applied sciences

MDPI

Article

Automatic Receipt Recognition System Based on Artificial Intelligence Technology

Cheng-Jian Lin [1,2,*], Yu-Cheng Liu [1] and Chin-Ling Lee [3]

[1] Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taichung 411, Taiwan; simon880525@gmail.com
[2] College of Intelligence, National Taichung University of Science and Technology, Taichung 404, Taiwan
[3] Department of International Business, National Taichung University of Science and Technology, Taichung 404, Taiwan; merrylee@nutc.edu.tw
* Correspondence: cjlin@ncut.edu.tw

Abstract: In this study, an automatic receipt recognition system (ARRS) is developed. First, a receipt is scanned for conversion into a high-resolution image. Receipt characters are automatically placed into two categories according to the receipt characteristics: printed and handwritten characters. Images of receipts with these characters are preprocessed separately. For handwritten characters, template matching and the fixed features of the receipts are used for text positioning, and projection is applied for character segmentation. Finally, a convolutional neural network is used for character recognition. For printed characters, a modified You Only Look Once (version 4) model (YOLOv4-s) executes precise text positioning and character recognition. The proposed YOLOv4-s model reduces downsampling, thereby enhancing small-object recognition. Finally, the system produces recognition results in a tax declaration format, which can upload to a tax declaration system. Experimental results revealed that the recognition accuracy of the proposed system was 80.93% for handwritten characters. Moreover, the YOLOv4-s model had a 99.39% accuracy rate for printed characters; only 33 characters were misjudged. The recognition accuracy of the YOLOv4-s model was higher than that of the traditional YOLOv4 model by 20.57%. Therefore, the proposed ARRS can considerably improve the efficiency of tax declaration, reduce labor costs, and simplify operating procedures.

Keywords: receipt recognition; deep learning; YOLO; handwritten receipt; printed receipt; human machine interface

Citation: Lin, C.-J.; Liu, Y.-C.; Lee, C.-L. Automatic Receipt Recognition System Based on Artificial Intelligence Technology. *Appl. Sci.* **2022**, *12*, 853. https://doi.org/10.3390/app12020853

Academic Editors: Teen-Hang Meen and Chun-Yen Chang

Received: 7 December 2021
Accepted: 7 January 2022
Published: 14 January 2022

1. Introduction

Receipts are vouchers that people obtain through exchanges in daily life. They are also used as the basis for taxation. As the number of transaction receipts between enterprises and individuals increases, the management of receipts becomes more cumbersome. Paper receipts are easily damaged by external environmental impacts, thus rendering their conservation a difficult task. Therefore, the information on a receipt must be manually entered into a computer promptly. For example, for tax declaration, the word track, uniform number, date, and consumption amount on each receipt must be recorded in sequence. Each of such receipts contains a large quantity of data for recording, and the various types of receipts are complex; consequently, they are usually processed manually. Processing personnel classify the receipts and compare the information on each receipt. After checking the information, they record it in the system; this is a time-consuming and labor-intensive task. For tax filing, accounting firms must process tens of thousands of receipts. To save human resources, improve work efficiency, and reduce human errors in tax-filing processes, an automatic system for recognizing receipt information must be developed.

Because manually transferring the receipt information to a computer is a labor-intensive, time-consuming, less efficient, and error-prone process, automatic information recognition systems have been developed to address these limitations. For example, optical

character recognition (OCR) technology [1] has been developed for enabling computers to read text images, extract information, and recognize text. OCR is widely used in various fields, such as identity card recognition [1], automatic license plate recognition [2], and receipt recognition [3]. An OCR system includes the following components: image input, image preprocessing, image correction, text positioning, character segmentation, and character recognition [4].

In recent years, researchers have used OCR systems to retrieve data on receipts. Liu et al. [3] used a single-shot multibox detector neural network to determine information areas on receipts. Shi et al. [4] combined specific signs and template matching on receipts to accurately locate receipt information. Meng et al. [5] applied You Only Look Once, version 3 (YOLOv3) to accurately locate, segment, and intercept key information areas on invoice images.

Some researchers have used machine learning algorithms to identify text content. These algorithms extract features from the text content and then use machine learning to classify the content. For example, Wang et al. [6] used a Gabor filter for character recognition in grayscale images. Cui and Ruan [7] used a Gabor filter for feature extraction and a support vector machine (SVM) algorithm for gray character recognition. Furthermore, Xie [8] employed dynamic thresholding to segment characters and used gray value normalization to extract features; a least-squares SVM was then used to classify characters based on the extracted features. Hazra et al. [9] adopted OCR to extract distinct features from input images and used k-nearest neighbor classification to recognize handwritten or printed text. Liu et al. [3] combined a convolutional neural network (CNN) and gated recurrent unit network for character recognition. Meng et al. [5] used a connectionist text proposal network to detect text blocks and adopted densely connected convolutional networks to recognize the detected text. Smith [10] used a CNN and long short-term memory to detect and recognize text in images. Liu [11] proposes an automatic taxi receipt text recognition application system for character recognition. In the recognition system, a Single Shot Multi-Box Detector (SSD) neural network is used to determine the data region then a CNN is designed for character recognition. Xie [12] designed a deep learning recognition method to extract effective information from receipts. The proposed connectionist text proposal network (CTPN) is used to locate the text region in the receipt, then the convolutional recurrent neural network (CRNN) is adopted to convert the receipt from an image into text. Liu [13] used YOLO V3 to identify the text from the triplicate uniform invoice and solve the cumbersome and error-prone problem of manual processing. Nguyen [14] compared the effectiveness of towards document image quality assessment (TDIQA), Yolov5, seq2seq, and transformer in invoice identification. They found that yolov5 has the highest accuracy among these methods, but it also requires more training time. The methods mentioned above recognize the computer-printed text in the receipt. If you encounter a handwritten receipt, the accuracy will drop significantly.

To improve operational efficiency and reduce human errors in the processing of receipt information for tax declaration tasks, the present study developed an automatic receipt recognition system (ARRS). The proposed ARRS involves the following operations: receipt image reading, data preprocessing, text positioning, character segmentation, character recognition, review and error correction, and database upload. In Taiwan, receipts can be classified into two categories: printed and handwritten receipts. In the proposed system, these two types of receipts are processed separately, and their images are preprocessed. The system operates as follows: First, a high-resolution image of the receipt is created using a scanner. Subsequently, the receipt is automatically classified as a printed or handwritten receipt according to the features of its characters. For a handwritten receipt, template matching is applied, and the fixed features of the receipt are used for text positioning. Subsequently, character segmentation is executed through a projection method. A CNN is then used to perform character recognition. For a printed receipt, a YOLOv4-s model is used for precise positioning of the text and character recognition. The YOLOv4-s model reduces downsampling, thereby enhancing the ability to recognize small objects. Next,

after the characters on the two types of receipts are recognized, the results are uploaded to a database, and interface review and error correction are performed. Finally, the system provides an output in a tax declaration format, which can be uploaded to a tax declaration system. The proposed ARRS can substantially improve the efficiency of tax declaration, reduce labor costs, and simplify processing procedures. The contributions of this study are summarized as follows:

- This study developed an automatic receipt recognition system (ARRS) for recognizing printed and handwritten receipts.
- The ARRS includes receipt image reading, data preprocessing, text positioning, character segmentation, character recognition, review and error correction, and database upload function, which can help users quickly convert paper receipts into electronic files.
- To improve the ability to recognize small objects, a YOLOv4-s model is developed in this study.
- The proposed ARRS system provides a human–machine interface to enable users to review the results and perform error correction.

The remainder of this paper is organized as follows. Section 2 provides an overview of the proposed ARRS. Section 3 presents the experimental results and a comparison of a previously proposed YOLOv4 model with the YOLOv4-s model. Finally, Section 4 provides the conclusion.

2. Proposed ARRS

This section explains the operations of the proposed ARRS. First, the receipt information is read by a scanner, and the receipt image is then preprocessed. Next, the receipt is analyzed and classified as handwritten or printed according to the features of its characters. For a handwritten receipt, the information is obtained through receipt positioning, projection and segmentation, and CNN-based text recognition. For a printed receipt, the YOLOv4-s model is used to directly execute character recognition. Figure 1 illustrates the flowchart of the ARRS.

2.1. Image Preprocessing

First, various receipts are directly read by the scanner. To ensure an efficient OCR process, the receipt must be converted into an image for preprocessing. Distorted, occluded, or noisy images affect the accuracy of OCR [15]. Before text recognition, the light and dark areas of the receipt content must be analyzed to ensure the correct recognition of each character. The preprocessing operation comprises standardization, binarization, and morphological processing tasks. The trademark and stamp on the receipt are removed to facilitate the subsequent character segmentation and recognition tasks.

2.1.1. Standardization

Even if receipts in the same format are scanned, the width and height of the images would differ slightly. To facilitate the subsequent segmentation and OCR tasks, the resolution of the receipt image must be adjusted to a fixed size.

2.1.2. Binarization

Image binarization involves converting a color image into a black-and-white image. Accordingly, the color receipt image is converted into a binary image by using adaptive mean thresholding to facilitate the correct reading of the receipt information. Adaptive mean thresholding is a local thresholding method wherein the threshold at the pixel location is determined according to the pixel value distribution in the pixel neighborhood. Therefore, local image areas with different brightnesses and contrast levels have different

local thresholds. If the image width is W, the image length is H, and the pixel grayscale value is f(x,y), then the adaptive mean threshold $T(x,y)$ of a pixel can be expressed as follows:

$$T(x,y) = Mean(\sum_{i=x-r}^{x+r}\sum_{j=y-r}^{y+r} f(i,j)) - C, (0 \leq \mathrm{x} < \mathrm{W})(0 \leq \mathrm{y} < \mathrm{H}) \quad (1)$$

where r is the size of the neighborhood used for calculating the local threshold and C is a constant term. The function $Mean(x,y)$ represents calculating the mean value of the gray image. The final threshold is the difference between the average threshold calculated in the neighborhood and the constant term.

