THESIS TITLE:

VEHICLE LICENSE PLATE DETECTION AND RECOGNITION

Committee Members:

He, Zhihai

Tony, Han

Duan, Ye

Defence:

Ning, Guanghan

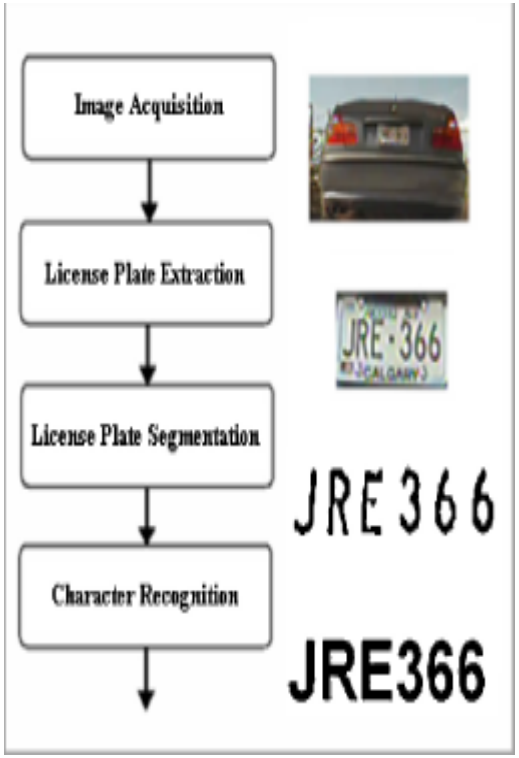
E-mail:
gnxr9@mail.missouri.edu
Cell-phone:
(573)825-8230

Overview or Column:

- 1. License Plate Detection and Recognition: What and Why
- 2. Dataset
- 3. Challenges
- 4. Previous Related Work
- 5. Proposed Method
- 6. The Contribution
- 7. Demo

1. PROBLEM: LICENSE PLATE DETECTION AND RECOGNITION

ALPR SYSTEM: SEVERAL STAGES



1. PROBLEM: LICENSE PLATE DETECTION AND RECOGNITION

APPLICATIONS

1. Automatic toll collection
2. Traffic law enforcement
3. Parking lot access control
4. Road traffic monitoring

2. DATASET

- Images

1600*1200

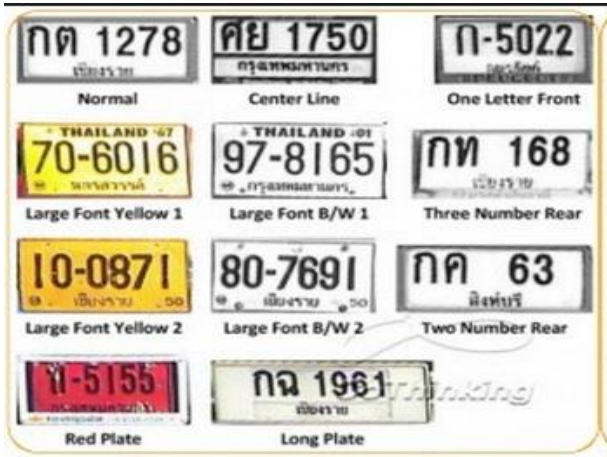
1920*1104

All resize to 909*523

3.CHALLENGES

(1) PLATE VARIATION

- 1. location
- 2.quantity
- 3. size
- 4.color
- 5.font
- 6.standard versus vanity
- 7.occlusion
- 8.inclination
- 9.plates contain frames and screws



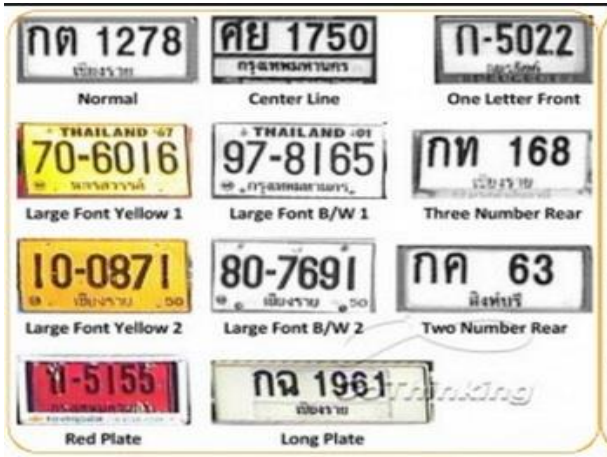
Thailand License Plate

Vietnam License Plate

3.CHALLENGES

(1) PLATE VARIATION

- *1. location*
- *2.quantity*
- *3. size*
- *4.color*
- 5.font
- 6.standard versus vanity
- *7.occlusion*
- *8.inclination*
- *9.plates contain frames and screws*



Thailand License Plate

Vietnam License Plate

3.CHALLENGES
(2) ENVIRONMENT VARIATION

- 1. illumination
- 2.background

4.Previous Related Work

License plate extraction:

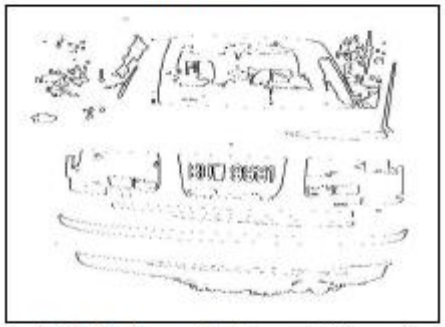
1.using edge information

- 1).Edge Detection 2). Filters
- 3). Transforms 4). Block-based
- Typical method: VEDA

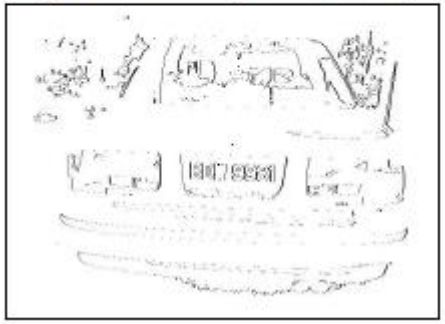
```
Create a white blank image as Img(x,y);
// sum_{x=0}^{height} sum_{y=0}^{width} Img(x,y) = 255 ;
For every pixel in ULEA image output
  center=1; left=1; right=1;

  If(four center mask values = black)
        center=0;
  End if
  If(two right mask values = black)
        right=0;
  End if
  If(two left mask values = black)
        left=0;
  End if

  If(!center AND !right AND !left)
    two center column values of Img =white;
    /*
      Img(x,y)=255;
      Img(x+1,y)=255;
    */
  End if
End for
```

Algorithm2: VEDA pseudocode

(a) Input image



(b) Sobel's result (time=130msec)

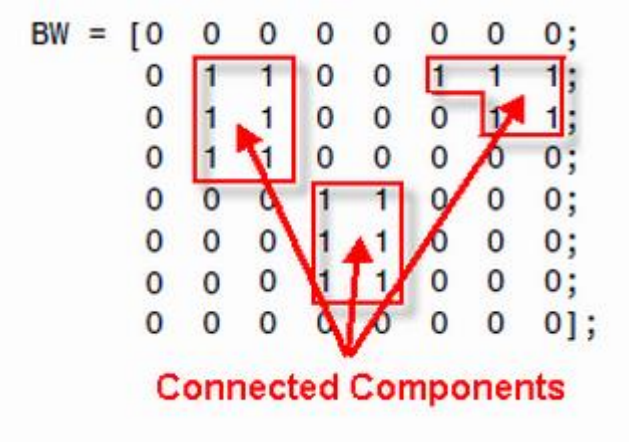


(c) Our algorithm's result (time=15msec)

