Research Article

Efficient Localization of Multitype Barcodes in High-Resolution Images

Jinwang Yi [ID] **and Yuanbiao Xiao** [ID]

Xiamen University of Technology, Xiamen 361024, China

Correspondence should be addressed to Yuanbiao Xiao; 1922031040@s.xmut.edu.cn

Barcode positioning technology is one of the important components of barcode technology. However, most of the current algorithms are only applicable to a single type of barcode positioning or limited to low-resolution images due to a large amount of calculation, and thereby it is still a challenge to locate multitype barcodes in high-resolution images. In response to the above problems, this paper proposes a reliable multitype barcode localization method for multibarcode localization in high-resolution images where one-dimensional (1D) barcodes, two-dimensional (2D) barcodes, or multitype barcodes are present simultaneously. The method consists of three main steps: first, extracting multiple types of barcode features through a joint edge detection algorithm; next, marking target barcode regions with a bidirectional contour labeling method; and finally, extracting barcode regions by an improved affine transformation. The experimental results show that, in terms of localization accuracy, the proposed method has a better accuracy of 97.83% than existing algorithms in low-resolution images and can locate multitype barcodes in high-resolution images with an accuracy rate of 98.04%. Besides, in terms of time cost, the proposed method effectively reduces the time cost by 50% and improves the barcode localization efficiency.

1. Introduction

Barcode positioning technology is widely used in many industries and fields, such as warehousing, library management, health care, industrial production, and express logistics [1, 2]. However, its efficiency is insufficient for multiple types of barcode localization in high-resolution images because most of the existing methods are only applicable to a single type of barcode localization and the localization process uses complex algorithms, which lead to high time costs.

Produced by Gallo and Manduchi, a simple and fast algorithm is used to calculate the vertical and horizontal gradients of each pixel and perform global binarization for the segmentation of the 1D barcode region [3], which, however, is unsuitable for locating slanted barcodes. Yun and Kim [4] proposed a new 1D barcode localization method that detects barcode features based on the similar structure of entropy and edge orientation in 1D barcodes,

thus realizing the localization of arbitrarily tilted 1D barcodes. Literature [5] exploited the stacked edge features inside 2D barcodes to develop a histogram-based 2D barcode detection method, but it is vulnerable to the interference of dense text. Rincon et al. [6] locate quick response (QR) codes in complex contexts by detecting the structural features of three rectangles in QR codes, effectively addressing the impact of dense text. The above algorithms have high accuracy for single type barcode positioning, but none of them consider the localization of multiple types of barcodes in one image. D. T. Lin and C. L. Lin [7] proposed a multitype barcode localization framework, which utilizes an adaptive thresholding method to extract black and white linear features inside barcodes to achieve multitype barcode localization. However, this method only performs multitype barcode positioning tests in low-resolution images with a resolution of 300×400 pixels. Katona and Nyúl [8] performed edge feature detection of barcodes by improving the Sobel and Feldman algorithm [9], which enables the

framework to perform edge detection of multiple types of barcodes in barcode images with a resolution of 720×480 pixels, compared to the previous method, but its algorithm complexity is high. These algorithms are limited to feature detection of a single type of barcode, or their algorithms have high complexity and face the disadvantages of high time cost and computational inefficiency in high-resolution images.

In the image processing-based barcode localization, barcode region marking is also of great interest. D. T. Lin and C. L. Lin [7] developed a two-channel connected domain analysis algorithm, which can improve the efficiency of barcode region labeling to replace the traditional method. Another important contribution of this algorithm is the ability to label multiple types of barcode regions. Chang et al. [10] studied a contour-detection-based marking method that has a wide range of applications by using the contour-detection technique to track and label the external and internal contours of each target location. By improving this algorithm, Chen et al. [11] proposed a two-stage method for marking the external contours of barcode regions, namely, orientation-based region contour tracking and contour-based region marking. This algorithm can not only mark multiple types of barcode regions but also do it more efficiently because global marking is not required. Liu et al. [12] successfully applied the above algorithm to multiple 2D barcode regions, and the experimental results also showed that the contour-detection-based barcode region marking method is better than the connected domain analysis method, but its marking time is affected by the resolution size of a barcode image.

