The entire code repository can be found at `https://github.com/dgsaf/comp3007_assignment`.

Contents

List of Figures

List of Tables

1 Introduction

2 Problem Analysis

We are concerned with the detection and classification of two similar but distinct types of signs. The properties specific to each type of signage are discussed in subsection 2.1 and subsection 2.2. Here, we remark on the properties common to both of them, which are numerous and allow for a broadly unified approach to their detection and classification.

Each sign consists of a black background with white characters (digits and directional arrows) forming the foreground of the sign. The black background is rectangular in shape, and bounds the characters regions. Furthermore, the contrast between the black background and the white foreground is strong across all colour channels and in grayscale.

The digits are uniform in their construction, being printed in a monospace font with fixed height and allocated a fixed width; although each digit does not necessarily fully extend across its allocated width - as can be seen for the digit 1. Likewise, the directional arrows are also uniform in their construction but their dimensions are distinct from those of the digits - noticeably having a smaller height than the digits.

The digits do not overlap with each other (unless subject to camera artifacts/blurring) but are closely adjacent to each other, and regularly spaced. Each sign contains at least one sequence of three digits, which are ideally surrounded by a black sub-region of the sign; although for task 2, the gap between the leftmost digit and the edge of the sign can be very small.

In summary, the following properties of the signs are almost always observed, providing a robust foundation for their detection:

- The uniformly white colour of the characters;
- The uniformly black background of the sign;
- The strong contrast between the white characters and the black background across all red, green, and blue channels as well as in grayscale;
- The uniform height of the digits;
- The uniform spacing of the characters horizontally;
- The height of the digits exceeding the width;
- The uniform dimensions of the arrows;
- The presence of a chain of exactly three digits;
- A sizeable sub-region of the sign above and below the chain of digits, which is uniformly black.

However, the assumption of the above properties may be invalidated by any one of the (non-exhaustive) list of conditions:

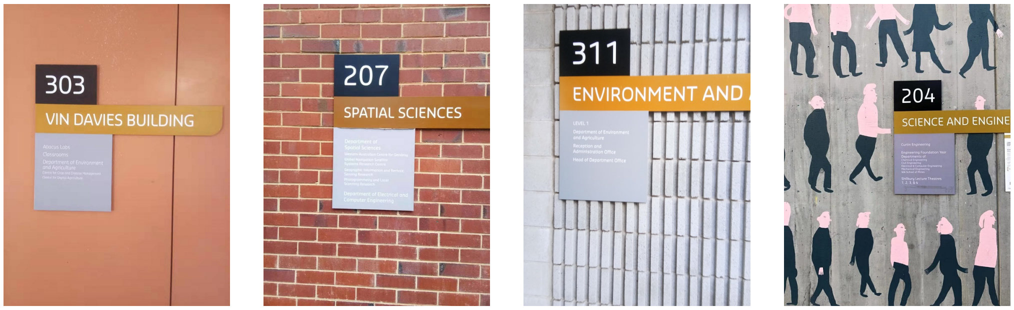
- The presence of strong radial, or motion blurring which may cause character to overlap;
- The presence of strong shadowns, glare, or other variations in lighting conditions across the character;

- The presence of a foreign object, such as a sticker or a mark, on the sign (violating the assumptions on uniformity of layout) or across the characters (violating the assumptions on uniformity of characters).

It should also be noted that often there are other monospace white characters present near the sign - typically letters forming the name of the location marked by the sign. These characters can be distinguished from the digits (and arrows) by the fact that they almost always form chains of more than three characters, and presented on their own signs which almost never have a black background.

2.1 Task 1

Further refinement of the problem analysis is possible for task 1. We are concerned with finding only one chain of three digits, and we are not concerned with directional arrows at all. Futhermore, there exists a sizeable gap between the chain of digits and the edge of the sign both horizontally and vertically.
Bricks and other objects which are of approximately uniform dimensions and uniformly spaced are regularly present in images for task 1 - hence, when exploiting the uniformity of the layout of the digits, care must be taken to avoid or handle the false detection of these other objects.
Examples of the images expected for task 1, which provide the justification for the assumptions made on the properties of the sign are shown in Figure 1. Examples of images which may be encountered for task 1, but which challenge the assumptions made on the properties of the sign are shown in Figure 2.