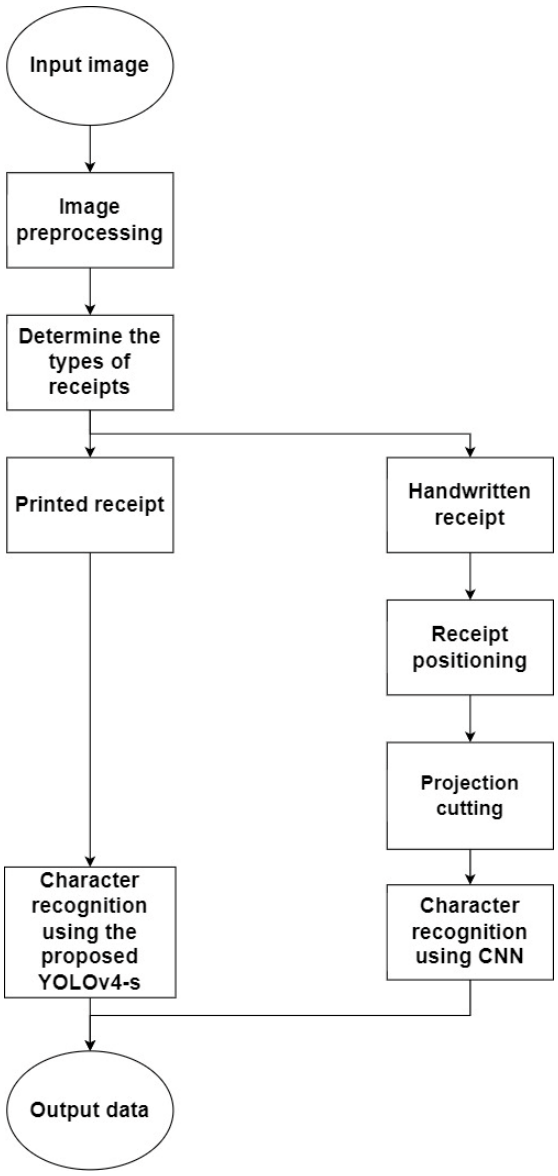


Figure 1. Flowchart of the proposed ARRS.

2.1.3. Elimination of Trademarks

Some receipts have colored trademarks or prints on the top or around the sides. To prevent a trademark from influencing the image projection and segmentation processes, our system filters the interference information [16] by converting an RGB image into the HSV color space. The range of the trademark or print is identified and eliminated such that the color of this range is the same as the background color of the receipt.

2.2. Text Positioning

The image positioning process involves the following tasks: rough positioning and precise positioning. These tasks are detailed as follows.

2.2.1. Rough Positioning

Although a receipt contains a substantial amount of information, only eight pieces of information must be recorded. As mentioned in the preceding operational steps, the receipt image is standardized. In most receipts, the location of the information to be extracted is fixed. Therefore, the rough positioning process is used to locate different types of information on the basis of experience. This reduces the range of positioning and the calculation time.

2.2.2. Precise Positioning

In the layout of receipts, specific signs (or fixed words) denote the position of necessary information. On the basis of these signs, the proposed system can execute template matching to precisely identify the information location. The precise positioning process comprises the following steps:

Step 1: Template images are collected in advance to facilitate image matching.

Step 2: Template matching is executed using the root mean square deviation (Equation (2)). Specifically, the proposed system uses a template image to sequentially traverse the entire image for a similarity comparison.

$$R(x,y) = \frac{S(x,y)}{\sqrt{\sum_{x'y'} T(x',y')^2 . \sum_{x'y'} I(x+x',y+y')^2}} \quad (2)$$

Subsequently, a matching matrix is returned after the comparison process. The position of the template image in the image to be detected is determined as shown in Figure 2. Here, the normalized sum of the squared difference in Equation (2) is standardized according to the sum of the squared difference in Equation (3).

$$S(x,y) = \sum_{x'y'} \left(T(x',y') - I(x+x',y+y')\right)^2 \quad (3)$$

where the $R(x,y)$ is normalized squared difference at pixle (x,y), $S(x,y)$ is squared difference at pixle (x,y), T is subgraph, I represents original image.

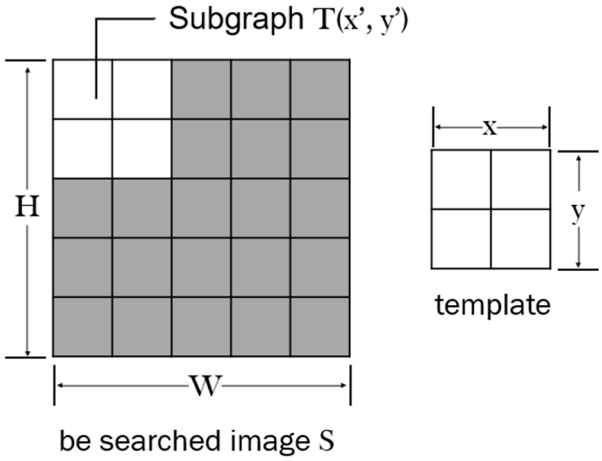


Figure 2. Diagram of template matching.

2.3. Character Segmentation

For character segmentation, the proposed ARRS uses a projection method to project the input image horizontally and vertically. The system then counts the number of pixels with a gray value of 0 in each row (column) of the input image. The areas with the peak number of such pixels correspond to the image edges; this quantification is conducted in different rectangular boxes [17]. The horizontal projection (along the y-axis) equation and vertical projection (along the x-axis) equation are expressed as follows:

$$I_y = \sum_{y=0}^{H-1} f(x,y), \quad (0 \le x \le W) \tag{4}$$

$$I_x = \sum_{x=0}^{W-1} f(x,y), \quad (0 \le y \le H) \tag{5}$$

where I_y is the number of pixels in the horizontal direction, I_y is the number of pixels in the vertical direction, $f(x,y)$ is the original image, W is the image width, and H is the image height.

Projection is extensively used in character segmentation [18]. The basic steps of the projection process are outlined as follows:

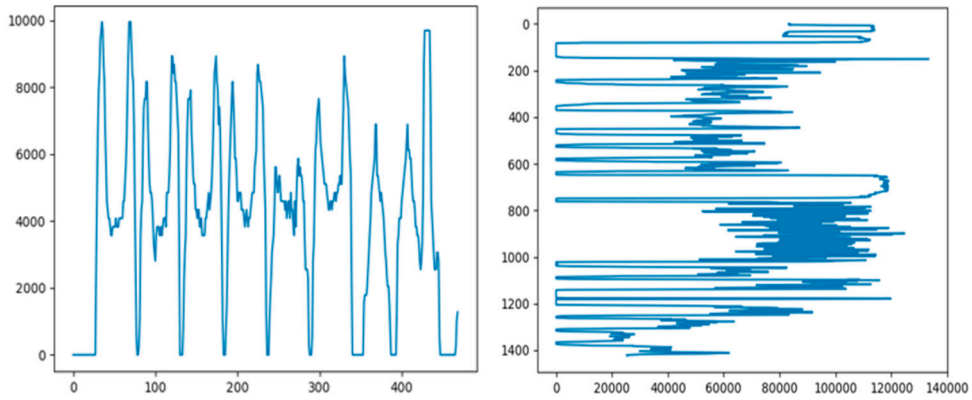
Step 1: The horizontal projection process is performed on the input image. That is, the number of pixels in each column in the horizontal direction I_y is calculated.

Step 2: The vertical projection process is performed on the image. Initially, I_y is equal to 0 because of the lack of positioning information. The pixel for which I_y is greater than 0 is considered to indicate the starting point of the relevant line of a character on the row. The image is scanned until another pixel for which I_y is equal to 0 is identified; this pixel indicates the end of this line of character. Because the scanning is conducted in a straight line, all segments of each line of character can be found.

Step 3: A vertical segmentation process is performed according to the segment points of each line of character obtained in Step 2.

Step 4: To avoid overlapping characters in the vertical direction, each vertically segmented character is projected vertically and scanned horizontally. This method is similar to horizontal projection; thus, each line of characters in the receipt can be segmented separately.

Because the text on a standard receipt is neatly arranged and the size of the printed characters is uniform, projection-based segmentation can be easily performed on such receipts. However, for other types of receipts with connected characters, improved projection-based segmentation methods must be used to achieve higher accuracy. Accordingly, the proposed ARRS performs morphological operations before segmentation to remove noise from the image and separate characters that may be connected. Because the size of the characters printed on a receipt is uniform, the average length and width of the characters are calculated as thresholds to determine whether the characters are accurately segmented. Therefore, the receipt image is projected horizontally and vertically, as displayed in Figure 3. Figure 3a presents the vertical projection results, indicating eight wave crests, each of which represents a character. Thus, the position of the character can be easily judged, and the character can be cut smoothly. Figure 3b presents the horizontal projection results for the entire receipt image; the distance between the troughs can also be used to obtain the paragraph of each line of text.



(**a**) (**b**)

Figure 3. (**a**) Vertical projection of the receipt image (**b**) Horizontal projection of the receipt image.

2.4. Character Recognition

As mentioned, receipts in Taiwan can be divided into two types according to their characters: handwritten and printed receipts. The proposed ARRS recognizes characters on these receipts separately. For the recognition of handwritten characters, the input image is preprocessed, template matching is executed for text positioning, and character segmentation is executed. Subsequently, the CNN [19] is used to recognize the handwritten characters. For the recognition of printed characters, the input image is projected and segmented, after which the YOLOv4-s model—a modified version of the YOLOv4 model [20] implemented in this study to improve the recognition accuracy—is used for character recognition. The YOLOv4-s model can optimize the parameters and architecture of the original YOLOv4 model.

