Average Precision

Average Precision is the primary metric used in the COCO challenge. Average Precision is the area under the precision-recall curve. The pixel-level quality of our segmentation tasks was evaluated using this evaluation criterion at a fixed IoU threshold of 0.5.

### E. OCR Accuracy

To evaluate the quality of our OCR/STR model, the evaluation simply compares the predicted values for labels in the combined test set of 1430 samples to the ground truth labels defined in the annotations. This is calculated as follows:

$$\text{OCR Accuracy} = \frac{\text{No. labels correctly "read"}}{\text{Total no. of labels}} \tag{4}$$

### F. Overall System Performance

To evaluate the overall performance of the system, the evaluation verifies whether the correct button is matched to the correct label, not counting it if the button is missed entirely. The accuracy measurement is chosen such that positive selection is determined by the IoU between the predicted selection and the ground truth selection. For these purposes, the IoU threshold is set at 0.5. Thus, the metric is calculated as follows:

$$\text{System Performance} = \frac{\text{No. of correct button selections}}{\text{No. of possible button selections}} \tag{5}$$

## VI. Results

This evaluation presents a comparison between several different detection systems. The system presented in [8] is denoted as "OCR-RCNN" in Tables IV, V, VI. ViT refers to the use of the Plain Vision Transformer Backbone in both the initial segmentation model and missed detections model, with "Weak" prefacing it denoting the deliberately weaker initial segmentation model. Versions of both the initial segmentation model and missed detections model were also trained using ResNet101 as the backbone, denoted "ResNet" in the tables, demonstrating the similar improvement the proposed recovery approach can make when CNN-based. A subscript of "nmd" refers to the absence of the missed detections model in the system, while "md" refers to its presence.

Tables II and III display the high precision of the missed detections model. A dataset was constructed which synthetically "misses" segments (per Algorithm 1) includes 5037 missed segments in the Internet dataset and 5360 missed segments in the Campus dataset. This simulates the initial detector missing detections so that the missed detection network's precision may be evaluated. These results are displayed in Table II. A second test dataset organically misses image segments by running the "Weak" ViT segmentation model. The missed detections model includes 102 missing segments in the Internet dataset and 3 in the Campus dataset. The missed detections model accurately makes predictions in a variety of missed detection circumstances while introduces very few false positives with precision measurements such as $99.33\%$ and $97.14\%$ at the commonly-used IoU threshold of 0.5 in artificial and organic misses, respectively.

Tables IV shows that the system presented in this paper achieves very high performance on the segmentation task for the campus dataset; achieving $99.8\%$ average accuracy on the Campus sub-dataset at an IoU of 0.5 for both the no-missed-detections model and the missed-detections model. OCR-RCNN achieves only $78.13\%$ on this task. Moreover, this performance is maintained at an IoU of 0.8, where OCR-RCNN drops off entirely, achieving only $0.16\%$.

Table V demonstrates the efficacy of the missed detections network. The average accuracy on the weak initial detector is $78.64\%$ at an IoU of 0.7, whereas adding the missed detections model to this detector boosts performance to $86.84\%$. It also boosts the already high scores achieved by the best-performing ViT model from $95.51\%$ to $96.69\%$ at an IoU of 0.7. Performance improvess across all IoU thresholds, as the systems utilizing missing button and label detection (subscripted "md") consistently outperform the ones without missing button and label detection (subscripted "nmd").

Table VI is an evaluation over the combined dataset; again exhibiting a substantial increase in average accuracy due to the missed detections recovery. The missed-detections network improves the performance of the "Weak" ViT from $83.46\%$ to $86.62\%$ at an IoU threshold of 0.5. Also across the combined dataset, the best performing system is presented Plain Vision Transformer Backbone with the missed detections model included, at all IoU thresholds.

| Dataset | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|---|---|---|---|---|---|
| Internet | 99.86% | 98.75% | 96.33% | 86.80% | 35.46% |
| Campus | 98.84% | 98.68% | 96.32% | 85.62% | 39.85% |
| Combined | 99.33% | 98.71% | 96.32% | 86.19% | 37.72% |

TABLE II
PRECISION OF MISSED DETECTIONS MODEL ON SYNTHETICALLY-MISSED FEATURES BY IOU THRESHOLD.