Effective extraction of barcode areas is a prerequisite for obtaining barcode information. Lin and Fuh [13] used the position detection patterns of QR codes to extract QR codes that could recognize different types of QR code images. In the literature [14], a QR code extraction algorithm based on the Hough transform [15] is proposed. The algorithm achieves QR code extraction by forming a bitmap with edge detection and Hough space and preserving the intersection of vertical line segments. Chen [16] provided an algorithm to effectively extract the tilted 1D barcode region. It first obtains the 1D barcode edge information in the image by Canny edge detection [17], then detects the straight lines inside the barcode using the Hough transform, obtains the tilt angle of the 1D barcode region, and finally rotates the extracted barcode to a horizontal state. The above barcode region extraction algorithm can effectively extract tilted single-type barcodes, but it does not consider the extraction of multitype barcodes, and the process requires high time cost due to complex preprocessing, especially in high-resolution images.

The development of deep learning and machine learning has inspired many new barcode positioning algorithms. Zamberletti et al. [18] introduced a machine learning-based barcode detection method in which the algorithm effectively recognizes 1D barcodes. Katona et al. [19] used a vector machine to learn the characteristics of QR codes and use them to extract QR barcodes. Unfortunately, none of the above algorithms take into account multitype barcodes. Ren et al. [20] provided a fast region-based convolutional neural network (CNN) model that can quickly detect and classify multiple targets. Based on this model, Tian et al. [21] developed a barcode detector called BAN, which can quickly detect multiple types of barcodes in barcode images. Unfortunately, it has only been tested in images with resolution of 1024×768 pixels. Jia et al. [22] developed a multitype barcode recognition model based on an improved CNN and trained it using images with resolution of 1920×1080 pixels. Additionally, another focus of this model is that the accurate position and distorted barcode shape can be determined and corrected. The performance of these localization algorithms is tied to the quality and quantity of training data and the training time of the model.

Based on the above research, the problems identified are high computational cost, restricted to low-resolution images, limited to barcode types, etc. In order to address these problems, a reliable localization method for multitype barcodes in high-resolution images is proposed. The main contributions of these works can be summarized as follows:

(1) Effectively reduces time costs and further improves barcode positioning accuracy.

(2) The proposed algorithm can perform barcode localization in high-resolution images and has excellent performance.

(3) Efficiently extracts arbitrarily tilted barcodes and provides good horizontal barcodes for decoders.

(4) The proposed method can locate 1D barcodes, 2D barcodes, and multitype barcodes.

2. Methods

Since the majority of current barcode localization algorithms are confined to a single kind of barcode and the high computational complexity is limited to low-resolution pictures, there is still room for improvement in their localization performance and efficiency. Aiming at the problems mentioned above, this paper proposes a barcode positioning method for high-resolution images. The method includes three main modules: feature extraction of the barcode edge, labeling of the barcode region, and extraction of the barcode region. The barcode positioning method is shown in Figure 1.

2.1. Feature Extraction of the Barcode Edge.
Effective extraction of edge features in the barcode region is a prerequisite for barcode localization. Nevertheless, existing techniques are only appropriate for extracting a single type of barcode feature or are limited to low-resolution images due to the high complexity of the algorithm. Therefore, this work proposes a simple and quick algorithm, namely joint edge detection.

2.1.1. Grayscale Conversion.
The input image is a high-resolution image with three red, green, and blue (RGB) channels. It not only fails to reflect the features of the barcode region but also leads to data redundancy. To reduce data redundancy and improve calculation efficiency, images
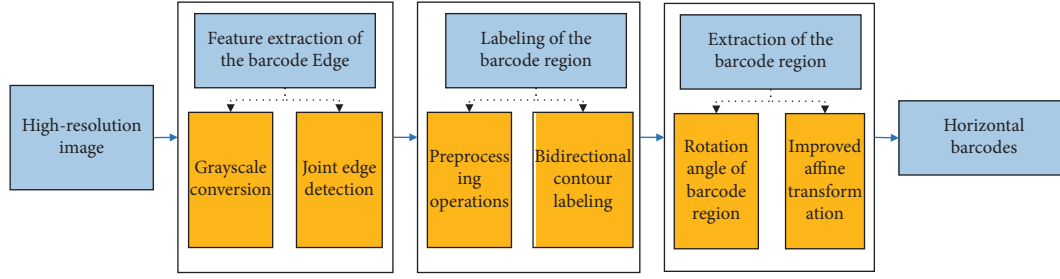
FIGURE 1: The barcode positioning method.

