UNIVERSITY OF TECHNOLOGY AND EDUCATION

FALCUTY OF MANUFACTURING AND ENGINEERING

---o0o---

# MACHINE VISION FINAL REPORT:

# EXTRACTING INFORMATION FROM MEDICAL LABELS


**Lecturer: Nguyen Van Thai**

| No | Full Name | Student ID |
|---|---|---|
| 1 | Vu Chi Dat | 21134004 |
| 2 | Tran Nhat Hoang | 21134008 |
| 3 | Lại Thế Trung | 21151405 |


*Ho Chi Minh City, June 2023*

# Contents

# I/ Abstract

Patient compliance with prescribed medication regimens is critical for maintaining health and managing disease and illness. To encourage patient compliance, multiple aids, like automatic pill dispensers, pill organizers, and various reminder applications, have been developed to help people adhere to their medication regimens. However, when utilizing these aids, the user or patient must manually enter their medication information and schedule. This process is time-consuming and often prone to error. For example, elderly patients may have difficulty reading medication information on the bottle due to decreased eyesight, leading them to enter medication information incorrectly. This study explored methods for extracting pertinent information from cylindrically distorted prescription drug labels using Machine Learning and Computer Vision techniques.

# II/ Introduction

## 1. Reason for the project

One of the crucial components of treating a disease is patient compliance with medication regimens; poor adherence or non-adherence can worsen illnesses and possibly lead to death. Medication non-compliance may include aspects such as not filling a prescription, taking an incorrect dose of medication, taking it at the wrong times, changing the frequency of doses, and taking more than prescribed . Medication adherence problems may occur when the medication regimen is complex or when patients have difficulties reading and understanding instructions. Various tools can be used to improve medication adherence, such as behavioral interventions, educational interventions, integrated care interventions, self-management interventions, packaging, and daily reminders; in one study, medication reminder systems showed a 65% improvement in adherence. Moreover, the introduction of medication dispensing systems improves the adherence of older adults within the first week or month .

Various automatic pill dispensers and mobile applications on today's market ensure the prescribed medications are taken on time. Some automatic pill dispensers can even recognize faces to guarantee medication was dispensed to or consumed by the correct patient. However, to utilize these aids, the user's medication information and schedule must be entered manually. Entering this information could be time-consuming and prone to errors since the average national number of prescriptions per capita in the US in 2013 was 12.2. In addition, this process can be difficult for the elderly and people with decreased eyesight since they would have problems reading the information from medication labels

## 2. Aim of the project

This project proposes an automated extraction of pertinent information from real-life medication container images.

As medication labels contain essential information such as patient name, drug name, drug strength, and directions of use. The labels can also be found on the prescriber name, a number of refills, date filled, expiration date, cautions, and description of pills. This information can be used to construct a unique medication schedule for each patient. We implemented a system that automatically extracted and processed information from the prescription drug labels.

This study is motivated by the observed need for such a system, especially for elderly people. To the best of the authors' knowledge, there is no research that solves the problem at hand in a simple way that requires no additional hardware. The proposed approach requires only a simple camera and can be deployed on mobile devices

## 3. Challenges

There are some challenges associated with this objective. First, each pharmacy utilizes a different label format for their medication bottles leading to pertinent information being located inconsistently. This information is spread across the label, requiring the user to take multiple images or a video to capture all of it. Since multiple frames are required, we need to stitch them together, and the first step is to correct cylindrical distortion. This distortion occurs because rectangular labels are wrapped around cylindrical containers.

Additionally, medication directions are provided by physicians and are in free-form text; therefore, there is no universal pattern that would fit all cases. The National Drug Code (NDC), a unique product identifier used in the US for human drugs, is not provided on the prescription drug labels. Therefore, we need to find a way to identify the drug and extract other relevant information.

# III/ Method

## 1.Related Works

Several studies focus on information extraction from drug and other product labels. One of the studies proposed a system that can identify unapproved and potentially dangerous medications based on the label using a deep learning approach; Connectionist Text Proposal Network (CTPN) to extract sub-images based on the text followed by OCR; then, the text was vectorized using universal sentence embedding for finding cosine similarity to reference images. Another study extracted metadata from retail product label images. This study used various Computer Vision and NLP techniques to assess the quality of the image and extract the brand and product name, nutrition facts, and net weight extraction. In addition, another study implemented a system with drug label detection functionality for the elderly. Connected Component Analysis (CCA) algorithm with two key steps of blob filtering and line construction to extract text from the label. Importantly, these methods analyzed.