4. Previous Related Work

LICENSE PLATE EXTRACTION: 2.USING GLOBAL IMAGE INFORMATION

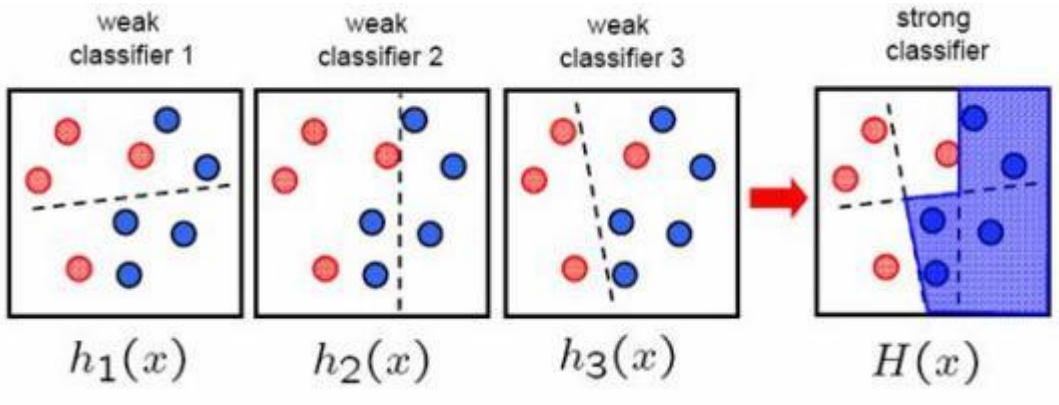
- 1). Connected Component Analysis
- 2). 2-D cross correlation with pre-stored license plate template

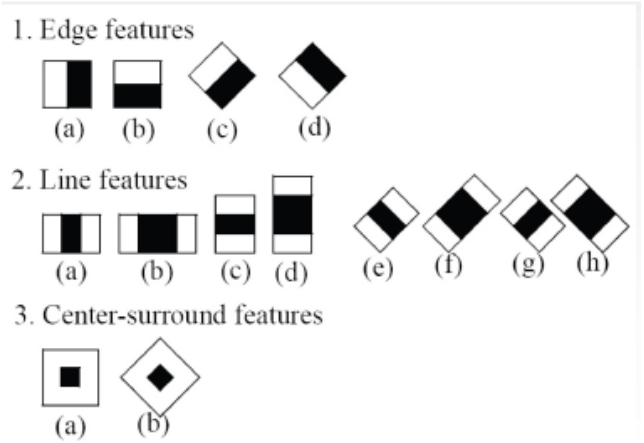


4.Previous Related Work

LICENSE PLATE EXTRACTION: 3.USING TEXTURE FEATURES

- 1). Vector Quantization
- 2). Sliding Concentric Window
- 3). Image Transformations For Textures: DFT, WT
- 4). Adaboost + Haar-like features





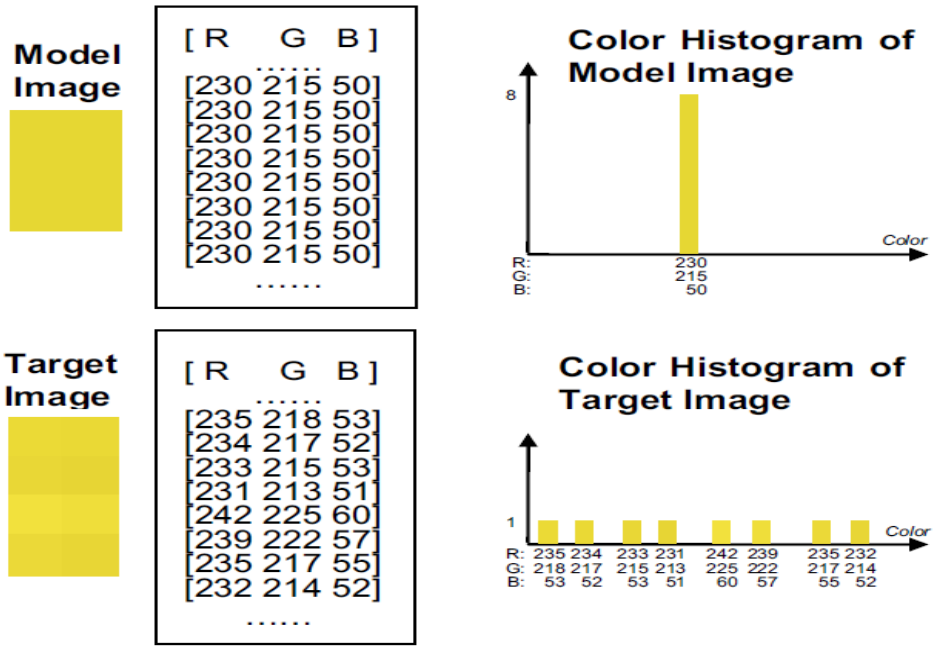
Haar features

4.Previous Related Work

License plate extraction:

4.using color features

- HLS(HIS), HSV color space(or RGB …)
- 1). Neural Network to Classify the Color of Each Pixel
- 2). Genetic Algorithm to Search License Plate Color
- 3). Gaussian Weighted Histogram Intersection
- 4). Fuzzy Logic Based(Deal With Illumination Variance)

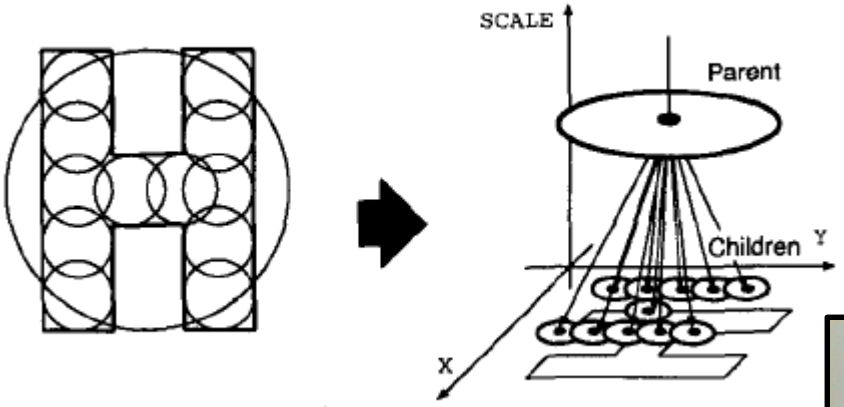


4.Previous Related Work

License plate extraction:

5.Using character features

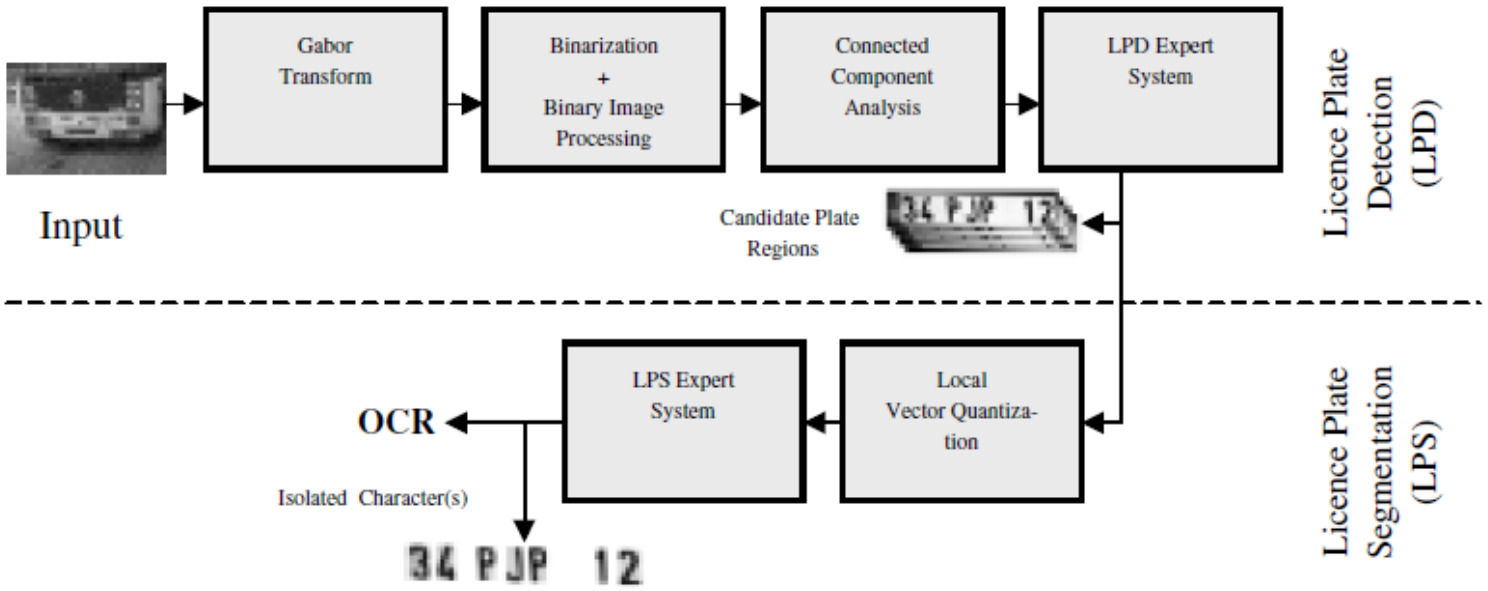
- 1). Repeating contrast changes
- 2). Same aspect ratio as characters
- 3). SVM-trained SIFT descriptors
- 4). MSER



Scale change as feature

4. Previous Related Work

LICENSE PLATE EXTRACTION: 6.COMBINING TWO OR MORE FEATURES



4.Previous Related Work

LICENSE PLATE SEGMENTATION:

1. USING PIXEL CONNECTIVITY

- Labeling the connected pixels in the binary license plate image
- The labeled pixels are analyzed and those which have the same size and aspect ratio of the characters are considered license plate characters.

What about joined or broken?

4.Previous Related Work

License plate segmentation:
2. using projection profiles

Color information!

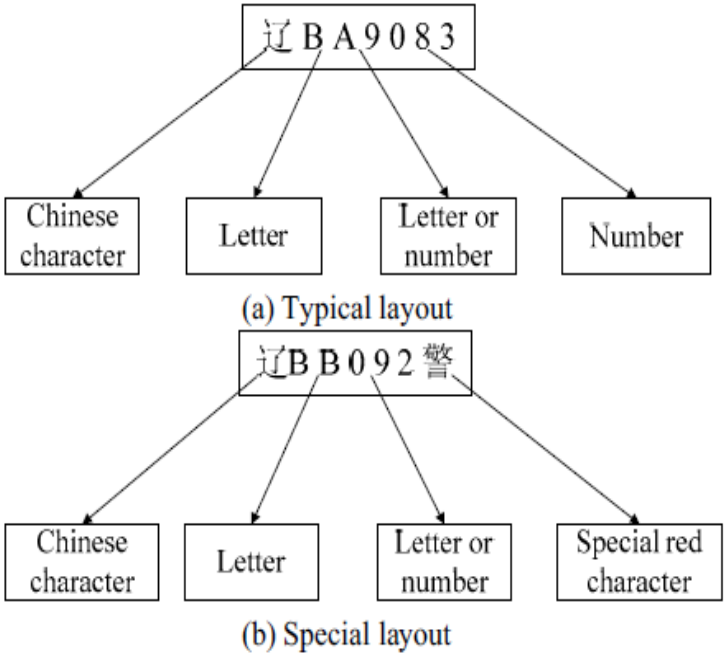
- First, project the binary extracted license plate vertically to determine starting and ending positions of the characters.
- Second, project the extracted characters horizontally to extract each character alone.

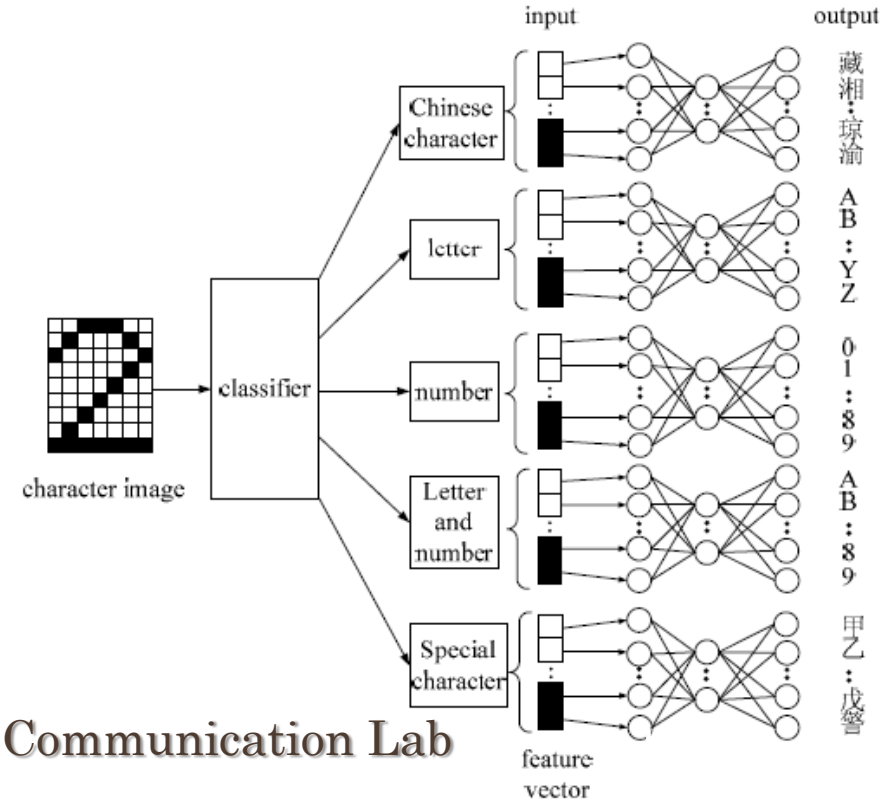
4. Previous Related Work

License plate segmentation:

3. Using prior knowledge of characters

- 1. number of the characters
- 2. colors of the background and characters





4.Previous Related Work

LICENSE PLATE SEGMENTATION: 4. USING CHARACTER CONTOURS

4.Previous Related Work

License plate segmentation: Comparison:

Methods	Pros	Cons
Using pixel connectivity	Simple and straightforward, robust to the license plate rotation.	Fails to extract all the characters when there are joined or broken characters.
Using projection profiles	Independent of character positions, be able to deal with some rotation.	Noise affects the projection value, requires prior knowledge of the number of license plate characters.
Using prior knowledge of characters	Simple.	Limited by the prior knowledge, any change may result in errors.
Using character contours	Can get exact character boundaries.	Slow and may generate incomplete or distorted contour.
Using combined features	More reliable.	Computationally complex.

4.Previous Related Work

CHARACTER RECOGNITION:

1.USING RAW DATA

- Template matching
- Normalized cross correlation to match extracted characters with templates
- Store templates of the same character with different inclination angles

Suit for single-font, non-rotated, non-broken, and fixed-size.
It is simple but limited.

4.Previous Related Work

CHARACTER RECOGNITION

2.USING EXTRACTED FEATURES

- 1. Raw
- 2. Ratio
- 3. Symmetry
- 4. Correlagram
- 5. SIFT
- ……..