are first converted to single-channel grayscale images by the following equation:

$$G(x, y) = 0.299 \times r(x, y) + 0.587 \times g(x, y) + 0.114 \times b(x, y), \tag{1}$$

where r, g, and b represent the pixel value of red, green and blue channels of the image at the (x, y) position, respectively. Through this operation, the high-resolution image is converted into a grayscale image, which effectively keeps the edge feature information of the barcode and reduces data redundancy.

2.1.2. Joint Edge Detection. The joint edge detection algorithm proposed in this paper mainly detects the edge features of barcodes in all directions through horizontal, vertical, and double diagonal directions. Detection directions are shown in Figure 2.

The suggested method is divided into three stages, as follows:

In the beginning, the horizontal and vertical gradients $I_x(n)$ and $I_y(n)$ for each pixel in the greyscale image G are computed in the horizontal and vertical directions. Then, they are combined in a nonlinear manner, and their absolute values are calculated. Finally, a horizontal-vertical edge gradient map $I_e(n)$ is generated by equation (2). This approach fully detects the barcode with horizontal (vertical) edge features and improves the robustness of the barcode horizontal (vertical) edge feature extraction direction.

$$I_e(n) = \left| |I_x(n)| - |I_y(n)| \right|. \tag{2}$$

Then, the barcodes with horizontal (vertical) edge features are extracted in the gradient map $I_e(n)$, and all the extracted barcode regions D_i are removed by equation (3) in the grayscale image G, forming a new grayscale image G_{new}, where i is the number of extracted barcode regions.

$$G_{new}(x, y) = G(x, y) - D_i(x, y). \tag{3}$$

Finally, 45-degree and 135-degree gradients $I_{45}(n)$ and $I_{135}(n)$ for each pixel n in the new greyscale image G_{new} are computed in double diagonal directions. Then, they are combined in a nonlinear manner, and their absolute value is calculated. Finally, a double diagonal edge gradient map $I_d(n)$ is generated by equation (4). Through this step, the edge features of inclined barcodes and 2D barcodes can be

effectively detected, and the robustness of the extraction direction of the edge features of inclined barcodes and 2D barcodes can also be improved.

$$I_d(n) = \left| |I_{45}(n)| - |I_{135}(n)| \right|. \tag{4}$$

The edge feature detection results of the high-resolution image through the joint edge detection algorithm are shown in Figure 3, where it can be seen from Figures 3(b) and 3(c) that the edge features of horizontal (vertical) barcodes, inclined barcodes, and 2D barcodes are effectively detected by this method. The higher the energy, the better the effect of barcode edge feature detection.

2.2. Labelling of the Barcode Region. The labeling of the barcode region is an important process for accurate barcode localization. The traditional connected-component marking approach requires that all pixels in the target region be marked. The efficiency is low, and the cost of marking time is heavily influenced by image resolution. Therefore, a new connected-component method has been developed to improve efficiency.

2.2.1. Preprocessing Operations. In order to reduce the influence of small background clutter and weak target areas in gradient maps $I_e(n)$ and $I_d(n)$, a block filter of size 35×35 over gradient maps $I_e(n)$ and $I_d(n)$ is used to reduce noise and improve the weak barcode region. The size of the filter was chosen based on the range of the resolution of the input images by our method. It is worth noting that block filtering can be implemented efficiently so that only few operations per pixel are required. In addition, the thresholds are determined by taking the adaptive thresholding approach of Otsu [23]. The binary images $B_e(n)$ and $B_d(n)$ are obtained by equation (5) after the thresholds T_e and T_d are obtained, and morphological techniques are utilized to fill in the gaps in the barcode feature region.