Some of outstanding method using traditional Computer Vision algorithms as well as Deep Convolutional Neural Networks (DCNN) to localize and unwarp the label. DCNNs showed a better performance and were found to be more resistant to lighting conditions and background changes. The processed label images were then stitched together to obtain the whole label. Different image preprocessing methods were explored to identify the best way for successful text extraction using Optical Character Recognition (OCR). Information such as patient name, drug name, drug strength, and directions of use were extracted from the recognized text using Natural Language Processing (NLP) techniques.

## 2.Project Solution

### *2.1 Study Approach :*

In this project, we use the basic approach mostly depending on Machine Vision algorithms method combine with OCR method to extract text information from drug labels.

The approach following several steps:

1) Collect images of medication label: the images will be taken using camera and rescaled to the standard size of the system.
2) Image preprocessing: diagnoise and increase the constract of the components in the object.
3) Label Dewarping: sometime the label will be distorted, declined or wrap into cylinder shape, these need to be dewarped or perspected into flat rectangle shape of labels.
4) Connecting dataset: the list of images after dewraping will be concatenate or stitching together.
5) Text detection: detect the region that have texts.
6) Text recognition: recognize the words.
7) Key Information extraction: extract key information from unstructured document. images to get key – value pairs.
8) Visualize and saving the result.

### *2.2 Image preprocessing:*

The aim of this step is diagnoise and get the region of the label.

To get easy to detect the drug product box or bottle, we setup the product in the black background.


*Figure 1 RGB to HSV image*

Changing the image from RGB to HSV channel. Calib the HSV mask to filt the black background

```
h_min = 0   #hue
h_max = 179 #hue
s_min = 0   #sat
s_max = 255 #sat
v_min = 90  #val
v_max = 255 #val
```

Always, the image will be taken in different enviroments, different light conditions.

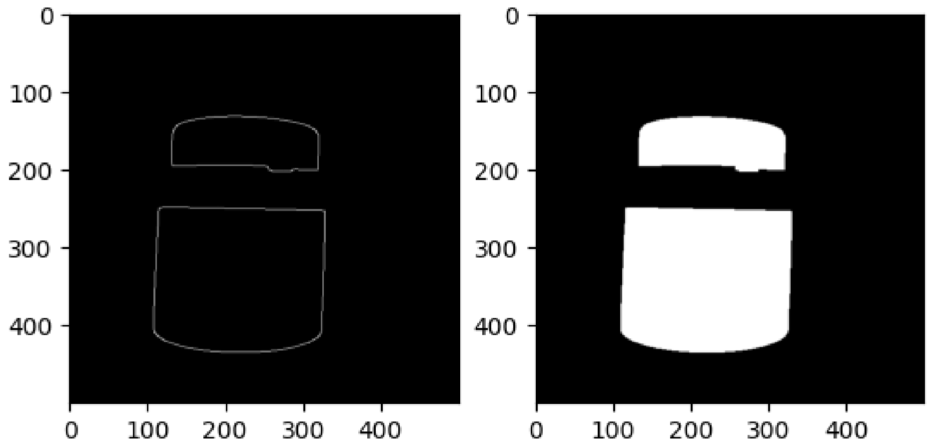

*Figure 2 the left image is to much lighting and dazzling, the right image is in good condition*

*Figure 3 Example of some inevitable noise, ( the noise is highlighted by red region)*

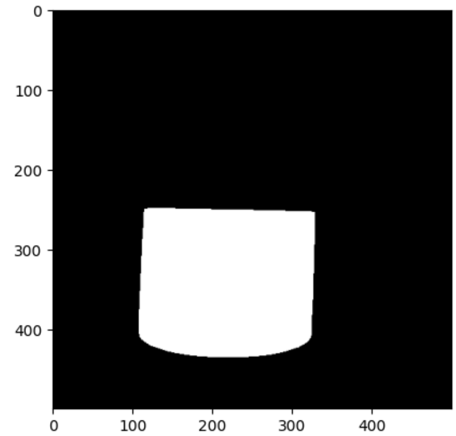
Using morphology methods, dilate the image with 1 iteration to vanish the black noise in the label, after that erode image with 6 iterations to reduce the noise caused by dazziling, finally, dialate with 5 iterations to make the image return to the original size. we diagnoise it using the following codes:

```
kernel = np.ones((5, 5), np.uint8)
#diagnoise
dialation = cv2.dilate(mask, kernel, iterations = 1)
erodesion = cv2.erode(dialation,kernel, iterations= 6)
diagnoised = cv2.dilate(erodesion, kernel, iterations = 5)
blur = cv2.GaussianBlur(diagnoised,(5,5),0)
edged = cv2.Canny(blur, 0, 100)
```