2.4.1. Recognition of Handwritten Characters through CNN

The proposed ARRS uses AlexNet, a CNN model, to recognize handwritten receipt characters. AlexNet is a classic model in image recognition. It consists of eight layers of neurons; the first five layers are convolutional layers and pooling layers, and the remaining three layers are fully connected layers. The convolutional layers are used to extract image features; the pooling layers are used to reduce the dimensionality of the features. The fully connected layers are used for image classification. Figure 4 shows the structure of AlexNet, and Table 1 presents its parameter settings for the proposed system.

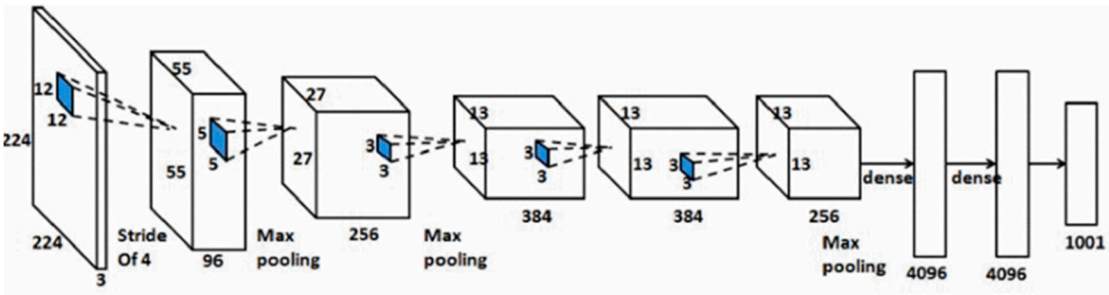


Figure 4. Structure of AlexNet.

Table 1. Parameter settings of AlexNet.

Layer	Filter	Kernel Size
Convolution Layer1	96	11 × 11
Max Pooling Layer1		3 × 3
Convolution Layer2	256	5 × 5
Max Pooling Layer2		3 × 3
Convolution Layer3	384	3 × 3
Convolution Layer4	384	3 × 3
Convolution Layer5	256	3 × 3
Max Poling Layer3		
Flatten Layer1		
Full Connected Layer1	4096	
Full Connected Layer2	4096	
Full Connected Layer3	10	

2.4.2. Recognition of Printed Characters through YOLOv4-s

The YOLOv4 model is used for object detection on images in computer vision. The object type must be identified, and the locations of these objects must be marked. The YOLOv4 model performs object recognition through OCR rather than using traditional character recognition; this thus improves the robustness of the recognition system. The YOLOv4 model is an instant and high-precision target detection system with 161 network layers. It applies a 1 × 1 convolution operation, which reduces the number of calculations required and increases operational speed. Figure 5 displays the network architecture of the YOLOv4 model. This model uses a cross-stage partial network (CSPNet) to reduce the duplication of gradient information and applies spatial pyramid pooling to separate the features of upper and lower layers. In addition, it applies a path aggregation network and feature pyramid network (FPN) to facilitate the transmission of low-level features to the top networks.

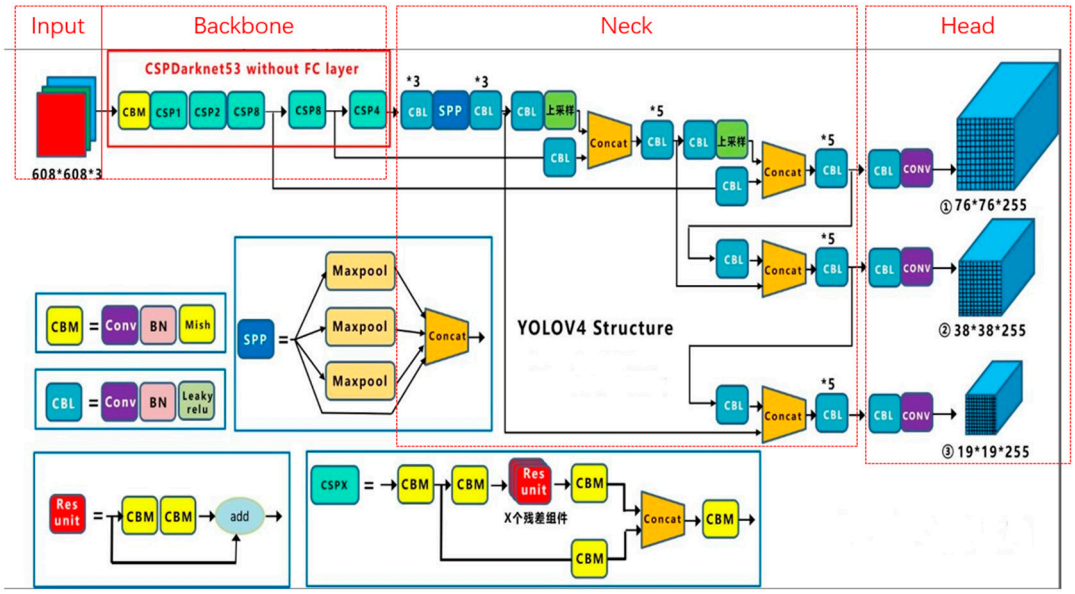


Figure 5. Architecture diagram of traditional YOLOv4.

Nevertheless, to reduce downsampling and enhance the recognition of small objects, the YOLOv4 model requires modification. Accordingly, the present study developed the YOLOv4-s model, a modified version that was realized by using k-means clustering to optimize the anchor parameters of YOLOv4 and by optimizing the backbone architecture of YOLOv4 to ensure small-object recognition.

Anchor Parameter Optimization Using K-Means Clustering

In the target detection model, the target is "framed" at a possible position by the preset border and then adjusted according to these preset borders. The YOLOv4 model uses an anchor box to generate borders at different positions of the image. The regional features corresponding to these borders are extracted and used for the regression of border positions. Figure 6 illustrates a real object determined using the anchor box; the blue box represents the real object, and the black box represents the anchor box. The real object is detected using the anchor box through the following formula:

$$\begin{aligned} b_x &= \sigma(t_x) + c_x \\ b_y &= \sigma(t_y) + c_y \\ b_w &= p_w e^{t_w} \\ b_h &= p_h e^{t_h} \end{aligned} \tag{6}$$

where b_x, b_y are the x, y center coordinates of box, b_w, b_h width and height of the box, t_x, t_y, t_w, and t_h represent the network outputs, c_x and c_y are the top-left coordinates of the grid, p_w and p_h represent the width and height of the anchor box, respectively, and c_x/c_y represents the distance between a cell and the top-left corner of the object. To ensure that the anchor box is close to the ground-truth box and reduce the deviation of the prediction network, the model uses k-means clustering to identify the most suitable p_w and p_h values.

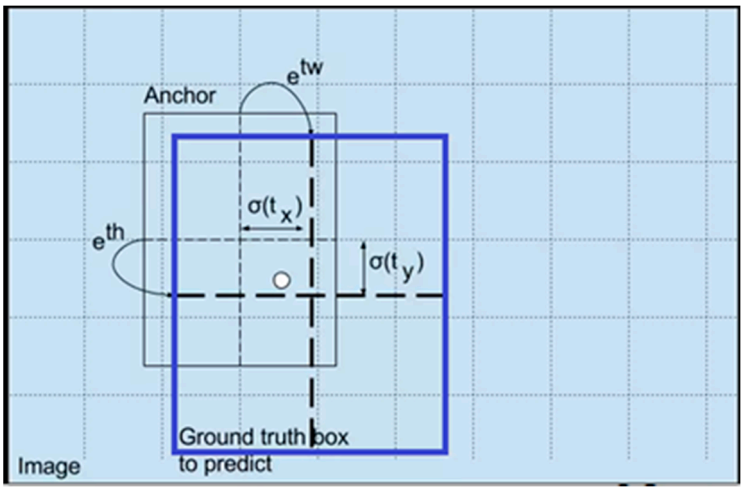


Figure 6. Identifying a real object by using an anchor box.

In general, k-means clustering is used in data mining. In this method, n points are divided into k clusters such that each point belongs to the nearest cluster center.

The steps for calculating the appropriate anchor box through k-means clustering are outlined as follows:

Step 1. The width and height of the box are normalized.

When a cluster analysis is executed on multiple boxes, the width and height of the box are used as features. Because the sizes of the receipt images in our dataset may differ, the width and height of each receipt image must be normalized to the width and height of the box. The normalization formula is expressed as follows:

$$w = \frac{w_{box}}{w_{img}}, \quad h = \frac{h_{box}}{h_{img}} \tag{7}$$

where w and h are normalized width and height, w_{box} and h_{box} are the box width and height, w_{img} and h_{img} are the image width and height.

Step 2. K boxes are randomly selected as the initial anchor.

The YOLOv4 model contains three layers of different sizes, and each layer selects three anchor boxes. Therefore, the size of K can be set to 3×3, implying a total of nine anchor boxes.

Step 3. Intersection over union (IoU) is used as a metric, and each box is assigned to the anchor closest to it. IoU is more suitable as a metric than the Euclidean distance in the original k-means clustering method. If anchor = (w_a, h_a) and box = (w_b, h_b), then the IoU between the anchor and the box is as follows:

$$\begin{aligned} IoU(box, anchor) &= \frac{intersection(box, anchor)}{union(box, anchor) - intersection(box, anchor)} \\ &= \frac{\min(w_a, w_b) \cdot \min(h_a, h_b)}{w_a h_a + w_b h_b - \min(w_a, w_b) \cdot \min(h_a, h_b)} \end{aligned} \tag{8}$$

where the w_a and h_a are the width and height of anchor, the h_a and h_b are the width and height of box.

The final distance d formula is expressed as follows:

$$d(box, anchor) = 1 - IoU(box, anchor) \tag{9}$$

If the box and the anchor completely overlap, the IoU value is 1, and the distance between them is 0.

Step 4. The average width and height of all the boxes in each cluster are calculated to update the anchor.

Assume that n boxes exist in the cluster:

$$cluster(box_0, box_1 \ldots box_n),\ box_n = (w_n, h_n) \tag{10}$$

where the w_n and h_n are the width and height of box_n.