[5]https://github.com/zhudelong/ocr-rcnn-v2

| Dataset | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|---|---|---|---|---|---|
| Internet | 97.06% | 90.20% | 84.31% | 61.76% | 26.47% |
| Campus | 100.00% | 100.00% | 100.00% | 33.33% | 33.33% |
| Combined | 97.14% | 90.48% | 84.76% | 60.95% | 26.67% |

TABLE III

PRECISION OF MISSED DETECTIONS MODEL ON ORGANICALLY-MISSED FEATURES BY IOU THRESHOLD.

| System | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|---|---|---|---|---|---|
| OCR-RCNN | 78.13% | 60.08% | 14.67% | 0.16% | 0.00% |
| $ResNet_{nmd}$ | 99.49% | 99.49% | 99.49% | 99.49% | 94.70% |
| $\mathbf{ViT_{nmd}}$ | 99.80% | 99.80% | 99.80% | 99.80% | 93.89% |
| $Weak\_ViT_{nmd}$ | 84.79% | 84.79% | 84.79% | 84.79% | 81.53% |
| $ResNet_{md}$ | 99.49% | 99.49% | 99.49% | 99.49% | 94.70% |
| $\mathbf{ViT_{md}}$ | 99.80% | 99.80% | 99.80% | 99.80% | 93.89% |
| $Weak\_ViT_{md}$ | 84.86% | 84.86% | 84.79% | 84.79% | 81.53% |

TABLE IV

AVERAGE ACCURACY ON CAMPUS SUB-DATASET BY IOU THRESHOLD. BOLDED SYSTEMS ARE THE BEST PERFORMING SYSTEMS.

| System | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|---|---|---|---|---|---|
| OCR-RCNN | 72.34% | 55.51% | 21.78% | 4.19% | 0.00% |
| $ResNet_{nmd}$ | 95.98% | 95.51% | 93.03% | 88.65% | 66.31% |
| $ViT_{nmd}$ | 97.75% | 97.28% | 95.51% | 91.13% | 70.45% |
| $Weak\_ViT_{nmd}$ | 80.38% | 80.06% | 78.64% | 75.41% | 57.53% |
| $ResNet_{md}$ | 96.81% | 96.22% | 93.74% | 89.48% | 67.02% |
| $\mathbf{ViT_{md}}$ | 98.94% | 98.94% | 96.69% | 92.20% | 71.04% |
| $Weak\_ViT_{md}$ | 90.7% | 89.76% | 86.84% | 80.61% | 58.39% |

TABLE V

AVERAGE ACCURACY ON INTERNET SUB-DATASET BY IOU THRESHOLD.

| System | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|---|---|---|---|---|---|
| OCR-RCNN | 76.39% | 58.70% | 16.81% | 1.37% | 0.00% |
| $ResNet_{nmd}$ | 98.43% | 98.29% | 97.55% | 96.23% | 86.15% |
| $ViT_{nmd}$ | 99.18% | 99.04% | 98.51% | 97.19% | 86.83% |
| $Weak\_ViT_{nmd}$ | 83.46% | 83.37% | 82.94% | 81.97% | 74.30% |
| $ResNet_{md}$ | 98.68% | 98.51% | 97.76% | 96.48% | 86.37% |
| $\mathbf{ViT_{md}}$ | 99.54% | 99.54% | 98.86% | 97.51% | 87.01% |
| $Weak\_ViT_{md}$ | 86.62% | 86.34% | 85.41% | 83.53% | 74.56% |

TABLE VI

AVERAGE ACCURACY ON COMBINED DATASET BY IOU THRESHOLD.

The use of the transformer-based PARSeq model for label reading provides the present system with a significant improvement in OCR/STR performance when compared to the RNN-attention model used by Zhu et al. [8]–[10]. OCR-RCNN's system performs with $66.88\%$ accuracy, whereas the fine-tuned PARSeq model performs with $97.48\%$ accuracy.

When looking at the pixel-level quality of segmentations as given by the AP50 metric from Table VII, the evaluation confirms the precise recovery of false negatives by the missed detections model, where the $ViT_{md}$ achieves the best performance (though all performances are quite good). Additionally, the evaluation yet again confirms that the missed detections model does not introduce false positives, as this would significantly harm AP50.

| System | Campus | Internet |
|---|---|---|
| $ResNet_{nmd}$ | 98.88 | 94.67 |
| $ViT_{nmd}$ | 99.01 | 96.93 |
| $ResNet_{md}$ | 98.88 | 95.12 |
| $\mathbf{ViT_{md}}$ | 99.01 | 98.40 |

TABLE VII

AP50 SCORES ON ALL TEST FEATURES.