And the result is better:



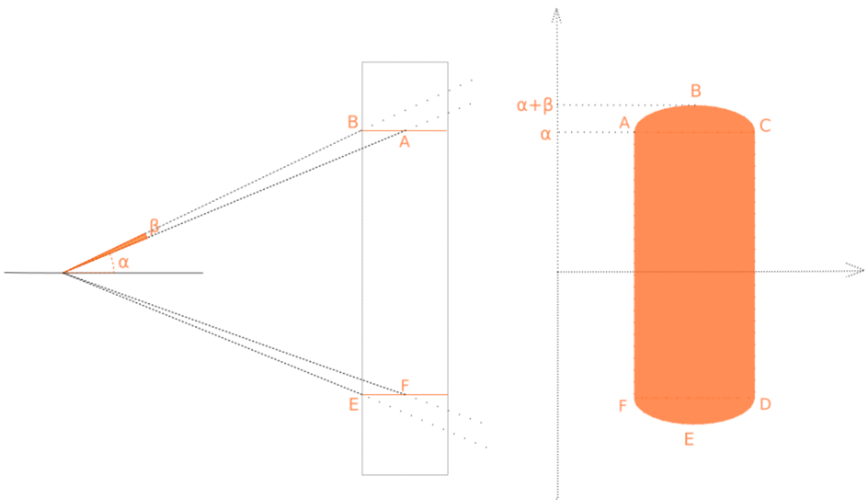

Before finish all the step above, we receive a mask distinguishing between the back ground and drug products. The following job is extract the coordinate of the label.To do that, we firstly find the contours and then caculate the Mean value of fx and fy. The Mean value is present the center point of the points in each contour. Comparing the center points position, we can get the the region of label. We get the final mask detect the region of label




### *2.3 Label Dewraping*

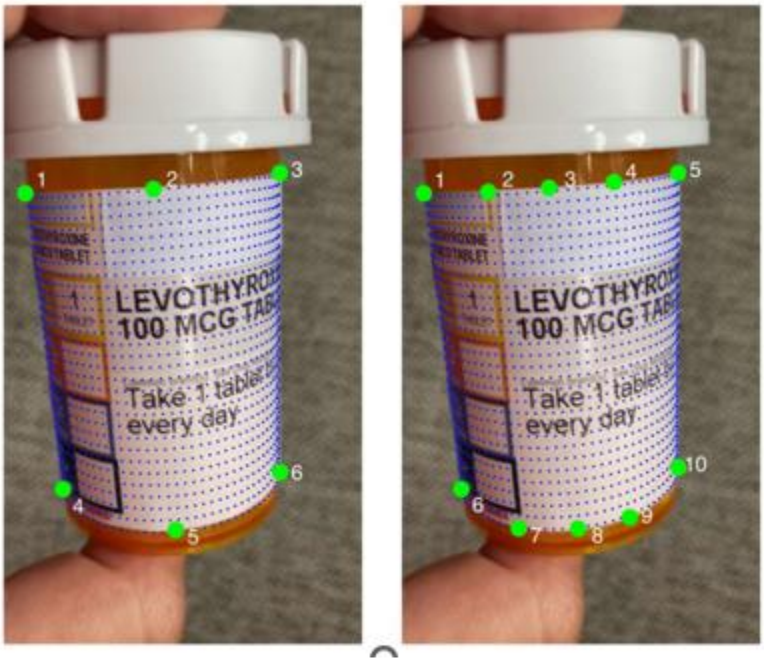
These containers usually have a cylindrical shape, and the rectangular labels are wrapped around them. Therefore, magnification reaches its maximum at the center of the label and minimum at its vertical edges. The model resembles barrel distortion but with different warping along the vertical axis. In our case, the upper and lower edges of the label are ellipses formed by circles viewed from an angle. The half-ellipses and vertical edges of the label defined our warp model.

We sampled an equal number of points on both ellipses with a fixed angle step. The equidistant points were placed on lines connecting points with the same angle on both ellipses. These points defined our warp mesh. Perspective rendering is used to transform quadrangles define by each point and its three neighbors into fixed rectangles. The resulting rectangles were concatenated to output the dewarped image. Since the quadrangles located close to vertical edges are narrow, the result of perspective rendering is blurry. Therefore, we cropped 15% from the left and right sides of the final images. The percentage was chosen empirically to reliably eliminate artifacts.