The average width and height of all boxes can be derived as follows:

$$mean(cluster) = \frac{\sum_{i=0}^{n}(w_i,\ h_i)}{n} \tag{11}$$

where the w_i and h_i are the width and height of box_i, n represent the total number of box.

Step 5. Steps 2 and 3 are repeated until the anchor does not change or the maximum number of iterations is reached.

Backbone Architecture Optimization for Small-Object Recognition

The backbone of the YOLOv4 model is characterized by a 32× downsampling, and features processed at 8×, 16×, and 32× downsampling features are sent back to the FPN. A higher downsampling rate results in a smaller feature map but a larger object. In other words, the features transmitted from the backbone architecture tend to be larger objects. The feature s of small objects gradually disappears after the convolution and downsampling processes.

To improve the accuracy of the YOLOv4 model in extracting the features of small objects, its backbone architecture must be modified. Accordingly, this study deleted the convolutional and bottleneck layers of CSPNet in the model. In addition, the downsampling features were modified to 4×, 8×, and 16×, thus yielding the optimized YOLOv4-s model that can retain more small-object features and send them to the FPN. The YOLOv4-s model is detailed as follows:

(1) The YOLOv4-s model does not contain the last layer of CSP4 in the backbone architecture of YOLOv4 (Figure 7a), as displayed in Figure 7b.
(2) The shortcut connection between the backbone and the FPN is moved between CSP2 and the first CSP8, as illustrated in Figure 7b.

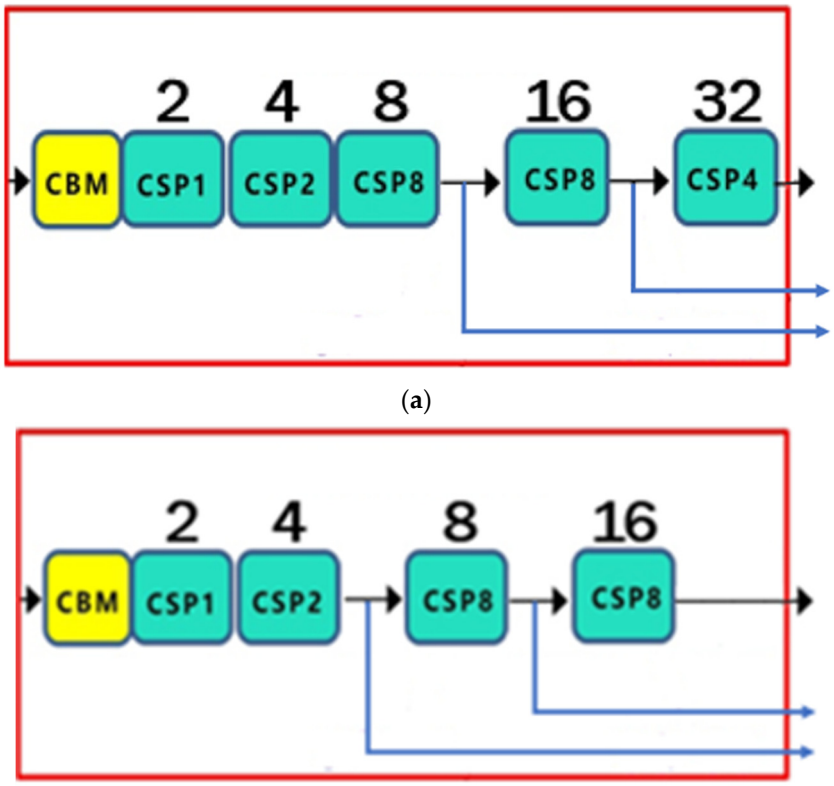


Figure 7. Architecture of (**a**) traditional YOLOv4 and (**b**) the proposed YOLOv4-s.

Figure 8 presents the complete architecture of the proposed YOLOv4-s model.

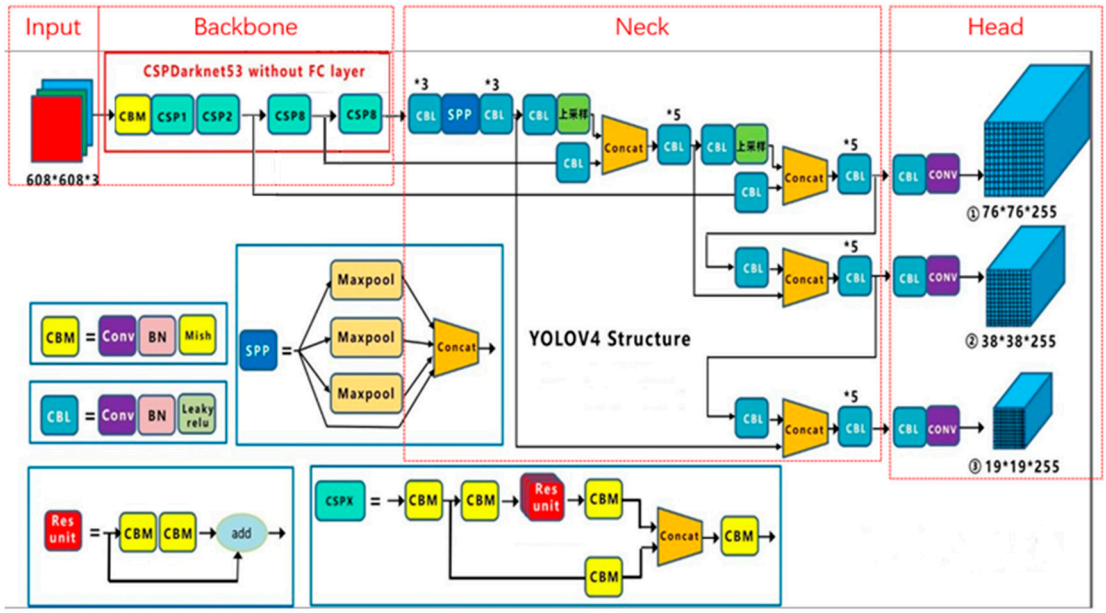


Figure 8. Complete architecture of the proposed YOLOv4-s.

3. Experimental Results

This section describes the overall experimental structure, data collection process, recognition results for handwritten and printed receipt characters, and human–machine interface of the proposed ARRS.

3.1. Overall Experimental Structure

Figure 9 illustrates the overall experimental structure. First, we used a scanner to convert a receipt image into a high-resolution image. Next, the receipt image was preprocessed through adaptive binarization, trademark removal, and receipt character classification (i.e., printed or handwritten characters). The handwritten and printed receipt characters were recognized using the methods described in the preceding section. The recognition results were uploaded to a database, and a web interface is provided for users to review and correct errors. Finally, the results were exported in the form of tax filing documents to complete the tax filing process.

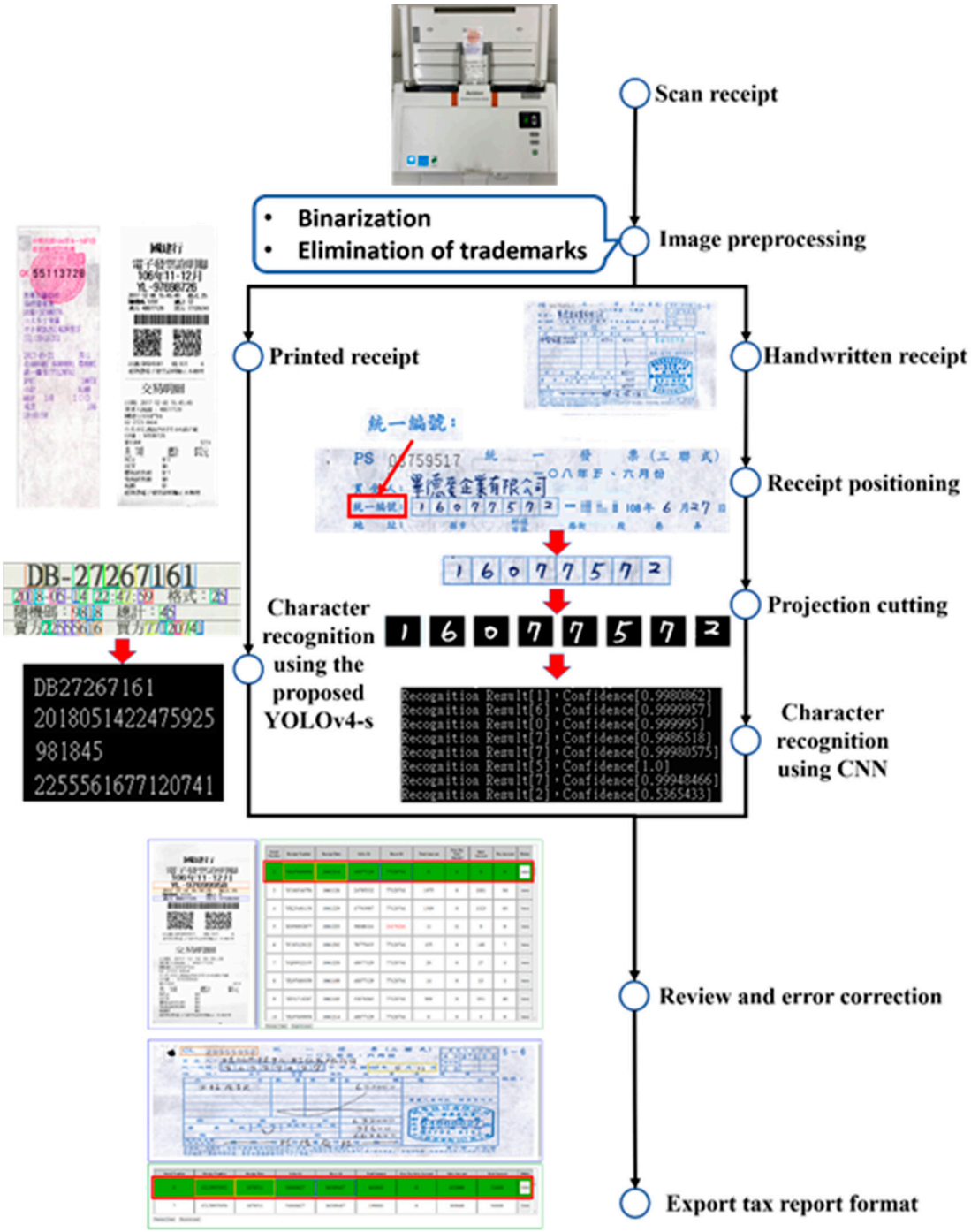


Figure 9. Overall experimental structure.