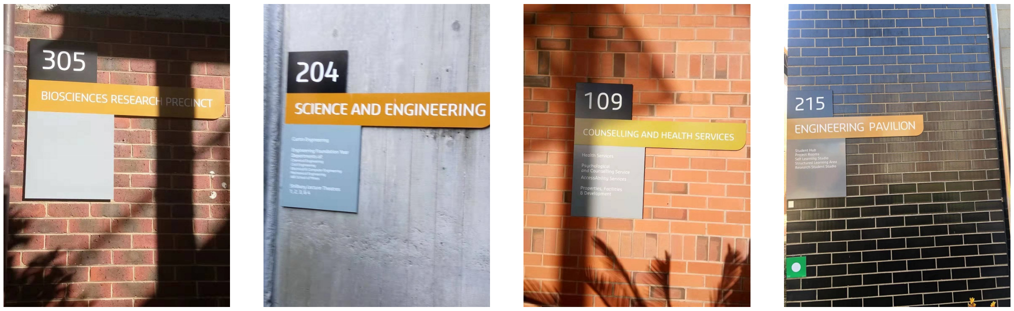


(a) BS02.jpg *(b) BS04.jpg* *(c) BS09.jpg* *(d) BS14.jpg*

Figure 1: Examples of training images for task 1, which exhibit the described distinguishing properties of the signs and digits.

2.2 Task 2

Similarly, further refinement of the problem analysis is possible for task 2. We are concerned with finding a variable number of chains of three digits, each of which has an associated directional arrow to their right. For each chain and their associated directional arrow, the geometric centres of the digits are arrow are collinear; hence we say that they form a line. Each line is vertically spaced (by a distance just larger than the height of the digits) and the set of digits and arrows from each line thus forms a grid-like layout.


305
BIOSCIENCES RESEARCH PRECINCT
204
SCIENCE AND ENGINEERING
109
COUNSELLING AND HEALTH SERVICES
215
ENGINEERING PAVILION

(a) `BS03.jpg` *(b)* `BS05.jpg` *(c)* `BS07.jpg` *(d)* `BS11.jpg`

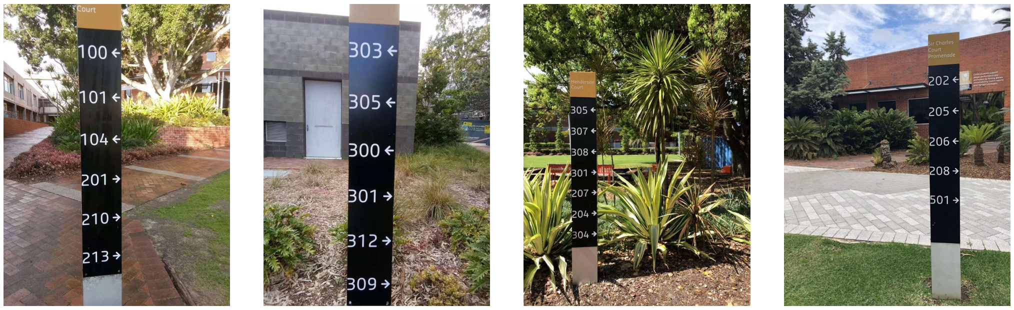
Figure 2: Examples of training images for task 1, which may challenge the assumptions made on the signs and thus necessitate the development of a robust detection algorithm. Sharp shadows, motion blur and monochromatic periodic objects, which may interfere with digit detection, can be observed.

A challenging aspect of these signs, compared to those for task 1, is that the gap between the leftmost digit on each line and the edge of the sign can be very small. In cases where an object, of similar colour/intensity to the white characters, is behind the sign but in a similar region of the image to the leftmost digit of a line, the digit and the object may prove difficult to distinguish as separate regions.

Another difficulty is that the number of lines in any particular sign is not known beforehand. However, there is a benefit in having multiple lines per sign as it provides a mechanism, given at least one fully known line or even partial information from multiple lines, to recover an expected location for characters which fail to be detected initially.