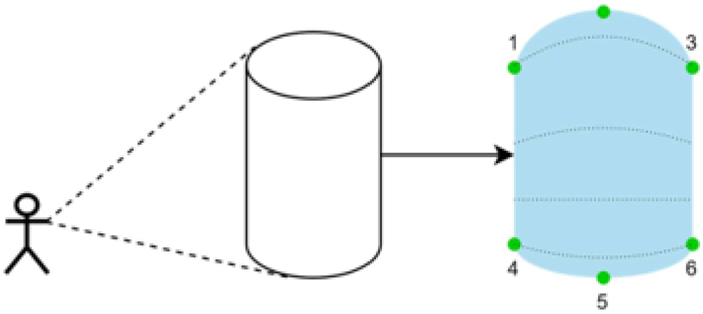



a. How the observer sees the label

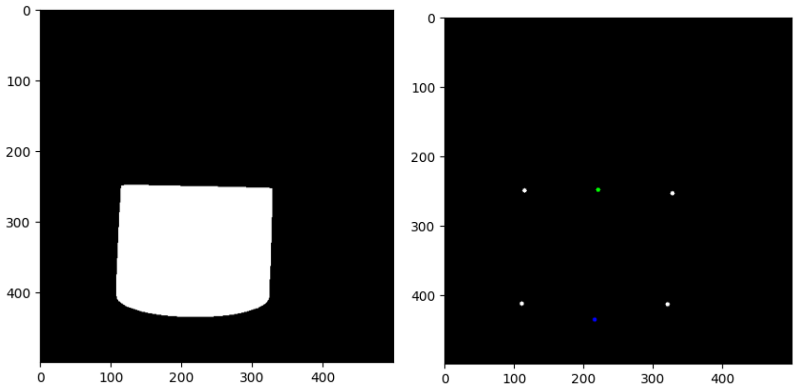
b. The actual picture of the label seen by the obsever




In this project, we apply a simple method used to detect 6 points
This method will only apply for the cylinder wrapping with the shape similar the figure below




1) using the bounding rectangle to detect the highest and lowest point
2) using the approx poly with 0.05% error to get 4 boundary points op the rectangle
Here is the result in practice:



Then apply the flatten algorithms to dewarping the warped image using interpolation

Here is the code:

```
def calc_dest_map(self):
        width, height = self.get_label_size()

        dx = float(width) / (self.COL_COUNT - 1)
        dy = float(height) / (self.ROW_COUNT - 1)

        rows = []
        for row_index in range(self.ROW_COUNT):
            row = []
            for col_index in range(self.COL_COUNT):
                row.append([int(dx * col_index),
                            int(dy * row_index)])

            rows.append(row)
        return np.array(rows)

def unwrap_label_interpolation(self, source_map):
        """
        Unwrap label using interpolation - more accurate method in terms of quality
        """
        from scipy.interpolate import griddata

        width, height = self.get_label_size()

        dest_map = self.calc_dest_map()

        grid_x, grid_y = np.mgrid[0:width - 1:width * 1j, 0:height - 1:height * 1j]

        destination = dest_map.reshape(dest_map.size // 2, 2)
        source = source_map.reshape(source_map.size // 2, 2)

        grid_z = griddata(destination, source, (grid_x, grid_y), method='cubic')
        map_x = np.append([], [ar[:, 0] for ar in grid_z]).reshape(width, height)
        map_y = np.append([], [ar[:, 1] for ar in grid_z]).reshape(width, height)
        map_x_32 = map_x.astype('float32')
        map_y_32 = map_y.astype('float32')
        warped = cv2.remap(self.src_image, map_x_32, map_y_32, cv2.INTER_CUBIC)
        self.dst_image = cv2.transpose(warped)
```

And the result is shown


*Figure 4 Original Warped label*

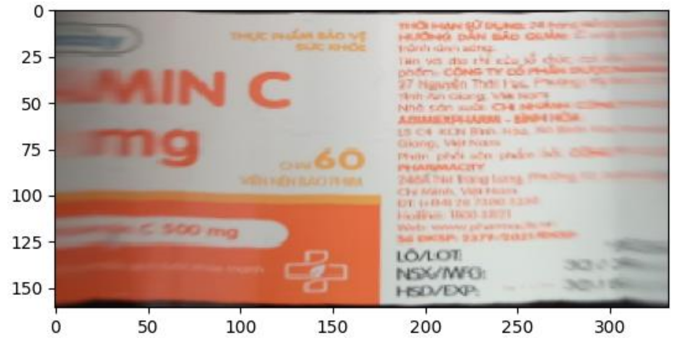



*Figure 5 Output of dewarping algorithms*

The same image processing method can be apply to the box and have very good result:



### *2.4 Connect Images*

Working with more than 1 image, we will conncet 4 or 10 images into 1 image using different method with different type of shape of the medical products.