The receipt information required for tax filing in Taiwan mainly consists of eight items (Figure 10): receipt number, receipt date, seller ID, buyer ID, total amount, tax-free sales amount, sales amount, and tax amount. Therefore, the proposed ARRS can primarily recognize these eight items.

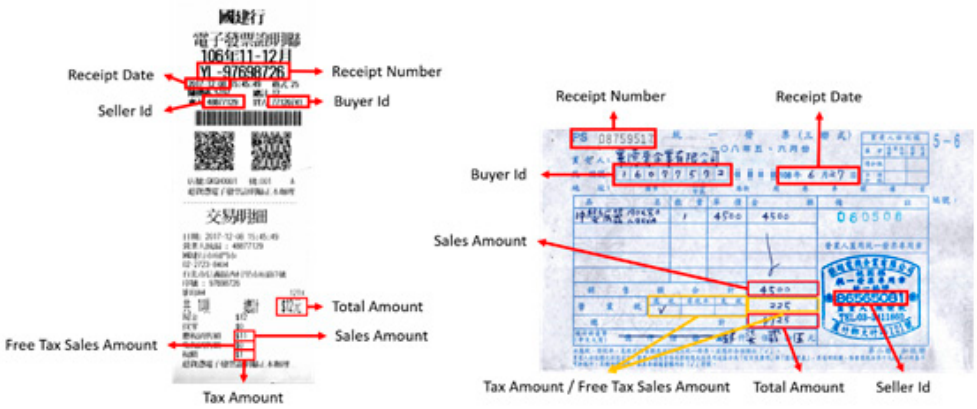


Figure 10. Receipt information required for tax filing.

3.2. Data Collection

The six common receipt formats in Taiwan are electronic receipt certification sheets, cash register unified receipts, triple unified receipts, double unified receipts, special unified receipts, and electronic computer unified receipts (Figure 11).

In this experiment, the receipt characters were classified as handwritten or printed characters during the recognition process. Accordingly, approximately 300 printed and 100 handwritten receipt characters were obtained. Therefore, the dataset was divided into the following categories: handwritten and printed datasets.

The numerical characters in the printed dataset were divided into 10 categories: 0 to 9. Each category comprised 500 numerical characters, of which 400 were used as training data and the remaining 100 were used as test data. Figure 12a presents the numerical characters. Moreover, the alphabetical characters were divided into 26 categories: A to Z. Each category comprised 200 characters, of which 160 were used as training data and the remaining 40 were used as test data. Figure 12b depicts the alphabetical characters. Figure 13a,b display the accuracy and loss function for the numerical and alphabetical characters in the printed dataset, respectively.

In the handwritten dataset, the numerical characters were also divided into 10 categories: 0 to 9. Each category comprised 900 characters, of which 720 were used as training data and the remaining 180 were used as test data. Figure 14 depicts these numerical characters. Figure 15a,b illustrate the accuracy and loss function of the numerical characters in the handwritten dataset, respectively.

3.3. Experimental Results for Handwritten Receipt Characters

A precaptured template image was used for template matching for the handwritten characters. After each receipt image is completely scanned, the matching image position was determined, and the text area was extracted using the relative position of the matching image. Figure 16 displays the template matching results.

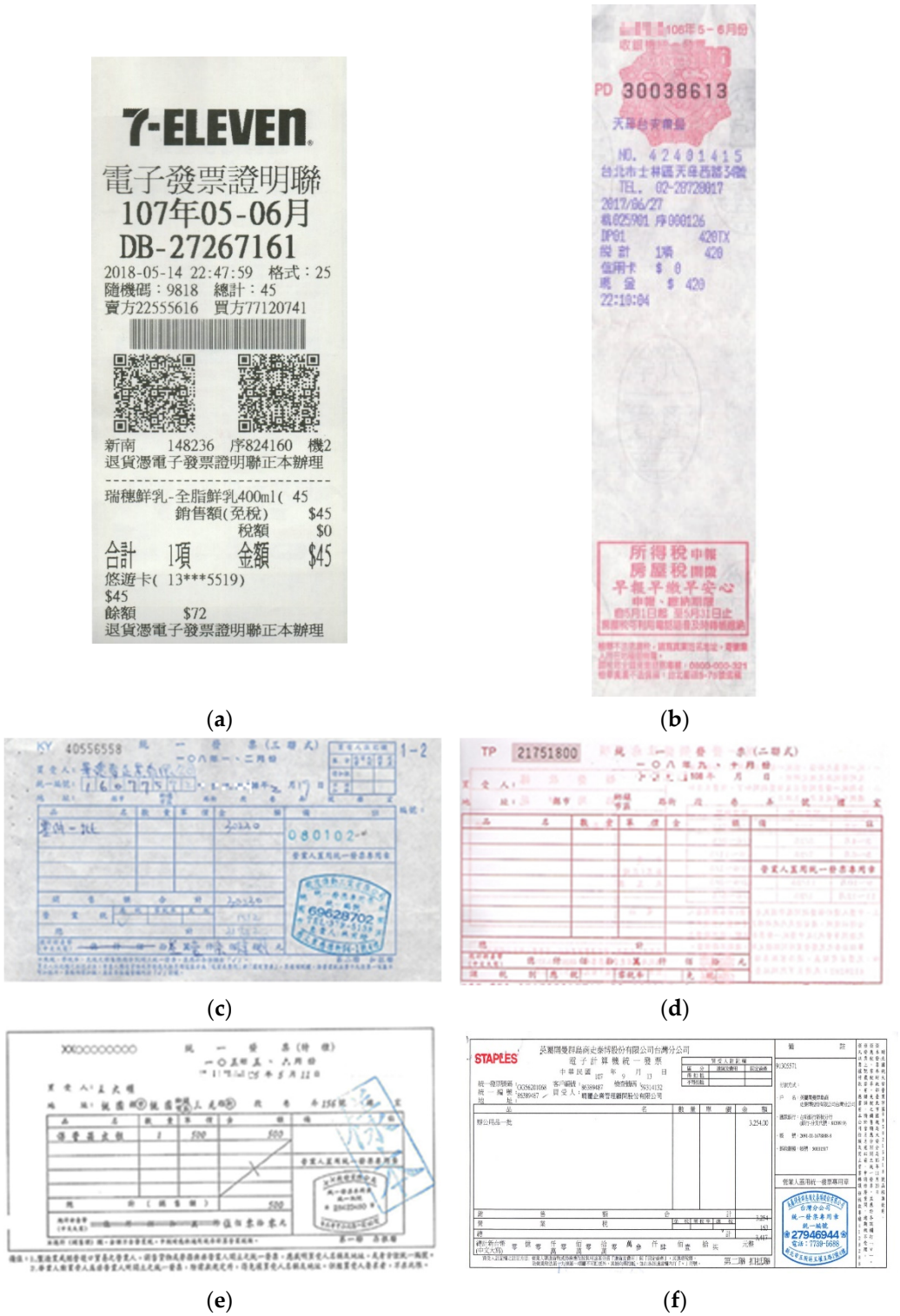
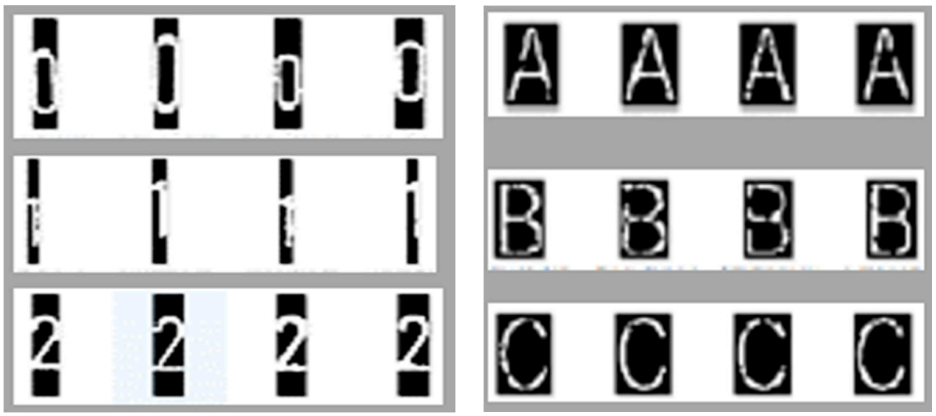
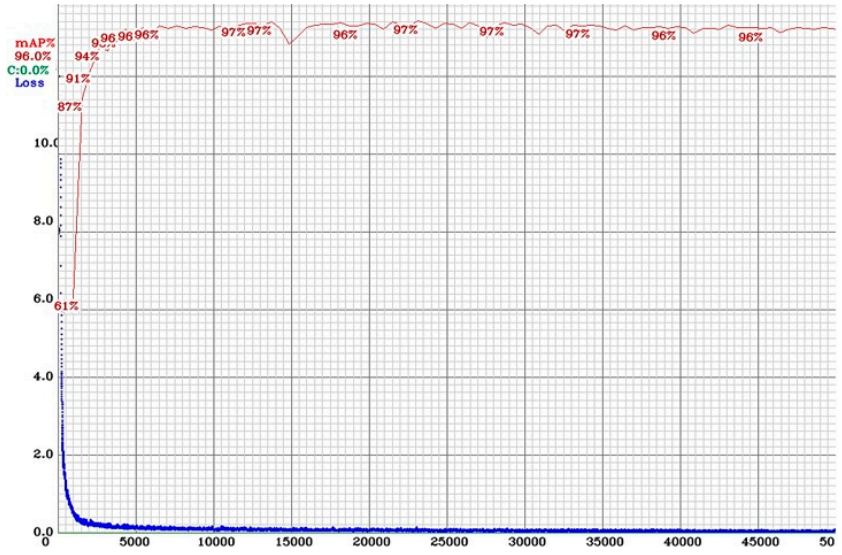


Figure 11. Various receipt formats: (**a**) electronic receipt certification sheet, (**b**) cash register unified receipt, (**c**) triple unified receipt, (**d**) double unified receipt, (**e**) special unified receipt, and (**f**) electronic computer unified receipt.