$$B_e(n) = \begin{cases} 1, & \text{if } I_e(n) \geq T_e, \\ 0, & \text{otherwise,} \end{cases} \text{ or } B_d(n) = \begin{cases} 1, & I_d(n) \geq T_d, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

In this formula, $I_e(n)$ is the horizontal-vertical gradient map, $I_d(n)$ is the diagonal gradient map. The preprocessing results of $I_e(n)$ and $I_d(n)$ are shown in Figures 4(b) and 4(c), respectively.
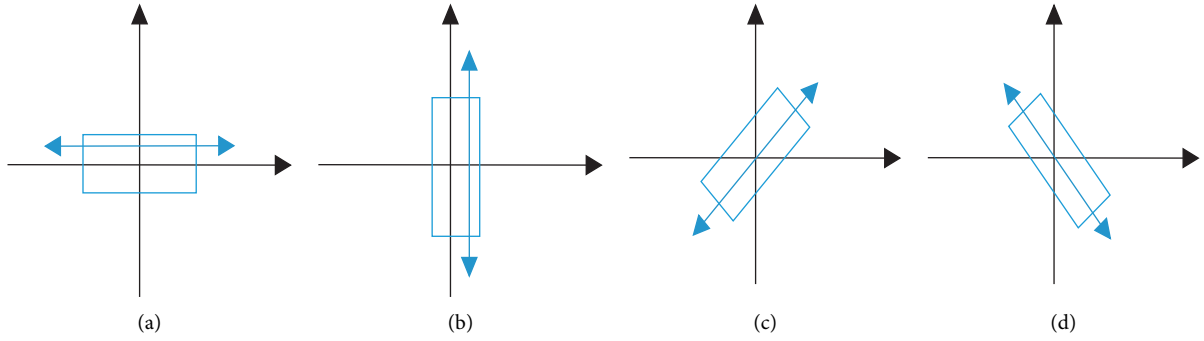
FIGURE 2: Detection directions: (a) horizontal, (b) vertical, (c) 45-degree, and (d) 135-degree.
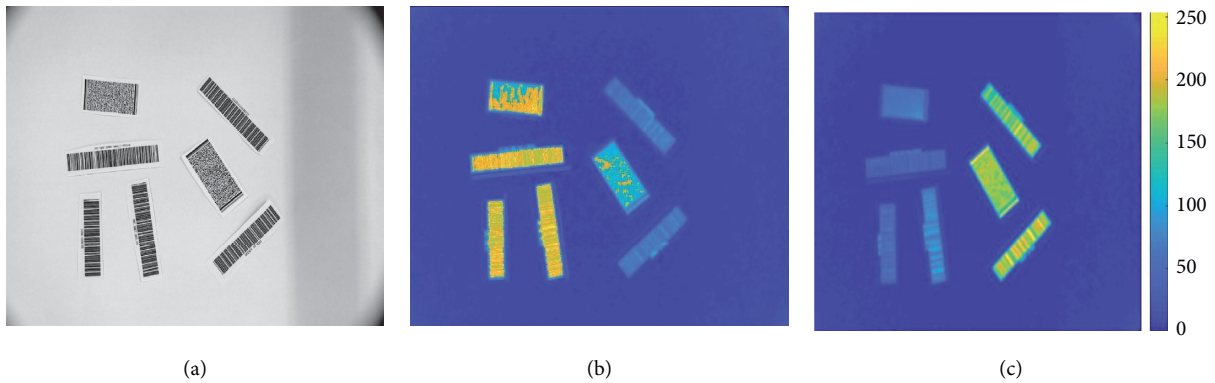