With the box shape, we simply concatenate all the input processed label images into 1 image


Multiple dewarped images were stitched together to create a complete label. There are various approaches for image stitching, and one of the most effective algorithms was introduced by Brown and Lowe in *Automatic Panoramic Image Stitching using Invariant Features* paper . OpenCV library's implementation of the image stitching algorithm is based on this algorithm. This algorithm can stitch multiple images together while ignoring unrelated ones. Moreover, the input images do not have to have a specific order. Due to these reasons, we have decided to use the standard OpenCV stitcher in the proposed system.

Using stiching library of python, After configurate some indicators, we got the acceptable performance of the method shown in the figure below:

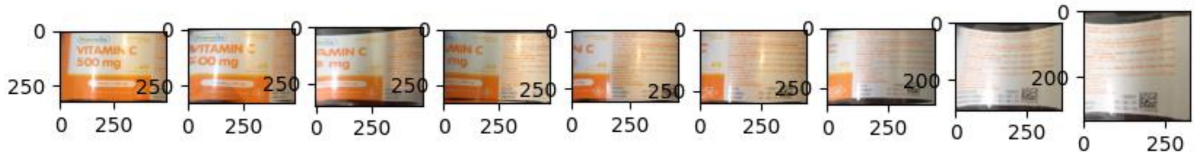


*Figure 6 the 9 input processed label images*

*Figure 7 The output panorama of the stiching method*

### *2.5 OCR applying*

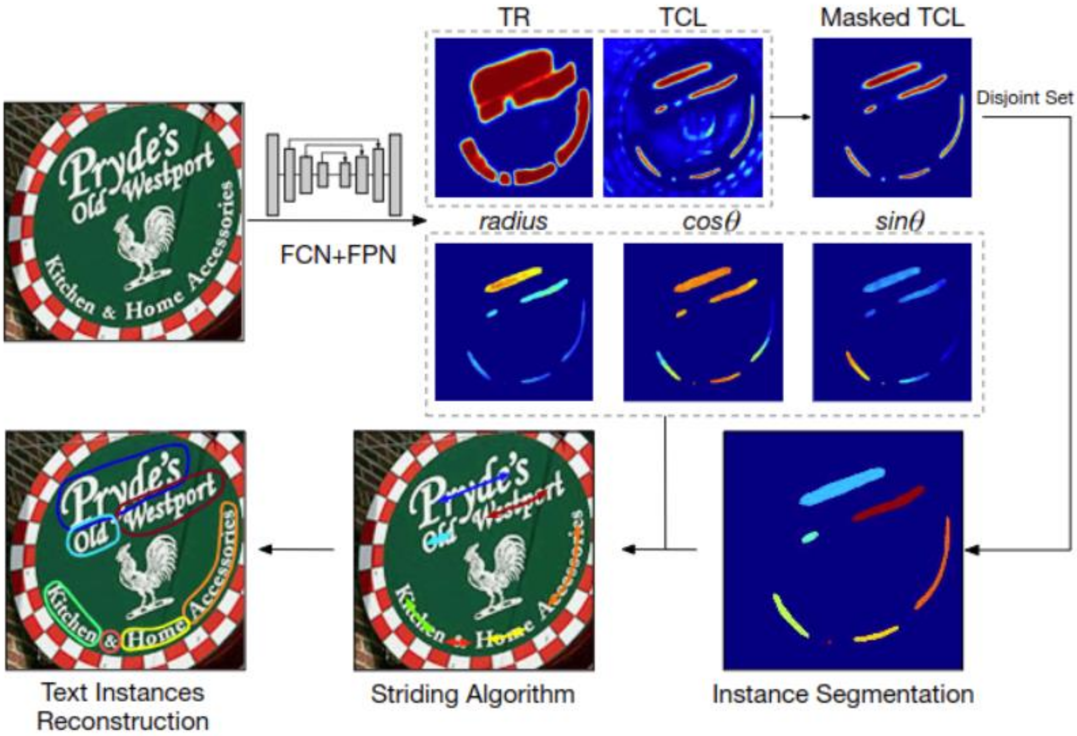
In this project we use the OCR library from open-mmlab.

*2.5.1 Text detection only:*

For text detection, we use TextSnake model which performs very good in detect text in different font and different poses and the output is the segmentation of words, moreover it can show the connection of words in a line but hard for the recognizer to recognise the words.