4.Previous Related Work

Result comparison of different methods

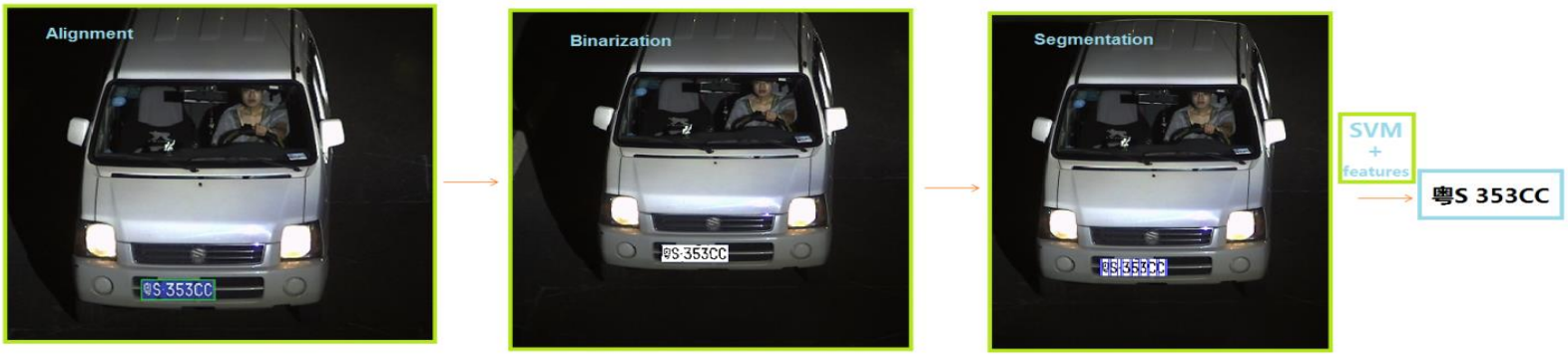
- No uniform evaluation way yet.

Methods	Pros	Cons
Using pixel values	Simple and straightforward.	Processing nonimportant pixels and slow, vulnerable to any font change, rotation, noise and thickness change.
	Be able to recognize tilted characters.	More processing time.
Using extracted features	Be able to extract salient features, robust to any distortion, fast recognition since the number of features is smaller than that of the pixels.	Feature extraction takes time, nonrobust features will degrade the recognition.

5.Proposed Method

Overview: which is my work?

(a) License Plate Detection

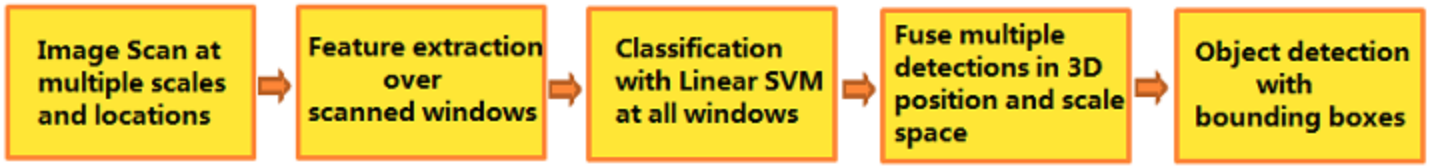


(b) License Plate Recognition

5.Proposed Method

License Plate Detection

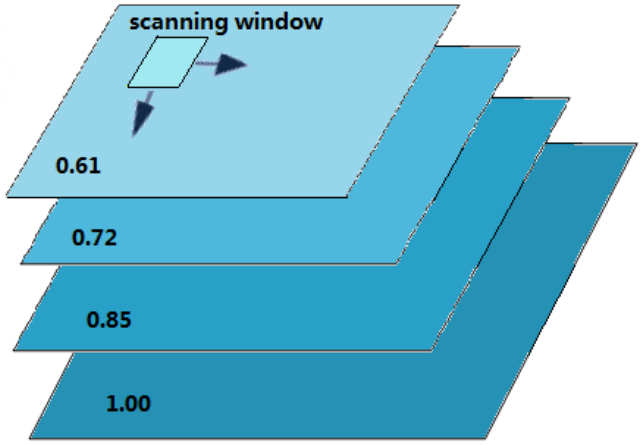
Several components:



Scanning Window
SVM
HOG
NMS
Speed-up

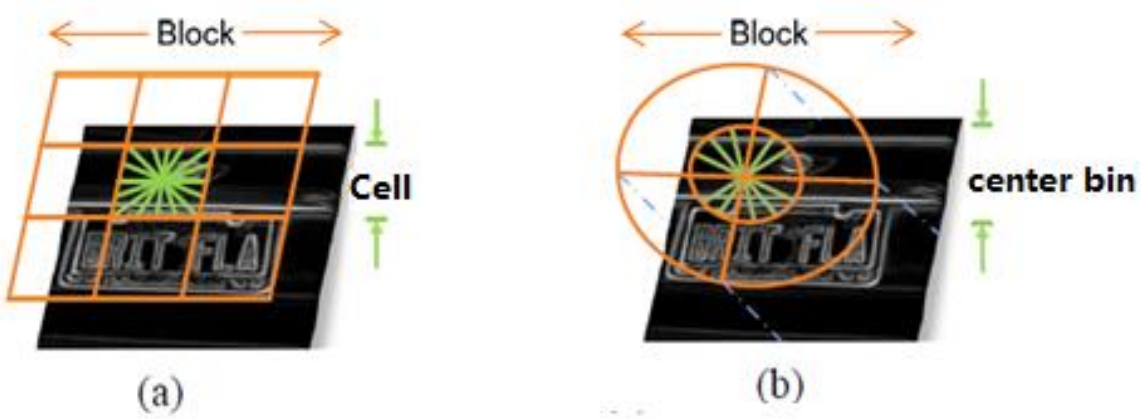
5.Proposed Method
License Plate Detection

Scanning Window



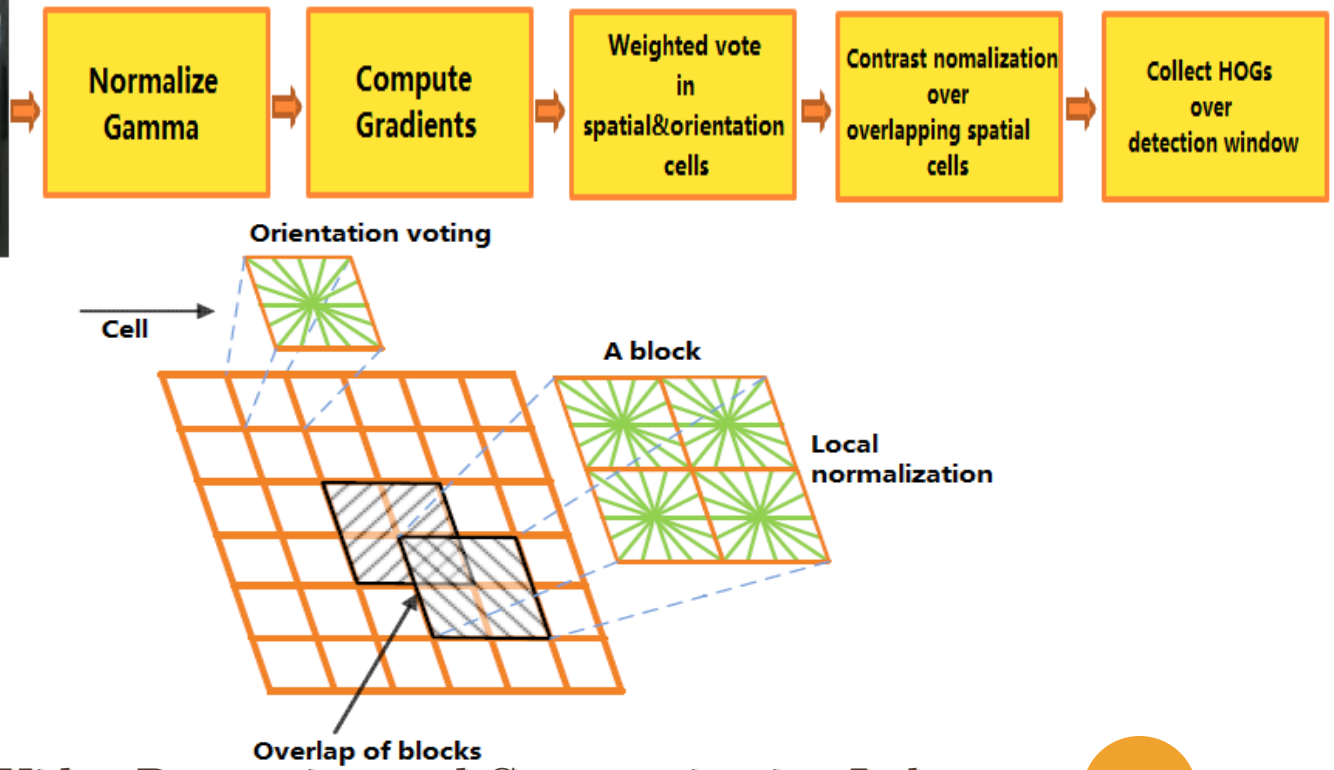
5.Proposed Method
License Plate Detection