FIGURE 3: Edge feature detection results: (a) high-resolution image, (b) the corresponding energy for $I_e(n)$, and (c) the corresponding energy for $I_d(n)$.
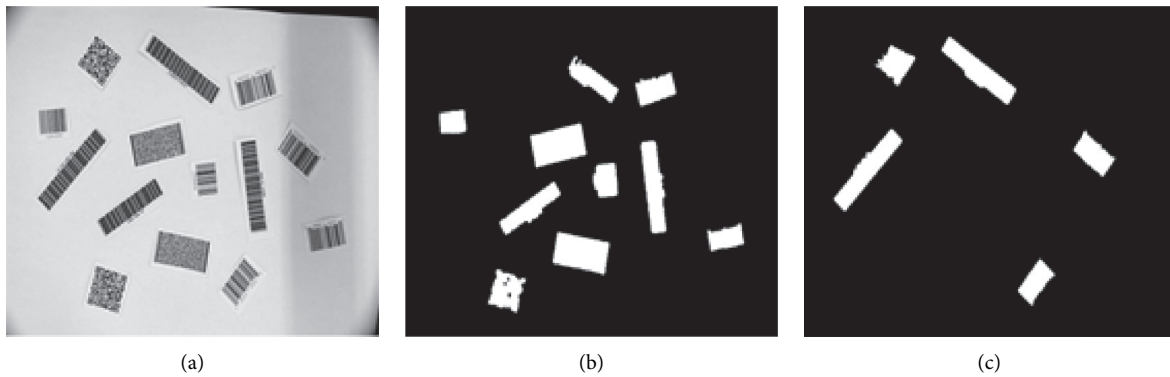

FIGURE 4: Preprocessing results: (a) high-resolution image, (b) preprocessing result of the horizontal-vertical gradient map, and (c) preprocessing result of the diagonal gradient map.

2.2.2. Bidirectional Contour Labelling. The traditional approach is abandoned in this section, and a bidirectional labeling method based on contour information is proposed. This method tracks and labels the outer contour of each barcode region clockwise and counterclockwise, as realized by parallel computing [24] on the CPU. The bidirectional contour labeling method consists of five steps used iteratively to examine all the pixels in the image.

In Step 1, the current pixel P_0 is the starting point of the outer contour of the new barcode region if it is a foreground point and the prior pixel P_{-1} is a background point, as illustrated in Figure 5(a). After the contour beginning point is found, the beginning and middle points P_S and P_C can be set as $P_S = P_{-1}$ and $P_C = P_0$, respectively, for initialized contour tracking.

In Step 2, the next contour point P_N is tracked clockwise and marked, with P_S as the starting point of clockwise tracking and P_C as the center point of eight-neighborhoods, as shown in Figure 5(b).

In Step 3, the next contour point P_N is tracked counterclockwise and noted, with P_S as the starting point of
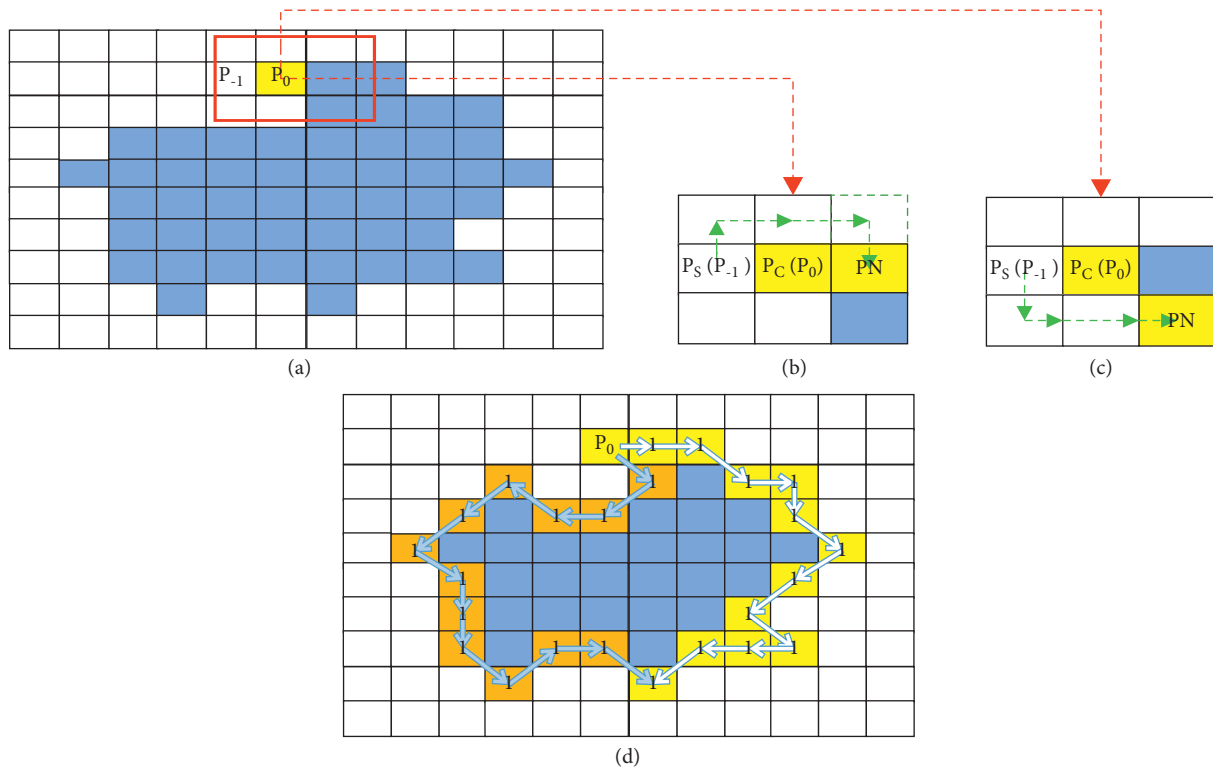
FIGURE 5: Bidirectional contour labelling process: (a) target region, (b) clockwise contour tracking, (c) counterclockwise contour tracking, and (d) marking results of the target region.