Driven by deep neural networks and large scale datasets, scene text detection methods have progressed substantially over the past years, continuously refreshing the performance records on various standard benchmarks. However, limited by the representations (axis-aligned rectangles, rotated rectangles or quadrangles) adopted to describe text, existing methods may fall short when dealing with much more free-form text instances, such as curved text, which are actually very common in real-world scenarios. To tackle this problem, we propose a more flexible representation for scene text, termed as TextSnake, which is able to effectively represent text instances in horizontal, oriented and curved forms. In

TextSnake, a text instance is described as a sequence of ordered, overlapping disks centered at symmetric axes, each of which is associated with potentially variable radius and orientation. Such geometry attributes are estimated via a Fully Convolutional Network (FCN) model. In experiments, the text detector based on TextSnake achieves state-of-the-art or comparable performance on Total-Text and SCUT-CTW1500, the two newly published benchmarks with special emphasis on curved text in natural images, as well as the widely-used datasets ICDAR 2015 and MSRA-TD500. Specifically, TextSnake outperforms the baseline on Total-Text by more than 40% in F-measure.

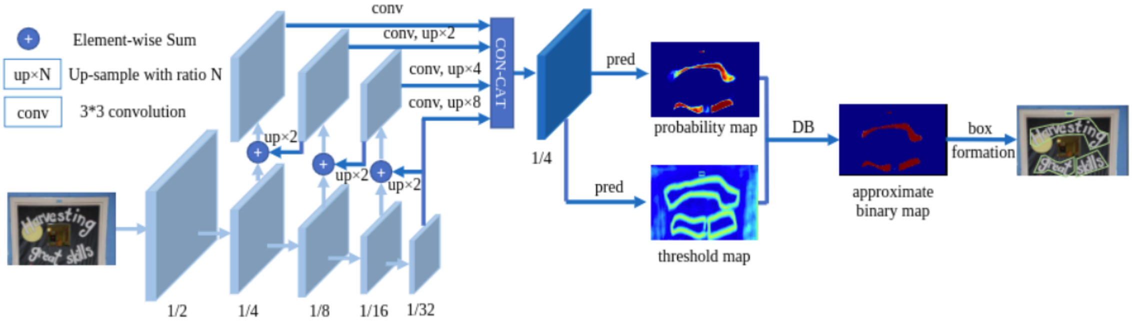



### *2.5.2 Text Detection + Recognition:*

The combination of the pair DBNet and SAR always give the overexpectation result in detect and recognize the words. The only weakness is that it does not show the relation between words. The solution we came up with is combine text detection+recognition method with text detection only method to show the better performance.

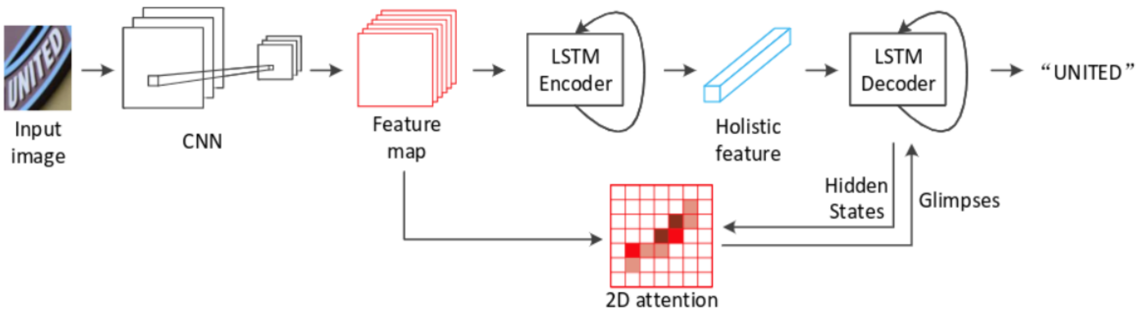
a) DBNet:

Recently, segmentation-based methods are quite popular in scene text detection, as the segmentation results can more accurately describe scene text of various shapes such as curve text. However, the post-processing of binarization is essential for segmentation-based detection, which converts probability maps produced by a segmentation method into bounding boxes/regions of text. In this paper, we propose a module named Differentiable Binarization (DB), which can perform the binarization process in a segmentation network. Optimized along with a DB module, a segmentation network can adaptively set the thresholds for binarization, which not only simplifies the post-processing but also enhances the performance of text detection. Based on a simple segmentation network, we validate the performance improvements of DB on five benchmark datasets, which consistently achieves state-of-the-art results, in terms of both detection accuracy and speed. In particular, with a light-weight backbone, the performance improvements by DB are significant so that we can look for an ideal tradeoff between detection accuracy and efficiency. Specifically, with a backbone of ResNet-18, our detector achieves an F-measure of 82.8, running at 62 FPS, on the MSRA-TD500 dataset.