HOG Features: Types



5.Proposed Method
License Plate Detection

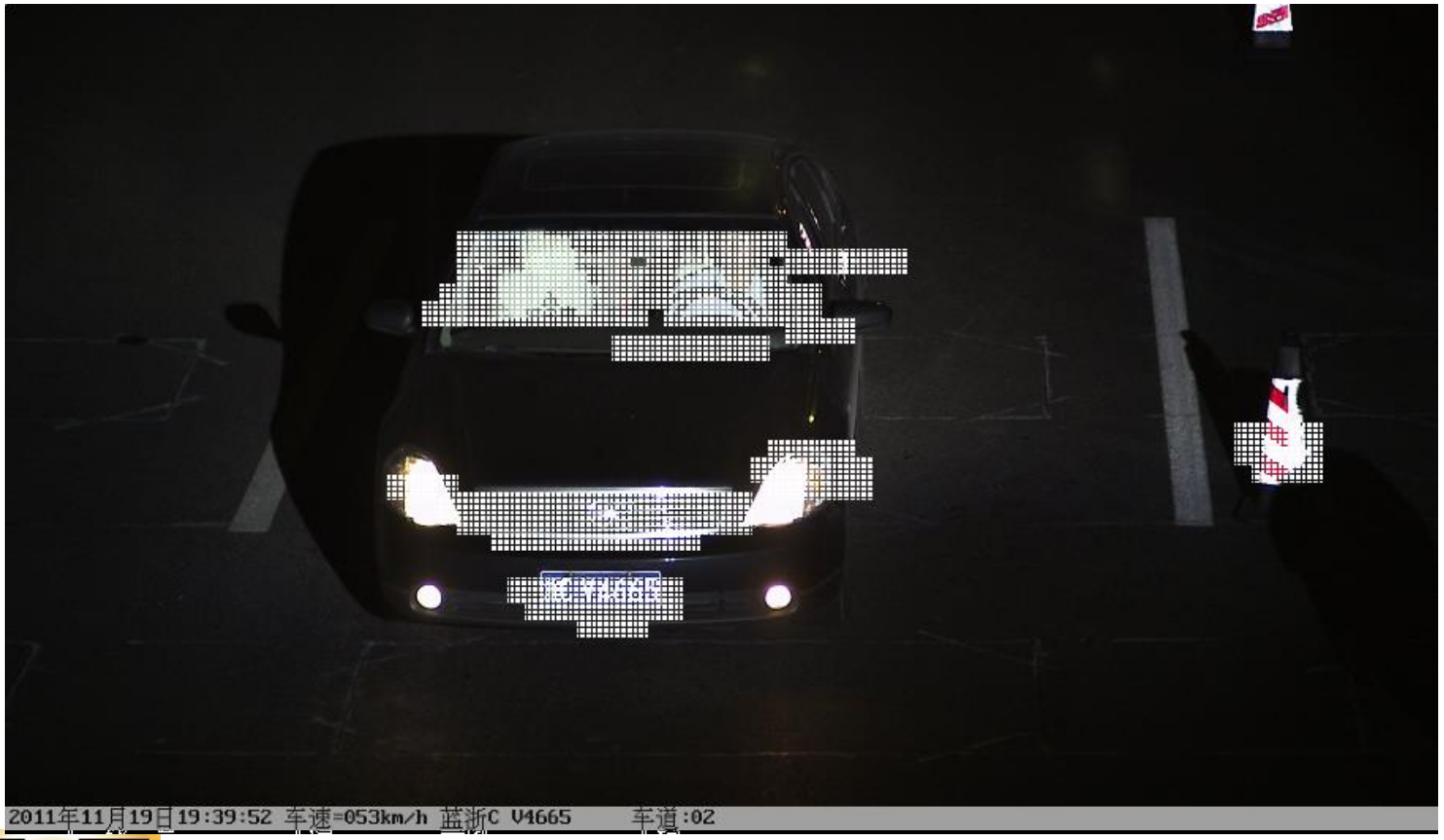
HOG Features: How to extract



Mizzou
University of Missouri

5.PROPOSED METHOD
LICENSE PLATE DETECTION

- Speed Up Using Edge Information



5.PROPOSED METHOD

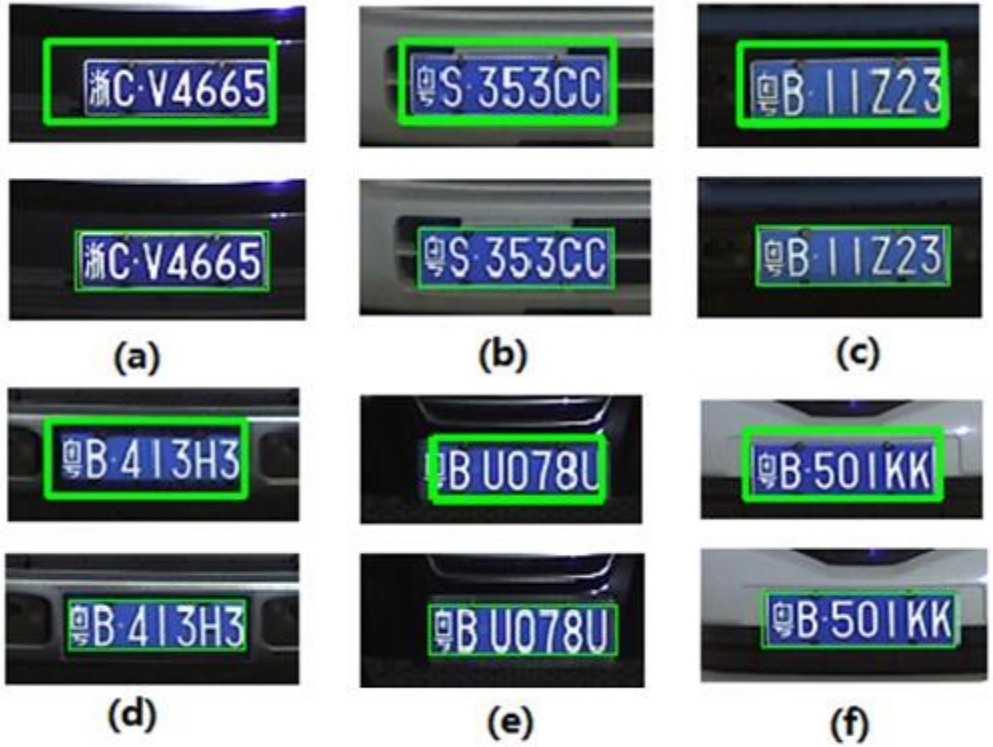
LICENSE PLATE RECOGNITION

OVERVIEW

5.PROPOSED METHOD

LICENSE PLATE RECOGNITION

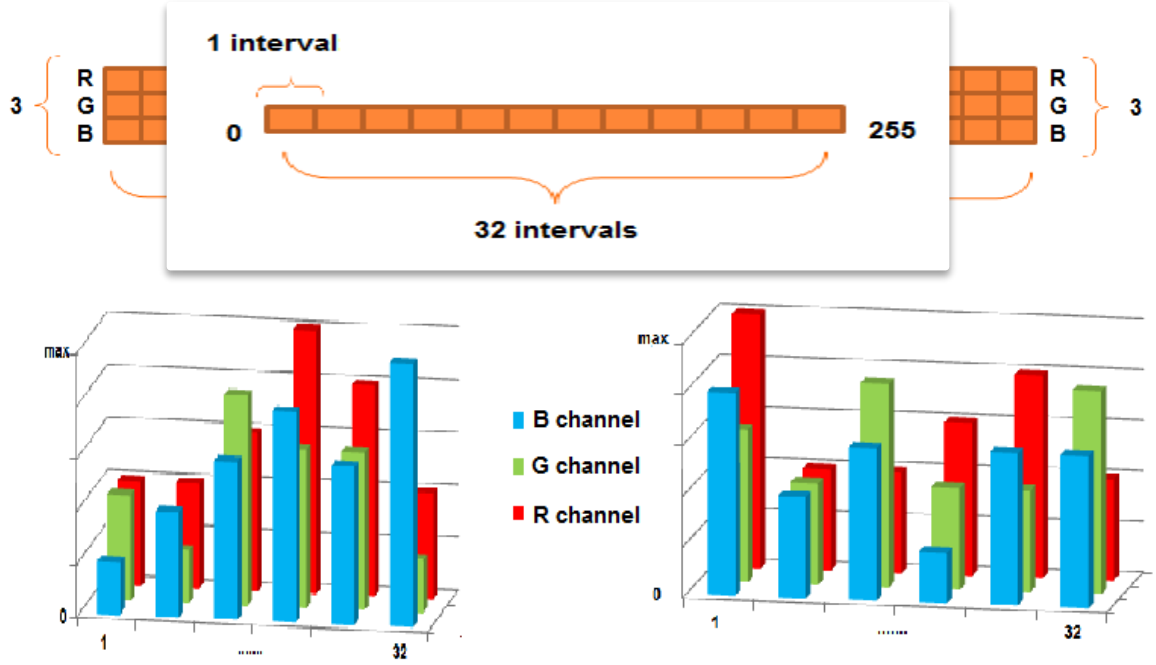
GLOBAL ALIGNMENT: RESULTS



5.PROPOSED METHOD