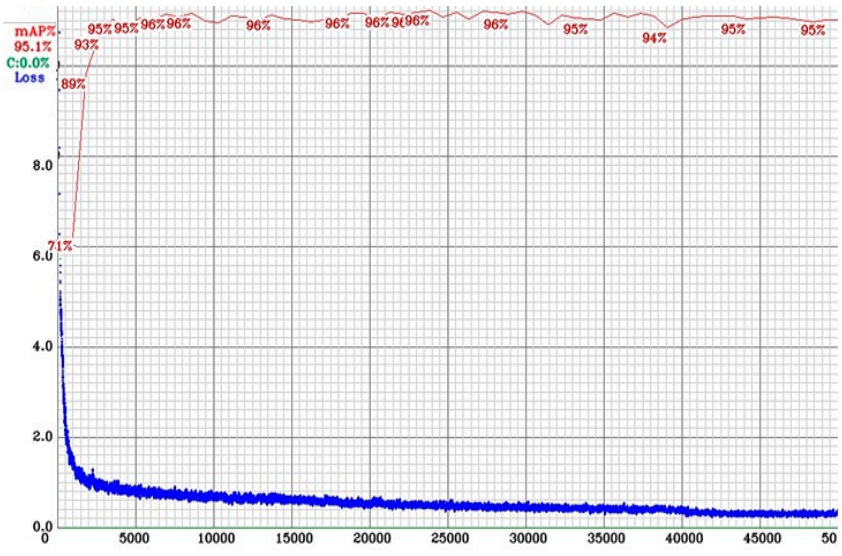


(**a**) (**b**)

Figure 12. (**a**) Numerical and (**b**) alphabetical characters in the printed dataset.



(**a**)



(**b**)

Figure 13. Accuracy and loss function of (**a**) numerical and (**b**) alphabetical characters in the printed dataset.

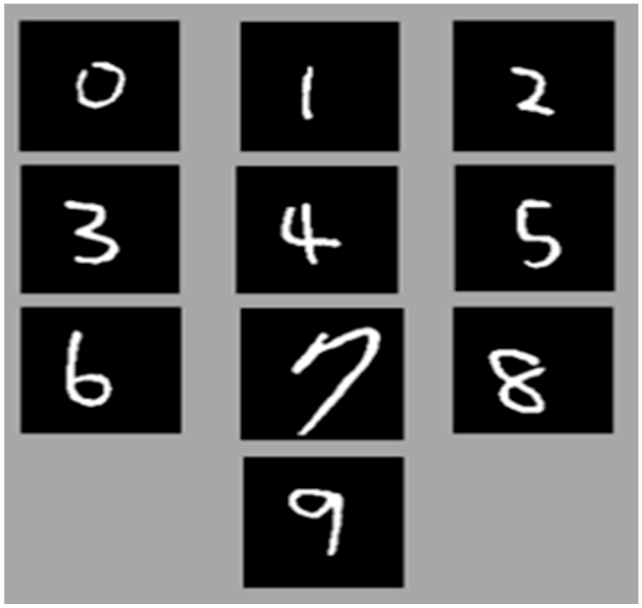
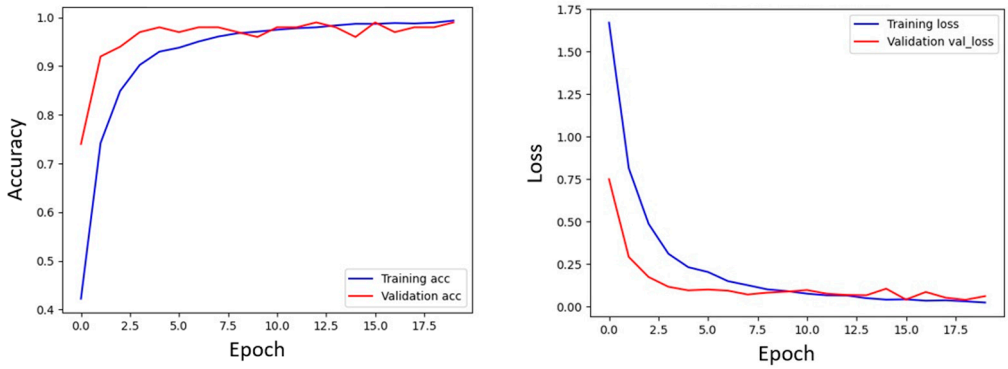


Figure 14. Numbers in the handwritten dataset.



(**a**) (**b**)

Figure 15. (**a**) Accuracy and (**b**) loss function of the numbers in the handwritten dataset.

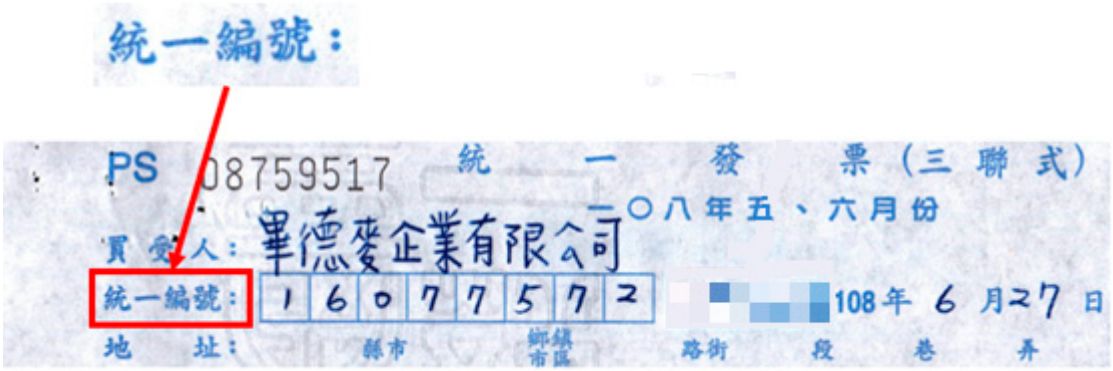


Figure 16. Template matching results.

The erosion method was used to extract the vertical lines in the table after the text area can be obtained. Subsequently, a projection method that determines the position of the vertical line was used to extract each box. Next, the character segmentation initiation point was identified, and the original receipt image was segmented to obtain the handwritten character images. Then, the handwritten character images were processed by using the

binarization method to binary images (Figure 17). Finally, CNN was adopted to identify relevant characters. Figure 17 shows the projection and segmentation results.

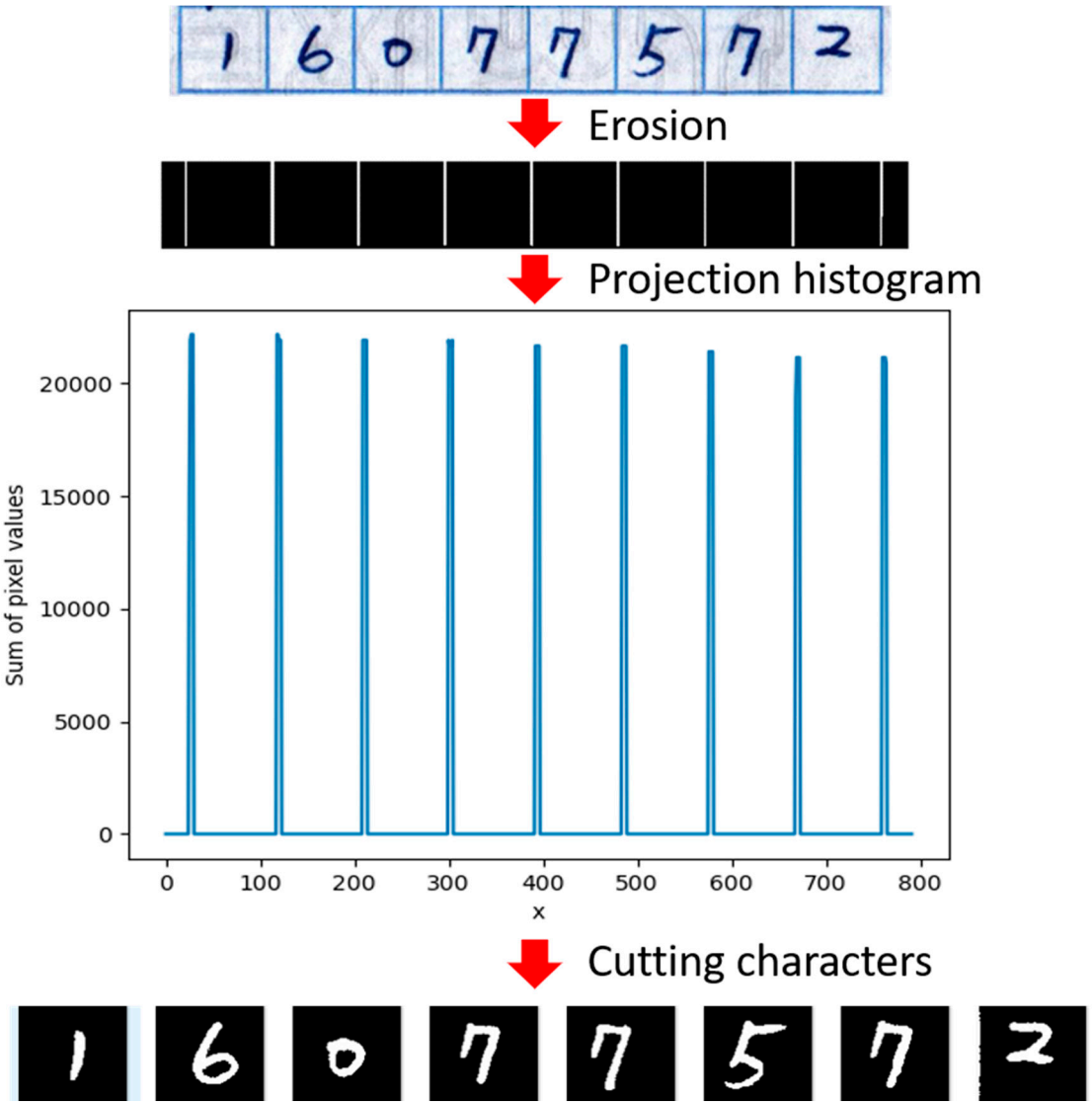


Figure 17. Projection and segmentation results.