Finally, the overall pipeline accuracy results when implementing the missed detections model shows a marked improvement in correctly recognizing and selecting the appropriate button for traveling to a given floor when compared to similar evaluations done by Klingbeil et al. [14]. More importantly, Table VIII indicates a very high accuracy on the Campus sub-dataset ($98.27\%$), which is intended to be representative of what a robot engaging in multi-story navigation in an urban environment might encounter. The overall pipeline accuracy for the same system on the Combined dataset is $96.30\%$.

## VII. CONCLUSION

This work presents an instance segmentation model trained to detect and eliminate missed detections in vision systems designed to identify and segment elevator button panels. The model accurately and precisely fills in missed detections using a specially-trained Mask-RCNN network that may be plugged in to any elevator panel feature segmentation system. To test the model's efficacy, an elevator button and label detection pipeline was created that leverages a Mask-RCNN segmentation network in order to identify buttons and labels, applies the missed detections model to recover undetected buttons and labels, reads the labels using PARSeq, and associates labels with the correct buttons, all with high precision. To train and evaluate the system, a dataset has been developed which has been made publicly available on the Texas Robotics Dataverse [24]. Additionally, the source code for this system can be found on the Living With Robots Lab GitHub Organiation.[6] A comparison is made between the present system and a recent state-of-the-art system performing an easier task. It is demonstrated that our system outperforms the current state-of-the-art system. This system will be incorporated into a larger manipulation and navigation pipeline in support of the Living with Robots Laboratory's mobile service robot projects.


## ACKNOWLEDGMENT

This work has taken place in the Living with Robots Laboratory (LWR) at UT Austin. LWR research is supported by NSF Grant #2219236, Cisco Systems, Army Futures Command, and the University of Texas at Austin Bridging Barriers: Good Systems program.


[6]https://github.com/Living-With-Robots-Lab/ElevatorPanelFeatureDetection

| | Campus | | | Internet | | | Combined | | |
|---|---|---|---|---|---|---|---|---|---|
| System | Identified | Total | Accuracy | Identified | Total | Accuracy | Identified | Total | Accuracy |
| ResNet | 962 | 982 | 97.96% | 382 | 423 | 90.31% | 1344 | 1405 | 95.66% |
| **ViT** | 965 | 982 | 98.27% | 388 | 423 | 91.73% | 1353 | 1405 | 96.30% |

TABLE VIII

OVERALL PIPELINE ACCURACY.

## REFERENCES


[1] P. Khandelwal, S. Zhang, J. Sinapov, M. Leonetti, J. Thomason, F. Yang, I. Gori, M. Svetlik, P. Khante, V. Lifschitz, J. K. Aggarwal, R. Mooney, and P. Stone, "BWIBots: A platform for bridging the gap between ai and human–robot interaction research," *The International Journal of Robotics Research*, vol. 36, no. 5-7, pp. 635–659, 2017. [Online]. Available: https://doi.org/10.1177/0278364916688949

[2] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, October 2017, pp. 2980–2988. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2017.322

[3] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2," https://github.com/facebookresearch/detectron2, 2019.

[4] He, Zhang, Ren, and Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 0, Las Vegas, Nevada, USA, June 2016, pp. 770–778. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2016.90

[5] Y. Li, H. Mao, R. Girshick, and K. He, "Exploring plain vision transformer backbones for object detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Tel Aviv, Israel: Springer-Verlag, October 2022, pp. 280–296. [Online]. Available: https://doi.org/10.1007/978-3-031-20077-9_17

[6] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proceedings of the International Conference on Learning Representations (ICLR)*, Virtual Only, May 2021. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy

[7] D. Bautista and R. Atienza, "Scene text recognition with permuted autoregressive sequence models," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Tel Aviv, Israel, October 2022, pp. 178–196. [Online]. Available: https://doi.org/10.1007/978-3-031-19815-1_11

[8] D. Zhu, T. Li, D. Ho, T. Zhou, and M. Q. Meng, "A novel ocr-rcnn for elevator button recognition," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Madrid, Spain: IEEE, October 2018, pp. 3626–3631.