LICENSE PLATE RECOGNITION

GLOBAL ALIGNMENT: METHOD



5.PROPOSED METHOD

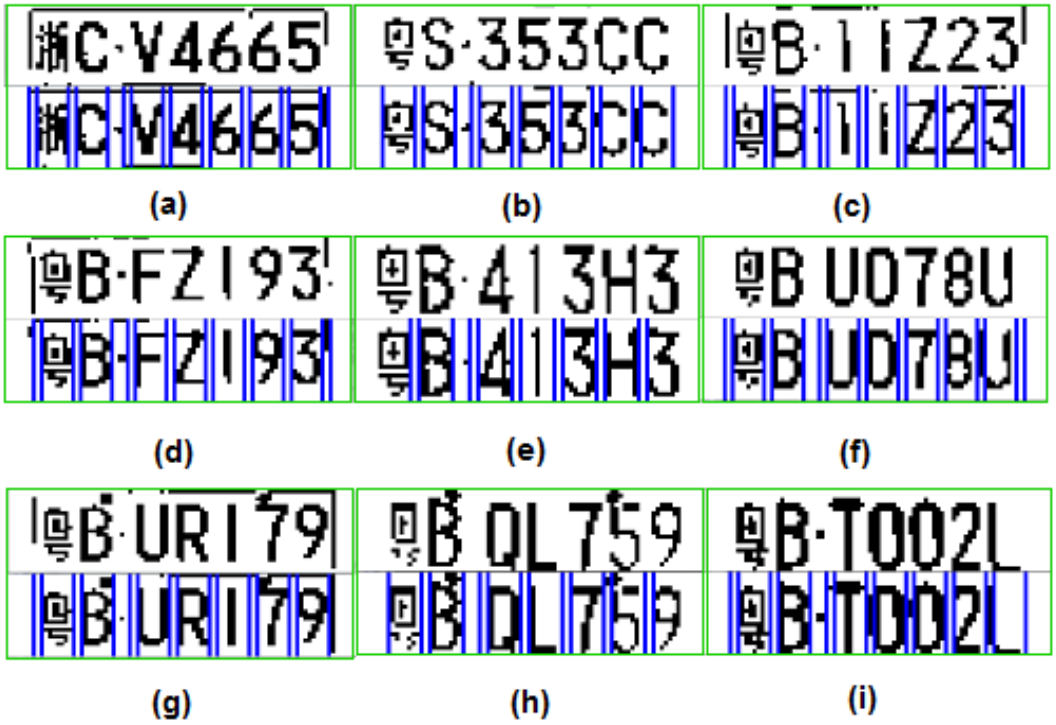
LICENSE PLATE RECOGNITION

BINARIZATION: K-MEANS

5.PROPOSED METHOD

LICENSE PLATE RECOGNITION

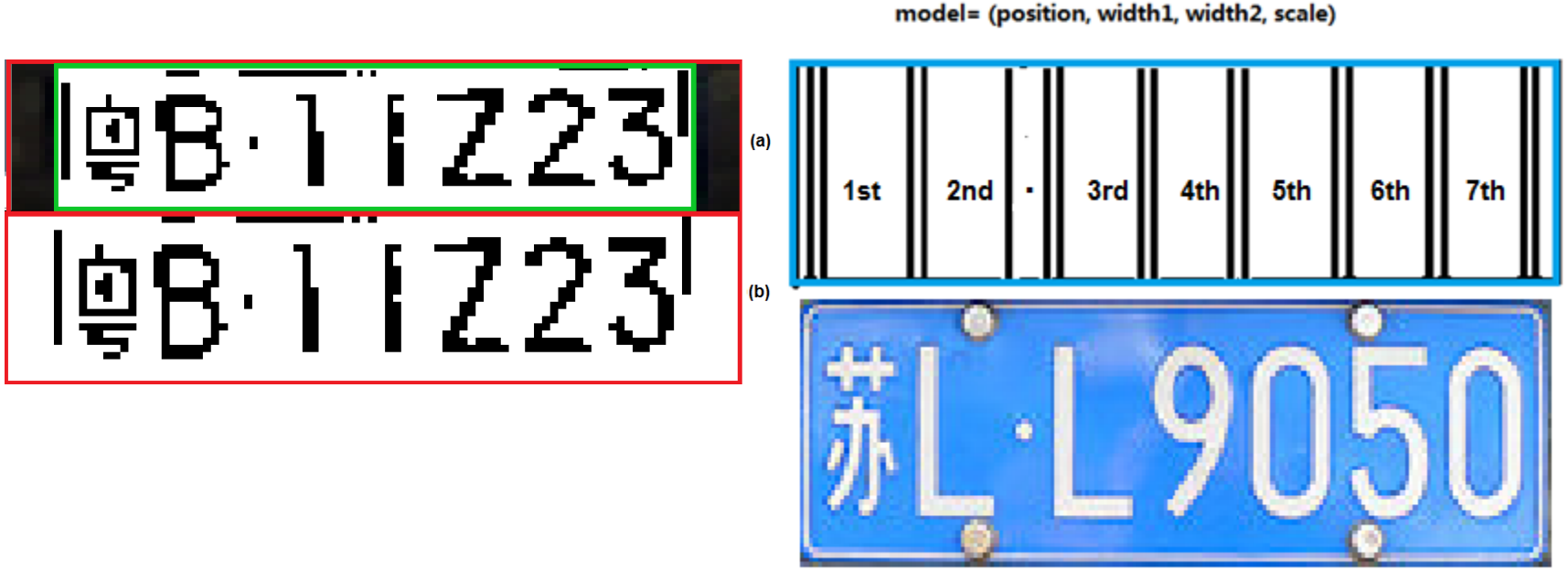
SEGMENTATION: RESULTS



5.PROPOSED METHOD

LICENSE PLATE RECOGNITION

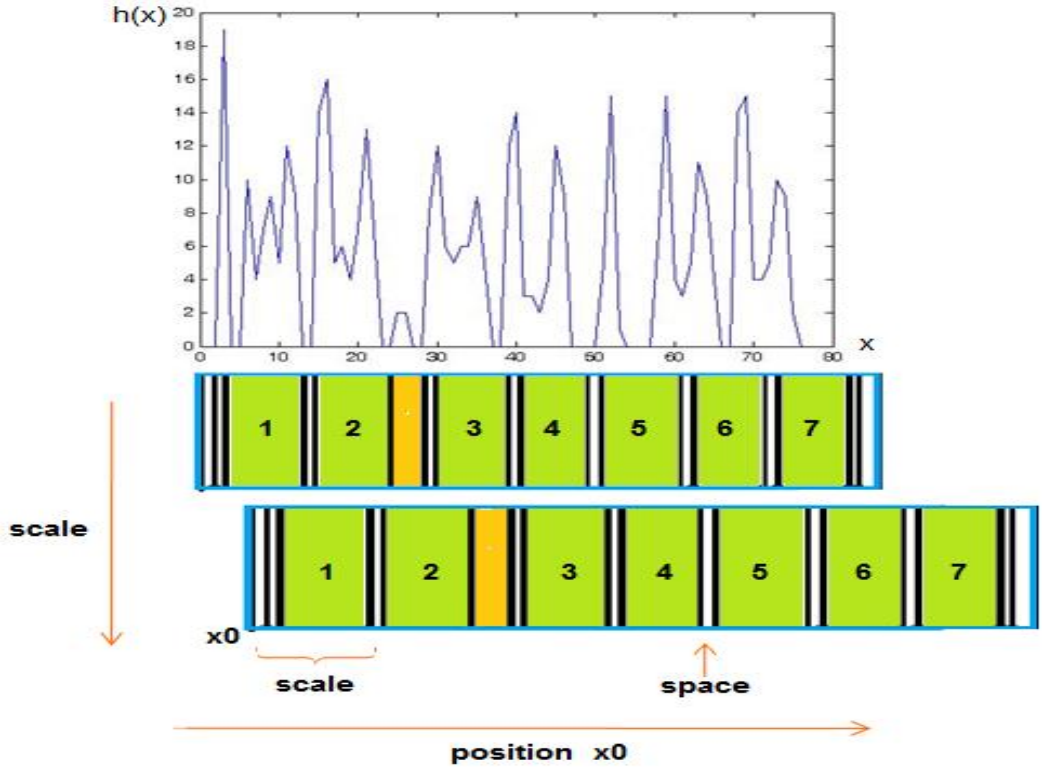
SEGMENTATION: MODEL



5.PROPOSED METHOD

LICENSE PLATE RECOGNITION

SEGMENTATION: OPTIMIZATION



(c)