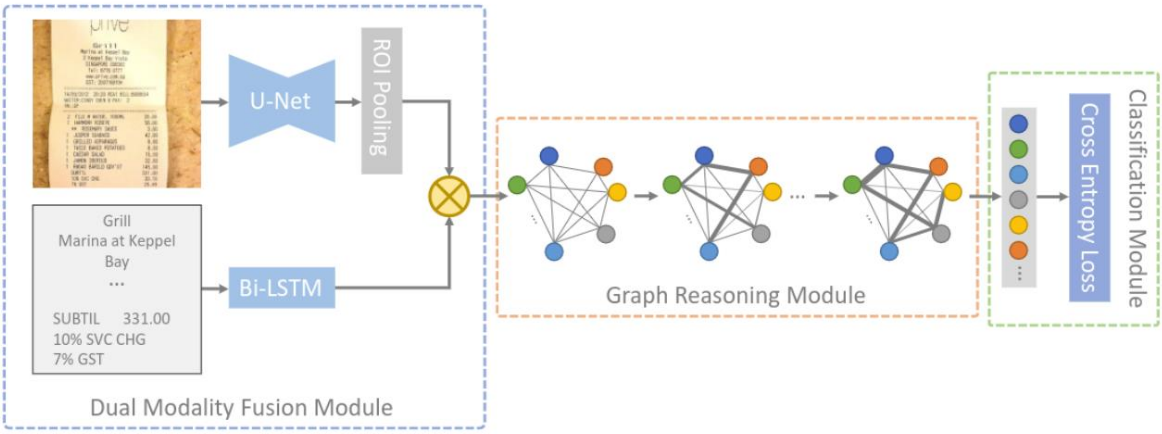

b) SAR:

Recognizing irregular text in natural scene images is challenging due to the large variance in text appearance, such as curvature, orientation and distortion. Most existing approaches rely heavily on sophisticated model designs and/or extra fine-grained annotations, which, to some extent, increase the difficulty in algorithm implementation and data collection. In this work, we propose an easy-to-implement strong baseline for irregular scene text recognition, using off-the-shelf neural network components and only word-level annotations. It is composed of a 31-layer ResNet, an LSTM-based encoder-decoder framework and a 2-dimensional attention module. Despite its simplicity, the proposed method is robust and achieves state-of-the-art performance on both regular and irregular scene text recognition benchmarks.




## *2.5.3 Key information extraction:*

Key information extraction from document images is of paramount importance in office automation. Conventional template matching based approaches fail to generalize well to document images of unseen templates, and are not robust against text recognition errors. In this paper, we propose an end-to-end Spatial Dual-Modality Graph Reasoning method (SDMG-R) to extract key information from unstructured document images. We model document images as dual-modality graphs, nodes of which encode both the visual and textual features of detected text regions, and edges of which represent the spatial relations between neighboring text regions. The key information extraction is solved by iteratively propagating messages along graph edges and reasoning the categories of graph nodes. In order to roundly evaluate our proposed method as well as boost the future research, we release a new dataset named WildReceipt, which is collected and annotated tailored for the evaluation of key information extraction from document images of unseen templates in the wild. It contains 25 key information categories, a total of about 69000 text boxes, and is about 2 times larger than the existing public datasets. Extensive experiments validate that all information including visual features, textual features and spatial relations can benefit key information extraction. It has been shown that SDMG-R can effectively extract key information from document images of unseen templates, and obtain new state-of-the-art results on the recent popular benchmark SROIE and our WildReceipt. Our code and dataset will be publicly released.


U-Net
ROI Pooling
Grill
Marina at Keppel
Bay
...
SUBTIL 331.00
10% SVC CHG
7% GST
Bi-LSTM
Dual Modality Fusion Module
Graph Reasoning Module
Cross Entropy Loss
Classification Module

# IV/ Result

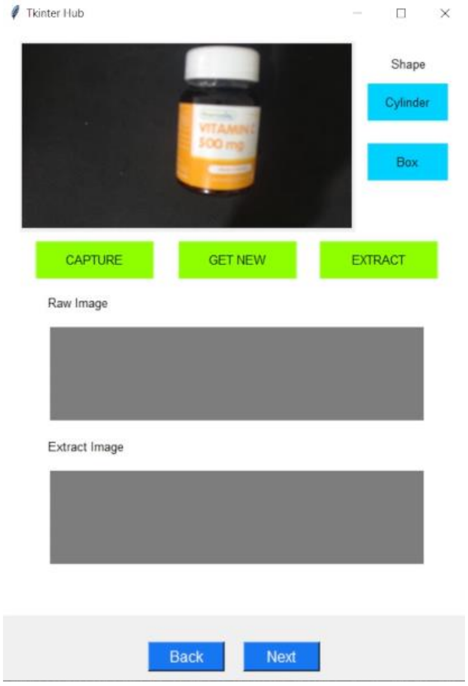
## 1. Application design and apply

### *1.1 Hardware:*

Logitech Camera C922 PRO HD STREAM WEBCAM (HD 1080p/30fps or 720p/60fps)