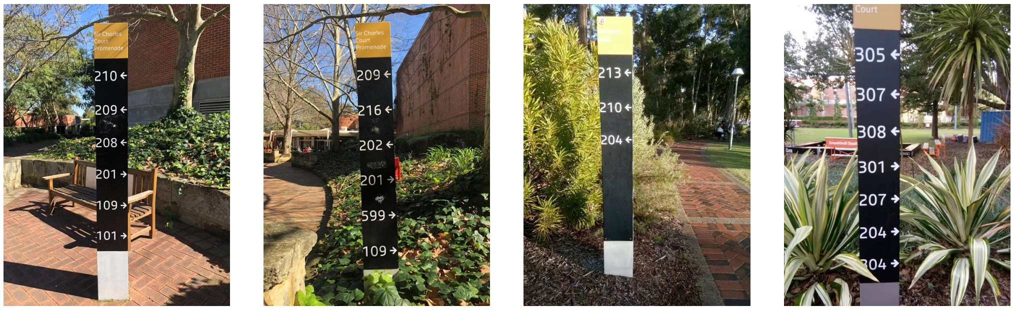
Similar to task 1, the scene environment can be expected to be quite noisy, with periodically repeating objects such as brickwork present, as well as other objects with many sharp edges, such as plants and trees. It also seen that foreign objects, such as stickers and marks, may be present on the signs, and that other objects, such as plants, may occlude the sign itself.

Examples of the images expected for task 2, which provide the justification for the assumptions made on the properties of the sign are shown in Figure 3. Examples of images which may be encountered for task 2, but which challenge the assumptions made on the properties of the sign are shown in Figure 4.



(a) `DS06.jpg` (b) `DS12.jpg` (c) `DS13.jpg` (d) `DS18.jpg`

Figure 3: Examples of training images for task 2, which exhibit the described distinguishing properties of the signs, and their digits and arrows.



(a) `DS02.jpg` (b) `DS07.jpg` (c) `DS10.jpg` (d) `DS11.jpg`

Figure 4: Examples of training images for task 2, which may challenge the assumptions made on the signs and thus necessitate the development of a robust detection algorithm. Characters overlapping with similar regions outside the sign, foreign objects on the sign, variable lighting conditions can all be observed.

3 Implementation

3.1 Overview

We make use of the commonality of the properties of the signs that are to be detected in task 1 and task 2 by designing a detection algorithm which can be adapted with minor changes for each task. The specific detection algorithms for each task are discussed in subsection 3.5 and subsection 3.6. Furthermore, as the digits to be classified in each task are of the same style, we also design a classification algorithm which is suitable for both tasks. Naturally, a separate classifier will be needed for the directional arrows, however we use the k-Nearest Neighbours (k-NN) algorithm for classifying both the digits and the arrows. Hence, we discuss the problem of classification, for both types of characters, in a general manner in subsection 3.3.

3.2 Detection

To detect the characters (digits and arrows) we utilise the method of Maximally Stable Extremal Regions (MSER) - first developed by [Matas et al., 2004]. We present our formalism in a similar manner to [Matas et al., 2004]. In this method, a number of regions, which are extremal in terms of light intensity and stable in the sense that the region boundary changes little under a varying threshold, are detected in the grayscale image. We suggest that this technique is suitable for our purpose, as the characters are uniformly white and contrast well against the surrounding uniformly black bounding area of the signs. Furthermore, in most cases the characters are well separated from other characters and the edge of the sign (except notably for the leftmost digits in task 2). Hence, these characters should form extremal regions which are relatively stable under affine transformations, and when the image is underexposed or overexposed. Theoretically, this technique may be susceptible to extreme variations in lighting conditions, such as strong shadows or glare, which form discontinuous non-monotonic changes in light intensity. However, in testing, this technique has proved tolerant to moderate shadows and glare, so we do not expect this to be a major issue.

With suitably chosen parameters for the MSER algorithm, the regions of interest, corresponding to the characters on the sign, can be reliably detected - but must be distinguished from the other regions detected which are not of interest. Using known properties of the characters (such as dimension, aspect ratio, bounding box fill) we are able to refine the set of regions.

To detect the regions corresponding to the sequences of digits, we then examine the geometric relationships between the regions to determine which pairs of regions are: similar in dimension, and possibly collinear. This results in a directed graph of regions, with paths in this graph constituting possible lines of similar regions. For each region in this graph we filter their set of edges to leave only the edge connecting it to the closest adjacent region - ensuring all paths are non-overlapping. We then extract all possible paths, which we call chains, and analyse each chain by length and by contrast (in gray scale as well as in red, green, and blue channels) to order the chains by their likelihood of being a sequence of three digits.