5.PROPOSED METHOD
LICENSE PLATE RECOGNITION

CHARACTER RECOGNITION: FEATURES

- (1). The histogram of vertical 0-1 inversion pattern. 16 dimensional
- (2). The histogram of horizontal 0-1 inversion pattern. 16 dimensional
- (3). The histogram of 0/1 ratio vertically. 16 dimensional
- (4). The histogram of 0/1 ratio horizontally. 16 dimensional
- (5). The raw feature. 16*32 dimensional

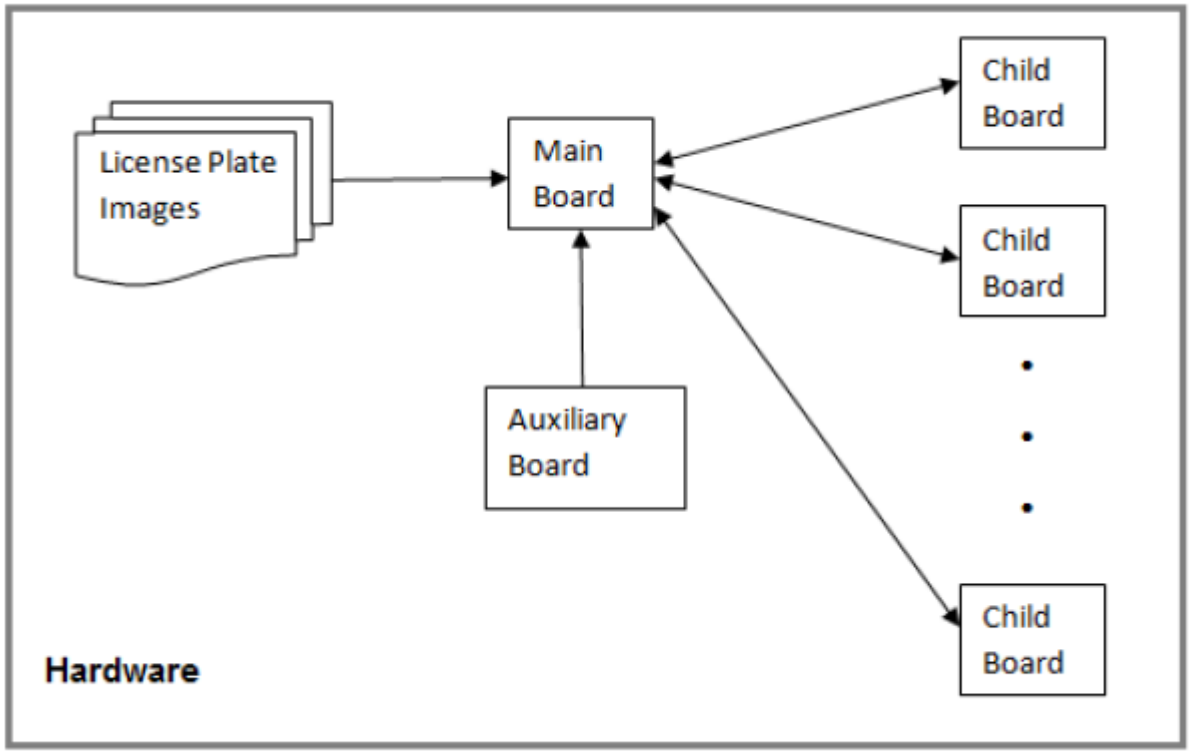
5.PROPOSED METHOD
LICENSE PLATE RECOGNITION