The derived character image was used as the input of the CNN model for recognition. The recognition results were stored in the database for subsequent review and error correction. Figure 18 displays the CNN recognition results.

A total of 145 handwritten receipt characters were obtained in the verification set. Moreover, a total of 6815 characters were noted, of which 5516 were successfully recognized, resulting in a recognition accuracy of 80.93%.

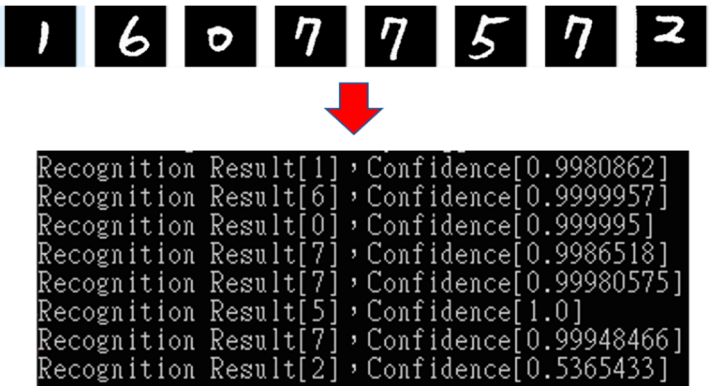


Figure 18. CNN recognition results.

3.4. Experimental Results for Printed Receipt Characters

Each of the printed receipt images was preprocessed, after which horizontal projection, segmentation, and rough positioning processes were executed. The proposed YOLOv4-s model was then applied to recognize the printed characters. In Figure 19, the left panel depicts an original receipt image; the dotted red frame in the middle indicates the information to be extracted. Moreover, in Figure 19, the upper-right panel presents the positioning results for the recognized characters, the middle panel presents the YOLOv4-s model recognition results, and the right panel presents the key information extracted.

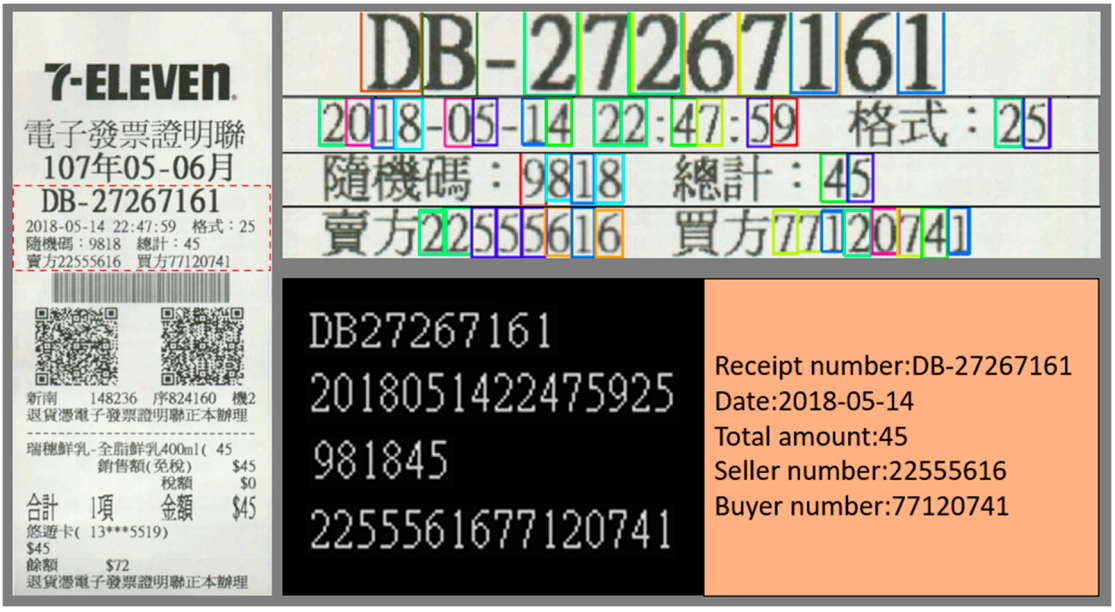


Figure 19. Recognition results of the printed receipts.

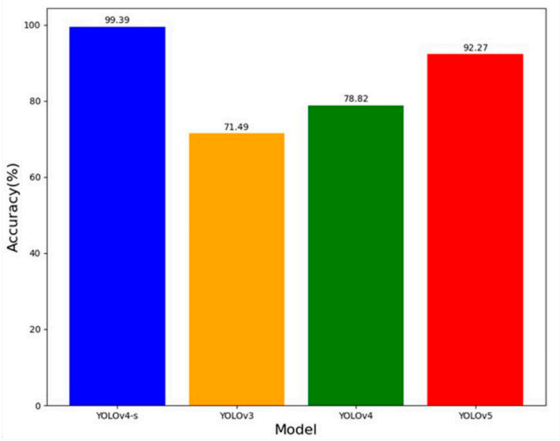
To verify the effectiveness of the proposed YOLOv4-s model in recognizing printed receipt characters, 140 receipts were used for testing. A total of 5452 numbers and English letters were used. At the same time, YOLOv3 and YOLOv5 models are also used to compare recognition performance. As illustrated in Figure 20, the recognition accuracy of the proposed YOLOv4-s model was higher than that of the traditional YOLOv4 model by 20.57%. Compared with YOLOv3 and YOLOv5, the proposed YOLOv4-s method is also higher than 27.9% and 7.12% in recognition performance, respectively. The recognition accuracy of the proposed YOLOv4-s is 99.39%, and only 33 characters were misjudged. Although the proposed model has the lowest FPS score, it can still obtain close to 60FPS.

That is to say, the proposed model has real-time recognition performance and is competitive. The definition of accuracy and error were shown as follows:

$$Accruacy = \left(1 - \frac{E_c}{T_c}\right) \times 100\% \tag{12}$$

$$Error = \frac{E_c}{T_c} \times 100\% \tag{13}$$

where the accuracy represents the accuracy of model identification, the T_c and E_c are the total characters and identify wrong characters in testing dataset, the error represents the error rate of model identification.



	YOLOv4-s		YOLOv3		YOLOv4		YOLOv5	
5452	Error	0.61%	Error	28.51%	Error	21.18%	Error	7.73%
Total characters	Accuracy	99.39%	Accuracy	71.49%	Accuracy	78.82%	Accuracy	92.27%
FPS	58FPS		90FPS		70FPS		138FPS	

Figure 20. Comparison of recognition accuracy of various methods.

3.5. Human–Machine Interface of the Proposed ARRS

Because the items on a receipt are interrelated, they can be used to determine the accuracy of recognition results. Accordingly, this study proposed four logical judgment rules for accuracy verification, which are explained as follows:

(1) Allowance + Taxable amount + Tax amount = Total amount. If the equation does not hold, it means that at least one of the four recognized numbers is wrong. Therefore, the system should issue an error message on the interface and mark the error. The user is reminded to manually modify the error content.

(2) The uniform number of the seller and the buyer is a combination of eight numbers. If this rule is not met, the system should issue an error message on the interface and mark the error.

(3) The character track consists of a combination of two uppercase alphabetical characters and eight numerical characters. If this rule is not met, the system should issue an error message on the interface and mark the error.

(4) If the display of the date from the Western year to the Chinese year is not within a reasonable range, the system issues an error. Some receipts, such as electronic certificate slips, have two date marks. If the two recognition results are not the same, the system should issue an error message and mark the error.

After the recognition of the receipt content, the information is stored in the database. After the content of the entire batch of receipts is recognized, the system automatically opens the detection interface for manual detection and confirmation. HyperText Markup Language was used to construct the interface of the system. The system interface can be adjusted to match various sizes and formats of receipts. This can enable users to complete

detection and confirmation in the shortest time. Figures 21 and 22 depict the system interfaces for handwritten and printed receipts, respectively.

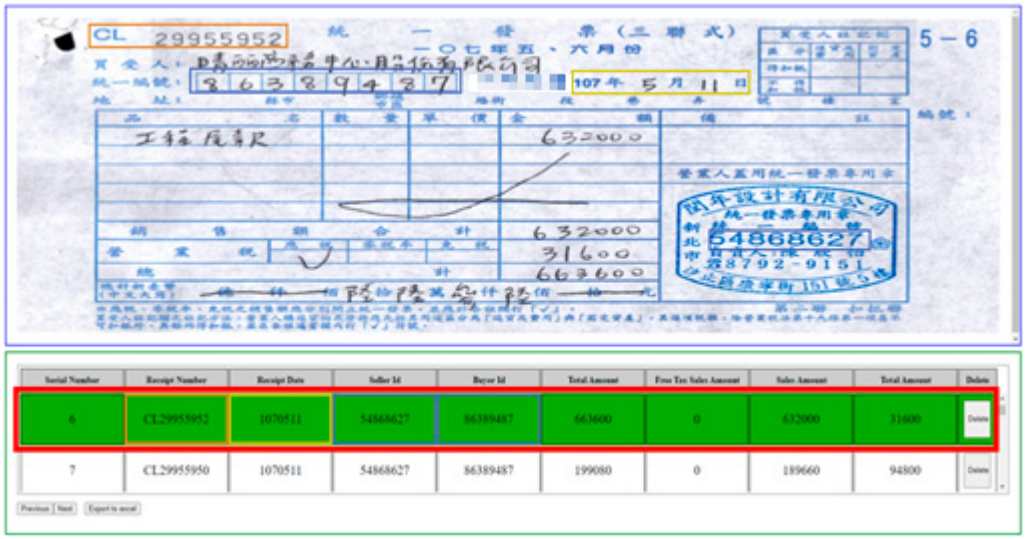


Figure 21. System interface for handwritten receipts.