### *1.2 Solfware application:*

Using tkinter to design the GUI




Using tutorial:
Step 1: Chosing the shape of the product. Right before the click, the maximum number of images in the dataset will be declared ( 4 for box shape and 10 for cylinder shape)
Step 2: Click the CAPTURE to take the dataset. If the input data is not good enough, the app will show the message window “ the image is unqualified”.
Step 3: When you get enough datas, click EXTRACT button to extract the data and save it in to data.csv file.
Step 4: To get new dataset, click GET NEW.

## 2. Testing and Evaluation

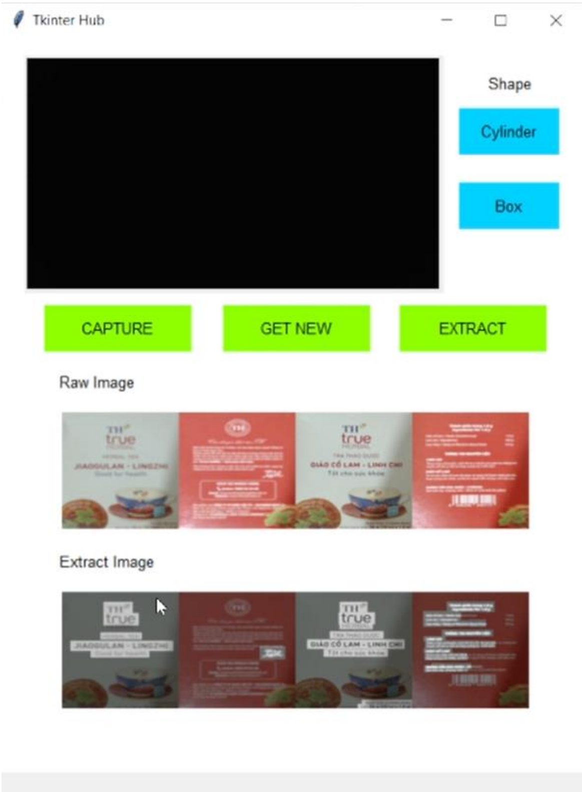



*Figure 8 Test for box shape product*

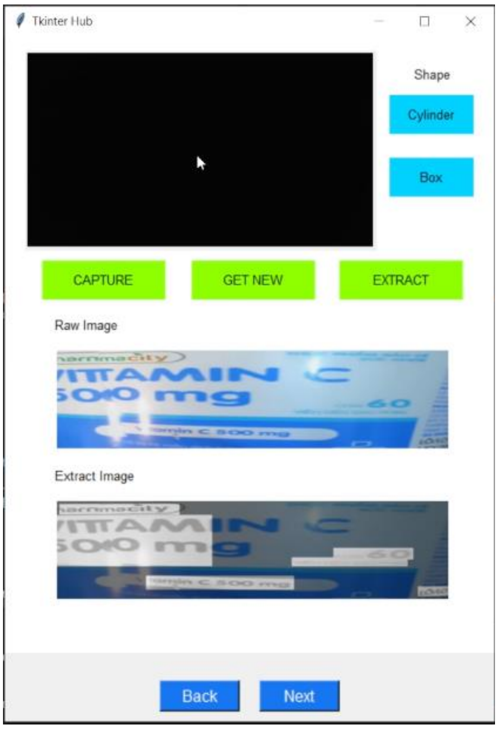



*Figure 9 Test for cylinder shape product*

Data.csv read:

```
coordinate:,word:
"[[2070, 437], [2226, 427], [2228, 457], [2072, 467]]",LOVO
"[[202, 116], [287, 82], [308, 134], [223, 168]]",it
"[[124, 39], [135, 39], [135, 46], [124, 46]]",-
```

Evaluation: The performance is not so good, But is show mostly the idea of this project.

# Reference


[1] *Automatic Extraction of Medication Information from Cylindrically Distorted Pill Bottle Labels* of Kseniia Gromova and Vinayak Elangovan

[2] https://github.com/open-mmlab/mmocr

[3] https://github.com/Nepherhotep/unwrap_labels