[9] D. Zhu, Y. Fang, Z. Min, D. Ho, and M. Q.-H. Meng, "OCR-RCNN: An accurate and efficient framework for elevator button recognition," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 1, pp. 582–591, 2021.

[10] D. Zhu, Z. Min, T. Zhou, T. Li, and M. Q. . H. Meng, "An autonomous eye-in-hand robotic system for elevator button operation based on deep recognition network," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–13, 2021.

[11] K. Chen, D. Holz, C. Rascon, J. R. des Solar, A. Shantia, K. Sugiura, J. Stückler, and S. Wachsmuth, "Robocup@home 2014: Rule and regulations," http://www.robocupathome.org/rules/2014_rulebook.pdf, 2014.

[12] S. Rosenthal, M. Veloso, and A. K. Dey, "Task behavior and interaction planning for a mobile service robot that occasionally requires help," in *Proceedings of the AAAI Conference on Automated Action Planning for Autonomous Mobile Robots*. San Francisco, California, USA: AAAI Press, August 2011, p. 14–19.

[13] R. Simmons, D. Goldberg, A. Goode, M. Montemerlo, N. Roy, B. Sellner, C. Urmson, A. Schultz, M. Abramson, W. Adams *et al.*, "Grace: An autonomous robot for the aaai robot challenge," *AI magazine*, vol. 24, no. 2, pp. 51–51, 2003.

[14] E. Klingbeil, B. Carpenter, O. Russakovsky, and A. Y. Ng, "Autonomous operation of novel elevators for robot navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, Alaska, USA, May 2010, pp. 751–758. [Online]. Available: http://dx.doi.org/10.1109/ROBOT.2010.5509466

[15] S. Gould, O. Russakovsky, I. Goodfellow, P. Baumstarck, A. Ng, and D. Koller, *The stair vision library*, ver. 2.4, Stanford, CA, USA, 2009. [Online]. Available: https://ai.stanford.edu/~sgould/svl/

[16] J. Liu, Y. Fang, D. Zhu, N. Ma, J. Pan, and M. Q.-H. Meng, "A Large-Scale dataset for benchmarking elevator button segmentation and character recognition," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, May – June 2021, pp. 14 018–14 024. [Online]. Available: http://dx.doi.org/10.1109/ICRA48506.2021.9562109

[17] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, Massachusetts, USA, June 2015, pp. 3431–3440.

[18] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Miami, Florida, USA, June 2009, pp. 248–255. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2009.5206848

[19] He, Chen, Xie, Li, Dollár, and Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 0, New Orleans, Louisiana, USA, June 2022, pp. 15 979–15 988. [Online]. Available: http://dx.doi.org/10.1109/CVPR52688.2022.01553

[20] H. Wang, L. Zhou, and L. Wang, "Miss detection vs. false alarm: Adversarial learning for small object segmentation in infrared images," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 8508–8517.

[21] H. Wang, S. Zhu, L. Chen, Y. Li, and T. Luo, "Completeinst: An efficient instance segmentation network for missed detection scene of autonomous driving," *Sensors*, vol. 23, no. 22, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/22/9102

[22] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, Eds., vol. 28. Montreal, Quebec, Canada: Curran Associates, Inc., December 2015. [Online]. Available: https://proceedings.neurips.cc/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf

[23] P.-Y. Yang, T.-H. Chang, Y.-H. Chang, and B.-F. Wu, "Intelligent mobile robot controller design for hotel room service with deep learning arm-based elevator manipulator," in *Proceedings of the International Conference on System Science and Engineering (ICSSE)*, New Taipei City, Taiwan, June 2018, pp. 1–6.

[24] N. Verzic, A. Chadaga, and J. Hart, "The Living with Robots Elevator Button Dataset," 2024. [Online]. Available: https://doi.org/10.18738/T8/ZMIWXS

[25] A. Dutta and A. Zisserman, "The VIA annotation software for images, audio and video," in *Proceedings of the ACM International Conference on Multimedia*, Nice, France, October 2019. [Online]. Available: https://doi.org/10.1145/3343031.3350535

[26] A. Dutta, A. Gupta, and A. Zissermann, "VGG image annotator (VIA)," http://www.robots.ox.ac.uk/ vgg/software/via/, 2016, version: 2.0.11, Accessed: October 18, 2022.