For task 1, we detect only the most likely chain as our sequence of digits, and the problem of detection is complete. For task 2, we cluster these chains by their likelihood and select the most likely subset of the chains. If any of the sequences of digits have been only partially found (e.g. two out of three digits), we make use of the grid-like layout of the digits to locate the remaining digit. A bounding box is drawn around where the remaining digit is expected to be present, and a connected-components algorithm is employed to construct a region for this digit which is limited to the scope of this bounding box. With all regions corresponding to the digit sequences now found, the collinearity of the digits is used to locate the region corresponding to the directional arrow associated with these digits. Thus for each task, the detected lines of digits (and directional arrows for task 2) are now located.

3.3 Classification by k-Nearest Neighbours

Prior to the classification stage, we utilise the collinearity of the lines, and the uniform height of the digits, to construct an affine transformation which rotates the line to be horizontal. While a full perspective transformation could be employed, we observed that an affine transformation was sufficient in the absence of extreme perspective distortions.

To address the problem of digit and directional arrow classification, we use the k-NN algorithm, comparing the regions expected to be digits and arrows with the training digits and arrows respectively. For each region, we define a binary image (with the dimensions of the bounding box of the region) which maps each point that is in the region to 1, and 0 otherwise. We then decompose the binary image into a set of bins (which are symmetric about the bounding box midpoint) and use the spatial occupancy of each bin as a feature vector for the region. We suggest that this approach is suitable, as the digits and directional arrows are of a uniform style and dimensions, and we note that this feature vector is scale-invariant. In testing with a suitably selected number of bins, this approach has proved to be reliably accurate despite its simplicity.

3.4 Mathematical Details

We introduce here various formal presentations of the concepts used in the detection and classification algorithms.

Image

We define an image I, with n channels and of width w and height h, to be a mapping

$$I : D \to S : (x, y) \mapsto I(x, y) = \big(I_1(x, y), \dots, I_n(x, y)\big), \tag{1}$$

where the domain $D \subset \mathbb{N}^2$ is of the form $D = \{0, \dots, w-1\} \times \{0, \dots, h-1\}$, and S is the codomain. Constraints can be specified on S, but for our purposes it suffices to suppose that either $S \subseteq \{0, \dots, 255\}^n$ or $S \subset \mathbb{R}^n$.

Thresholded Image

Suppose $I : D \to S \subset \mathbb{R}$ is a single-channel image. For each $t \in S$, we define the Boolean-valued image I_t to be of the form

$$I_t : D \to \mathbb{B} : (x, y) \mapsto \begin{cases} 0 & \text{if} \quad I(x, y) \leq t \\ 1 & \text{if} \quad I(x, y) > t \end{cases}, \tag{2}$$

and we say that I_t is a thresholded image.

Bounding Box

Suppose D is the domain of an image $I : D \to S$. For all $R \subseteq D$, we define the bounding box $B(R)$ of R to be

$$B(R) = [x_R, x_R + w_R] \times [y_R, y_R + h_R] \quad \text{such that} \quad R \subseteq B(R) \tag{3}$$

and where $w_R, h_R \in \mathbb{N}$ are minimal.

Connectedness

Suppose D is the domain of an image. An adjacency relation A on D is a Boolean-valued mapping

$$A : D \times D \to \mathbb{B} : (p, q) \mapsto A(p, q), \tag{4}$$

which indicates if the two points of the domain are considered to be adjacent. For any $p \in D$, we may define the neighbourhood $N(p)$ of p to be the set of all points which are adjacent to it; that is,

$$N(p) = \{q \in D \mid A(p, q)\}. \tag{5}$$

For any $p, q \in D$, we say that p and q are connected if there exists a finite sequence $(\rho_k)_{1 \leq k \leq n}$ in D such that

$$A(p, \rho_1) \wedge A(\rho_1, \rho_2) \wedge \cdots \wedge A(\rho_{n-1}, \rho_n) \wedge A(\rho_n, q) = 1. \tag{6}$$