CHARACTER RECOGNITION: RESULTS

5.PROPOSED METHOD

EMBEDDED SYSTEM

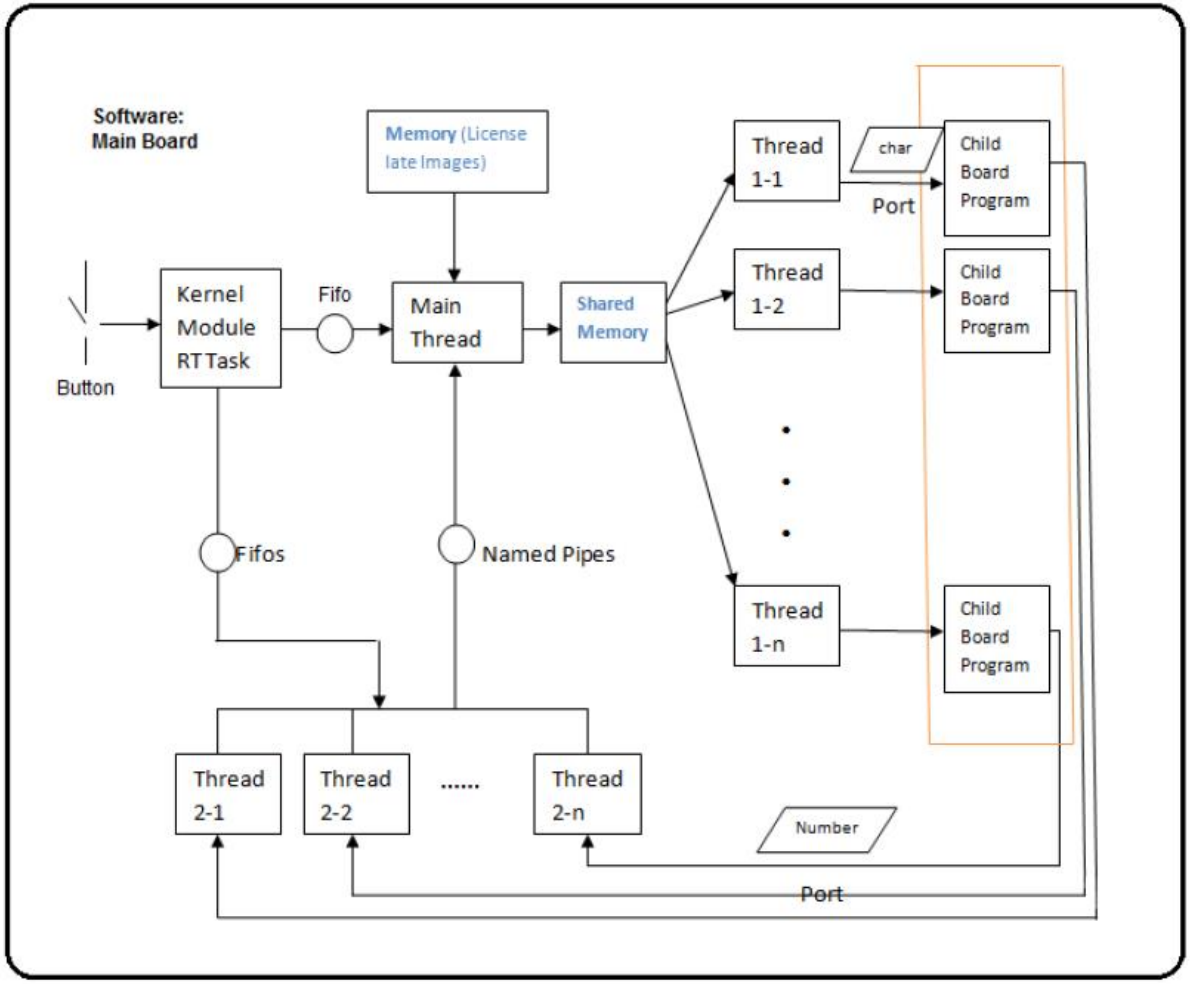
HARDWARE



5.PROPOSED METHOD

EMBEDDED SYSTEM

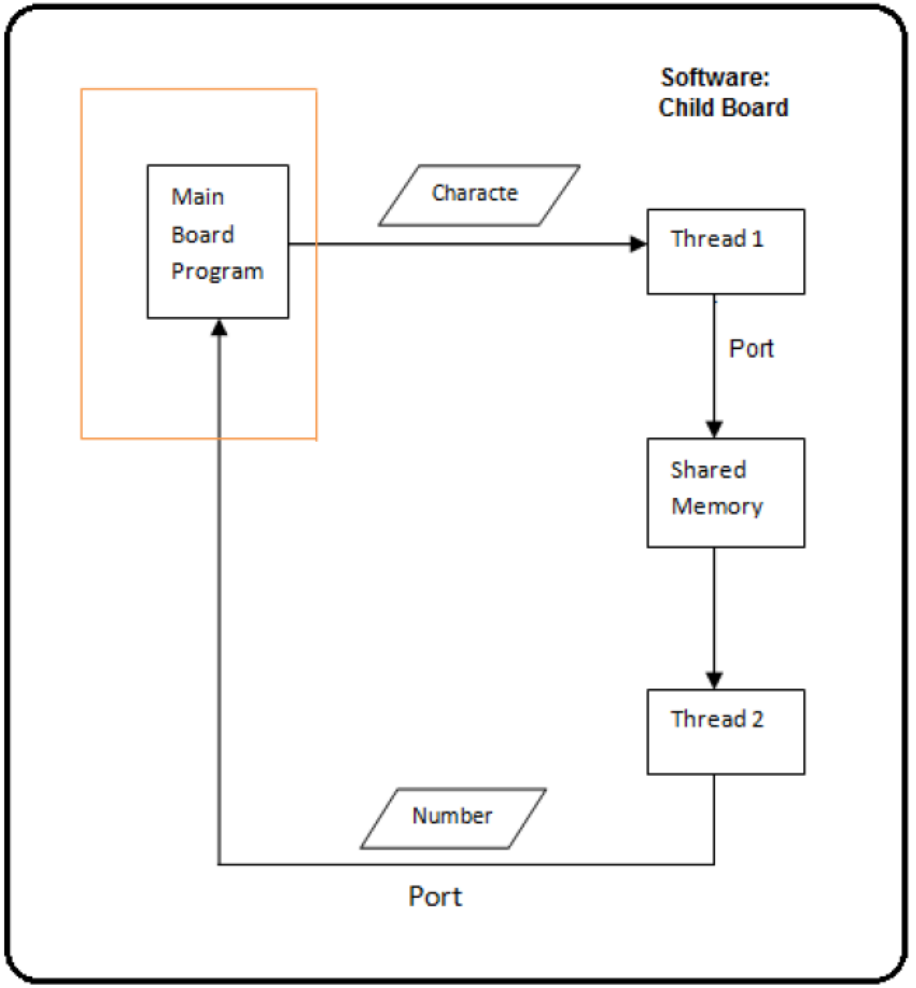
SOFTWARE: MAIN BOAR



5.PROPOSED METHOD

EMBEDDED SYSTEM

SOFTWARE: CHILD BOARD



6. THE CONTRIBUTION
LICENSE PLATE DETECTION AND RECOGNITION

- 1. The Global Alignment
- 2. Segmentation Model
- 3. Time saving for sliding window method
- 4. A framework on embedded system which can be used for future generic object recognition or even other applications:
- EDA or GA for deformable model
- Deep learning

7. DEMO
LICENSE PLATE DETECTION AND RECOGNITION

1. 150~200ms on small stride, 4 scales(night dataset)
   - 100~150ms on larger stride
   - around 50ms on single scale
2. Embedded System

THANK YOU!