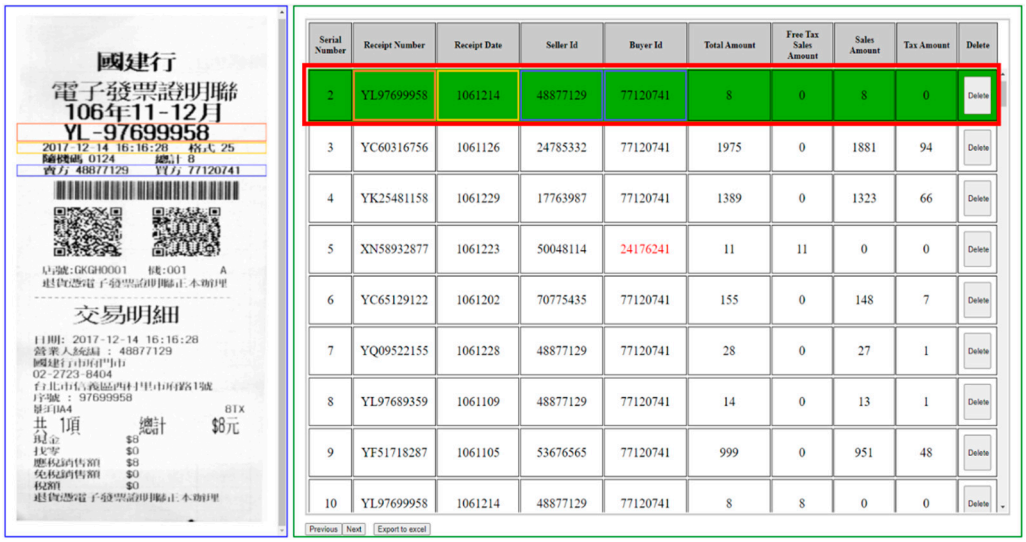


Figure 22. System interface for printed receipts.

The system interface provides the following functions:

(1) The current receipt information is marked with different colors to facilitate user detection and confirmation.
(2) If the receipt is not within the scope of this declaration, it is deleted.
(3) The up/down button can be used to switch between receipts.
(4) After the detection and confirmation of a batch of receipts, the information can be exported in the format of a tax declaration.

In Figure 22, the left panel presents the receipt image detected in this study, and the red box in the right panel presents the corresponding recognition results. The system interface can also mark errors in red, as displayed in the fifth receipt shown in the right panel of Figure 22.

4. Conclusions

This study developed an ARRS for improving operational efficiency and reducing human errors in the processing of receipt information for tax declaration tasks. In this system, a receipt is scanned into a high-resolution image, and the characters on the receipt are automatically classified into two categories according to the characteristics of the receipt characters: printed and handwritten receipts. Images of receipts with two types of characters are preprocessed separately. For handwritten characters, template matching and the fixed features of the receipt are used for text positioning, and a projection is for character segmentation. Finally, a CNN is used to achieve character recognition. For printed characters, the proposed YOLOv4-s model executes precise text positioning for character recognition. The proposed YOLOv4-s reduces downsampling features, thereby enhancing small-object recognition. The proposed system also provides a human–machine interface to enable users to review the results and perform error correction. The effectiveness of the proposed ARRS was tested experimentally. The experimental results reveal that the system had a recognition accuracy rate of 80.93% for handwritten receipt characters. Moreover, the recognition accuracy of the proposed YOLOv4-s model for printed receipt characters was 99.39%, and only 33 characters were misjudged. The recognition accuracy of the YOLOv4-s model was higher than that of the traditional YOLOv4 model by 20.57%.

The proposed ARRS's recognition accuracy for handwritten receipts was approximately 80%, which is not up to the industry standard. On the other hand, ARRS runs on a personal computer device, which causes a lot of inconveniences for users to carry-outs. In future research, we will focus on collecting more handwritten receipts and improving the recognition accuracy of the system. In addition, we will consider using ISO GUM as an indicator to evaluate the experimental results. ARRS will also be implemented in FPGA embedded systems to obtain faster recognition speed and improve its portability.

Author Contributions: Conceptualization, C.-J.L. and Y.-C.L.; Methodology, C.-J.L., Y.-C.L. and C.-L.L.; Software, C.-J.L. and Y.-C.L.; Data Curation, C.-J.L. and Y.-C.L.; Writing—Original Draft Preparation, C.-J.L.; Writing—Review and Editing, C.-J.L. and C.-L.L.; Supervision, C.-J.L.; Funding Acquisition, C.-J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Ministry of Science and Technology of the Republic of China, grant number MOST 110-2634-F-009 -024 and MOST 109-2218-E-005-002.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rahmat, R.F.; Gunawan, D.; Faza, S.; Haloho, N.; Nababan, E.B. Android-based text recognition on receipt bill for tax sampling system. In Proceedings of the 2018 Third International Conference on Informatics and Computing (ICIC), Palembang, Indonesia, 17–18 October 2018; pp. 1–5.
2. Laroca, R.; Severo, E.; Zanlorensi, L.A.; Oliveira, L.S.; Gonçalves, G.R.; Schwartz, W.R.; Menotti, D. A robust real-time automatic license plate recognition based on the YOLO detector. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–10.
3. Liu, W.; Yuan, X.; Zhang, Y.; Liu, M.; Xiao, Z.; Wu, J. An end to end method for taxi receipt automatic recognition based on neural network. In Proceedings of the 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 12–14 June 2020; pp. 314–318.
4. Shi, S.; Cui, C.; Xiao, Y. An invoice recognition system using deep learnin. In Proceedings of the 2020 International Conference on Intelligent Computing, Automation and Systems (ICICAS), Chongqing, China, 11–13 December 2020; pp. 416–423.
5. Meng, Y.; Wang, R.; Wang, J.; Yang, J.; Gui, G. IRIS: Smart phone aided intelligent reimbursement system using deep learning. *IEEE Access* **2019**, *7*, 165635–165645. [CrossRef]
6. Wang, X.; Ding, X.; Liu, C. Gabor filters-based feature extraction for character recognition. *Pattern Recognit.* **2005**, *38*, 369–379. [CrossRef]

7. Cui, L.; Ruan, Q. A novel SVM classifier for recognition of gray character using gabor filters. In Proceedings of the 7th International Conference on Signal Processing, (ICSP'04), Beijing, China, 31 August–4 September 2004; Volume 2, pp. 1451–1454.
8. Xie, J. Optical character recognition based on least square support vector machine. In Proceedings of the 2009 Third International Symposium on Intelligent Information Technology Application, Nanchang, China, 21–22 November 2009; pp. 626–629.
9. Hazra, T.K.; Singh, D.P.; Daga, N. Optical character recognition using KNN on custom image dataset. In Proceedings of the 2017 8th Annual Industrial Automation and Electromechanical Engineering Conference (IEMECON), Bangkok, Thailand, 16–18 August 2017; pp. 110–114.
10. Smith, R. An overview of the tesseract OCR engine. In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Curitiba, Brazil, 23–26 September 2007; pp. 629–633.
11. Liu, W.; Yuan, X.; Zhang, Y.; Wu, M.; Du, H.; Cui, Y. An automatic taxi receipt text recognition application system. In Proceedings of the 2020 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Paris, France, 27–29 October 2020; pp. 1–6.
12. Dong, X.; Colleen, P.B. Novel receipt recognition with deep learning algorithms. In Proceedings of the SPIE, Online, 22 May 2020; Volume 11400.
13. Liu, Y.C. Handwritten Invoice Recognition System Using Deep Learning. Master's Thesis, National Taiwan Ocean University, Keelung City, Taiwan, 24 June 2019.
14. Nguyen, C.M.; Ngo, V.V.; Nguyen, D.D. MC-OCR Challenge 2021: Simple approach for receipt information extraction and quality evaluation. In Proceedings of the 2021 RIVF International Conference on Computing and Communication Technologies (RIVF), Hanoi, Vietnam, 19–21 August 2021; pp. 1–4.
15. Wang, L.; Zhang, L.; Yang, X.; Yi, P.; Chen, H. Level set based segmentation using local fitted images and inhomogeneity entropy. *Signal Processing* **2020**, *167*, 107297. [CrossRef]
16. Bradski, G. The OpenCV Library. Dr. Dobb's Journal of Software Tools. 2000. Available online: https://www.bibsonomy.org/user/ross_mck (accessed on 2 December 2021).
17. Ma, R.; Zhang, S. An improved color image defogging algorithm using dark channel model and enhancing saturation. *Optik* **2019**, *180*, 997–1000. [CrossRef]
18. Zhang, J.; Ren, F.; Ni, H.; Zhang, Z.; Wang, K. Research on information recognition of VAT invoice based on computer vision. In Proceedings of the 2019 IEEE 6th International Conference on Cloud Computing and Intelligence Systems (CCIS), Singapore, 19–21 December 2019; pp. 126–130.
19. Mansoor, K.; Olson, C.F. Recognizing text with a CNN. In Proceedings of the 2019 International Conference on Image and Vision Computing New Zealand (IVCNZ), Dunedin, New Zealand, 2–4 December 2019; pp. 1–6.
20. Wu, M.; Zhang, W.; Han, Y.; Mao, Y. Rapid temporal information identification in paper worksheets using light-weight convolutional neural networks. In Proceedings of the 2021 40th Chinese Control Conference (CCC), Shanghai, China, 26–28 July 2021; pp. 8514–8519.