In the case where the adjacency relation A is symmetric, then connectedness defines an equivalency relation; whence we may write, for any connected $p, q \in D$ that $p \sim q$.
Note that we are primarily concerned with the adjacency relations associated with the Von Neumann neighbourhood (4-connectivity)

$$N_4(p) = \{p + n \in D \mid n \in \{(0, 1), (1, 0), (0, -1), (-1, 0)\}\}, \tag{7}$$

and the Moore neighbourhood (8-connectivity)

$$N_8(p) = \{p + n \in D \mid n \in \{-1, 0, 1\} \times \{-1, 0, 1\} \setminus (0, 0)\}. \tag{8}$$

Region

Suppose D is the domain of an image $I : D \to S$ and let $A : D \times D \to \mathbb{B}$ be a symmetric adjacency relation on D. We say that $R \subseteq D$ is a region if every element of R is connected to every other element of R; that is,

$$p, q \in R \implies p \sim q. \tag{9}$$

We define the (inner) boundary ∂R of a region R to be subset of points of R which are also connected to at least one point not in R; that is,

$$\partial R = \{p \in R \mid \exists q \in D \setminus R : A(p, q)\}. \tag{10}$$

We define the outer boundary ΔR of a region R to be the set of points of D which do not belong to R but are adjacent to a point of R; that is,

$$\Delta R = \{p \in D \setminus R \mid \exists q \in R : A(p, q)\}. \tag{11}$$

Maximally Stable Extremal Region (MSER)

Suppose $I : D \to S \subset \mathbb{R}$ is a single-channel image, and suppose $A : D \times D \to \mathbb{B}$ is the (symmetric) adjacency relation associated with either the Von Neumann neighbourhood or the Moore neighbourhood. Suppose $R \subseteq D$ is a region. We say that R is a minimal region if for all $p \in R$ and $q \in \Delta R$ we have $I(p) < I(q)$; which is equivalently written as the requirement that

$$\max_{p \in R} I(p) < \min_{q \in \Delta R} I(q). \tag{12}$$

Similarly, we say that R is a maximal region if for all $p \in R$ and $q \in \Delta R$ we have $I(p) > I(q)$; which is equivalently written as the requirement that

$$\min_{p \in R} I(p) > \max_{q \in \Delta R} I(q). \tag{13}$$

We say that R is an extremal region if it is either a minimal or maximal region.
The formulation of extremal regions in terms of the minimal and maximal intensity values of the image permits the usage of thresholding. Suppose that R is an extremal region, and suppose that $t \in S$. Consider the thresholded region R_t, defined by

$$R_t = \{p \in R \mid I(p) < t\}, \tag{14}$$

which is itself an extremal region, and for which we have that

$$\max_{p \in R_t} I(p) < t. \tag{15}$$

We note that $R_t \subseteq R$ for all $t \in S$. We also note that when $t_1 \leq t_2$, we have that $R_{t_1} \subseteq R_{t_2}$; that is, the thresholded regions form an increasing (by set inclusion) sequence of subsets of R. For any increasing chain $t_1 < \ldots < t_n$ in S, we have

$$\emptyset \subseteq R_{t_1} \subseteq \cdots \subseteq R_{t_n} \subseteq R. \tag{16}$$

In the MSER approach, the stability of an extremal region R is measured by examining the change in the cardinality of R_t with the change in the threshold t. That is, for a particular threshold $t \in S$ and threshold step $\delta \in S$, such that $t - \delta, t + \delta \in S$, the rate of growth of the extremal region R is given by

$$G_\delta(R; t) = \frac{|R_{t+\delta} \setminus R_{t-\delta}|}{|R_t|}. \tag{17}$$

An extremal region R_{t_0} is then said to be maximally stable if $G_\delta(R; t)$ has a local minimum at $t = t_0$. Such a thresholded region experiences minimal change (in cardinality) when the threshold t_0 is increased/decreased by δ.

Spatial Occupancy

Suppose that D is the domain of an image $I : D \to S \subset \mathbb{R}$. Suppose that $R \subseteq D$ is a region and that $B(R)$ it its corresponding bounded box, with width w_R and height h_R. We now describe how we partition $B(R)$ into a number of rectangular bins which are symmetric about the midpoint of $B(R)$. Let $k_x, k_y \in \mathbb{N}$ be the number of x and y bins respectively. To simplify the following expressions, we employ the coordinate transformation $(x, y) \mapsto (x', y') = (x - x_R, y - y_R)$, for which $B(R) = [0, w_R] \times [0, h_R]$. We partition the x' component of $B(R)$, $[0, w_R]$, by the chain

$$0 = x_0 < x_1 < \cdots < x_{k_x - 1} < x_{k_x} = w_R, \tag{18}$$

where

$$s_x = \left\lceil \frac{w_R}{k_x} \right\rceil, \quad c_x = \left\lfloor \frac{w_R - (k_x - 2)s_x}{2} \right\rfloor, \quad x_n = \begin{cases} 0 & \text{if} \quad n = 0 \\ c_x + (n-1)s_x & \text{if} \quad 1 \le n \le k_x - 1 \\ w_R & \text{if} \quad n = k_x \end{cases}. \tag{19}$$

We similarly partition the y' component of $B(R)$ as above, replacing w_R by h_R, and variables subscripted by x with y. Each partition (or bin) of $B(R)$ is then of the form

$$B_{i,j} = [x_i, x_{i+1}] \times [y_j, y_{j+1}] \quad \text{for all} \quad 0 \le i \le k_x - 1, \quad 0 \le j \le k_y - 1. \tag{20}$$

We note that while $B_{i,j}$ is the Cartesian product of real intervals, $B_{i,j} \cap \mathbb{N}^2$ is the set of pairs of natural numbers which lie within these intervals. Using this partition of $B(R)$, we now define the spatial occupancy $v(R) = (v_{i,j}(R)) \in \mathbb{R}^{k_x \times k_y}$ of R as

$$v_{i,j}(R) = \frac{|R \cap (B_{i,j} \cap \mathbb{N}^2)|}{|B_{i,j} \cap \mathbb{N}^2|} \quad \text{for all} \quad 0 \le i \le k_x - 1, \quad 0 \le j \le k_y - 1; \tag{21}$$

that is, the fraction of discrete points in $B_{i,j}$ that are also in R. We note that $v_{i,j}(R) \in [0, 1]$ for all $0 \le i \le k_x - 1$, $0 \le j \le k_y - 1$.

3.5 Task 1

We describe the detection algorithm for task 1 in detail.

- We begin with the supplied colour image, $I_C : D \to \{0, \ldots, 255\}^3$, for which we are to detect a sign with three digits. We denote the red, green, and blue channels of this image by $I_R, I_G, I_B : D \to \{0, \ldots, 255\}$.
- We create a grayscale transformation of this image, $I : D \to \{0, \ldots, 255\}$.
- We use the MSER method, with the following parameters:
  - minimum region size of 45;
  - maximum region size of 2000;
  - threshold step, $\delta = 20$.

  Applying the MSER algorithm with these parameters to the grayscale image I yields a set of maximally stable extremal regions $\mathcal{R} = \{R_1, \ldots, R_n\}$.
- The MSER algorithm can yield nested regions, or strongly overlapping regions, which interfere with the construction of chains of regions. Hence, we order the regions by decreasing area and filter out any regions which strongly overlap with a larger region. That is, we order $\mathcal{R}$ such

that for all $R_i, R_j \in \mathcal{R}$ we have $|R_i| \geq |R_j|$ if $i \leq j$. Then, for each $R_i \in \mathcal{R}$, for all $R_j \in \mathcal{R}$ with $j \geq i$, we remove R_j from $\mathcal{R}$ if

$$\frac{|R_j \cap R_i|}{|R_j|} \geq \lambda.$$

In testing, we have found $\lambda = 0.8$ to be suitable (although this could probably be much stricter).

- We then make use of the regularity of the aspect ratio of the bounding boxes of the digits to further filter $\mathcal{R}$. This step is effective at removing any regions corresponding to bricks (which tend to be wider than they are tall) which can be numerous, and thus impact the performance of the algorithm. For each region $R_i \in \mathcal{R}$ with corresponding bounding box $B(R_i)$, with aspect ratio $a_i = w_{R_i}/h_{R_i}$, we remove R_i from $\mathcal{R}$ if $a_i \notin [a_{\text{min}}, a_{\text{max}}]$. In testing, we have found $a_{\text{min}} = 1 : 1.2$ and $a_{\text{max}} = 1 : 3$ to be suitable.

- A consequence of the MSER algorithm is that it detects the interior black holes of digits - such as those in 0, 6, 8, 9 - as extremal regions in addition to the white digits regions. These regions can also interfere with the construction of chains of regions, and so we filter them out. We make use of the fact that the bounding box of a region, that is contained within another region, will be contained in the bounding box of that region also, and that these interior regions we wish to remove are simply connected (they have no holes of their own). However, we wish to remove only interior regions which are closely packed into another region - and avoid accidentally removing say a character region which may possibly be inside a region corresponding to the sign. Hence, for each region $R_i \in \mathcal{R}$ we remove any region $R_j \in \mathcal{R}$ from $\mathcal{R}$ if

$$B(R_j) \subseteq B(R_i) \quad \text{and} \quad \max_{q \in \partial R_j} \left(\min_{p \in \partial R_i} \text{dist}(p, q) \right) \leq \mu.$$

  That is, we only remove R_j if the maximum distance between the boundaries of R_i and R_j is less than μ. In testing, we have found $\mu = 10$ to be suitable (however, this should probably be modified to be scale invariant).

- We then make use of the regularity of the fill of the digits regions to further filter $\mathcal{R}$, by removing any regions which fill their bounding box above a certain point. This step is effective at removing any vertically oriented rectangles which might otherwise be possibly mistaken for digits. That is, for each region $R_i \in \mathcal{R}$, with corresponding bounding box $B(R_i)$, we remove R_i from $\mathcal{R}$ if

$$\frac{|R_i|}{|B(R_i) \cap \mathbb{N}^2|} \geq \xi.$$

  In testing, we have found $\xi = 0.85$ to be suitable.

- Having now filtered the regions quite heavily, we consider the dimensional similarity and geometric adjacency between regions to link sequences of similar, collinear regions into chains. We make use of the fact that the most reliable geometric property of the digits, which is least varying with perspective transformations, is the height of the bounding boxes of the digits. For all regions $R_i, R_j \in \mathcal{R}$:

- We consider their height to be similar if

  $$\left|\frac{h_{R_i} - h_{R_j}}{h_{R_i}}\right| \leq \zeta_h \quad \text{and} \quad \left|\frac{h_{R_i} - h_{R_j}}{h_{R_j}}\right| \leq \zeta_h.$$

  In testing, we have found that $\zeta_h = 0.2$ to be suitable.

- We consider their y placement to be similar if

  $$\left|\frac{y_{R_i} - y_{R_j}}{h_{R_i}}\right| \leq \zeta_y \quad \text{and} \quad \left|\frac{y_{R_i} - y_{R_j}}{h_{R_j}}\right| \leq \zeta_y.$$

  In testing, we have found that $\zeta_y = 0.5$ to be suitable.

- We consider them to be x adjacent if

  $$\left|\frac{x_{R_i} - x_{R_j}}{h_{R_i}}\right| \leq \zeta_x \quad \text{and} \quad \left|\frac{x_{R_i} - x_{R_j}}{h_{R_j}}\right| \leq \zeta_x,$$

  and if the overlap of their boxes satisfies

  $$\frac{|B(R_i) \cap B(R_j)|}{|B(R_i)|} \leq \zeta_B \quad \text{and} \quad \frac{|B(R_i) \cap B(R_j)|}{|B(R_j)|} \leq \zeta_B.$$

  In testing, we have found that $\zeta_x = 1.0$ and $\zeta_B = 0.25$ to be suitable.

If R_i and R_j are found to satisfy the above properties, we say that they are linked. We first find all links between all regions in $\mathcal{R}$, and then we filter these links - by minimal region-to-region distance - to ensure that each region is only linked to at most one region to its left and to at most one region on its right. As a result, all paths extracted from the graph formed by these regions and their edges will be non-overlapping. We extract all paths (of at least 2 regions) and refer to these paths as chains; that is, the chains are of the form $\mathcal{C}_i = \{R_{i_1}, R_{i_2}, \ldots, R_{i_{n_i}}\}$.

3.6 Task 2

4 Validation Performance

4.1 Task 1

4.2 Task 2

5 Conclusion

References

[Matas et al., 2004] Matas, J., Chum, O., Urban, M., and Pajdla, T. (2004). Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767. British Machine Vision Computing 2002.

A Source Code