counterclockwise tracking and P_C as the center point of eight-neighborhoods, as presented in Figure 5(c).

In Step 4, the starting and center points P_S and P_C can be set as $P_S = P_C$ and $P_C = P_N$, respectively, for clockwise tracking (counterclockwise tracking) when a new contour point P_N is tracked clockwise (counterclockwise) and marked.

In Step 5, the parallel computing technique is adopted to achieve simultaneous contour tracking and marking both clockwise and counterclockwise. The method proceeds to determine the starting point of the next contour if clockwise and counterclockwise tracking to the pixel point P_N has been marked, indicating that the contour tracking and labeling of the current barcode region have been finished. Figure 5(d) shows the outcome of the bidirectional contour marking approach in the target region.

2.3. Extraction of the Barcode Region. In order to improve decoding efficiency, the marked barcode areas need to be extracted. Since the existing barcode extraction technology is restricted to a particular type of barcode extraction, a quick and simple extraction algorithm is proposed in this section. The procedure consists of two steps: calculating the rotation angle of the inclined barcode and rotating the inclined barcode region to a horizontal state.

2.3.1. Rotation Angle of Barcode Region. Based on the rectangular shape of the barcode region, a maximum outer rectangle is fitted to the marked barcode region, and the fitted barcode region is extracted from the binary image

$B(n)$. The rotation operation with the extracted barcode region instead of the entire image can effectively reduce the data volume and improve the computation speed.

The minimum outer rectangle is fitted to the extracted barcode region, and the tilt angle θ of the barcode region is calculated based on the fitted minimum outer rectangle. The rotation angle for rotating the tilted barcode area to a horizontal state is calculated by the following two steps:

In Step 1, based on the width W and the height H of the minimum outer rectangle, the width-to-height ratio R of the minimum outer rectangle is calculated by the following equation:

$$R = \frac{W}{H}. \tag{6}$$

In Step 2, the rotation angle β is determined by the relationship between the ratio R and the threshold values R_1 and R_2, which are empirical values herein, as shown in the following equation:

$$\beta = \begin{cases} \theta, & R \geq R_1, \\ 90° + \theta, & 0R \leq R_2, \quad \theta \in [-90°, 0]. \\ \theta, & R_2RR_1. \end{cases} \tag{7}$$

In this formula, R_1 and R_2 are set to 1.2 and 0.9, respectively, based on experience, and the tilt angle θ is the angle of the first side of the minimum outer rectangle rotating clockwise and touching the X-axis of the coordinate system. The above side is the width W of the smallest outer rectangle, and the adjacent side is the height H, as shown in Figure 6. This method fully guarantees that the long side of
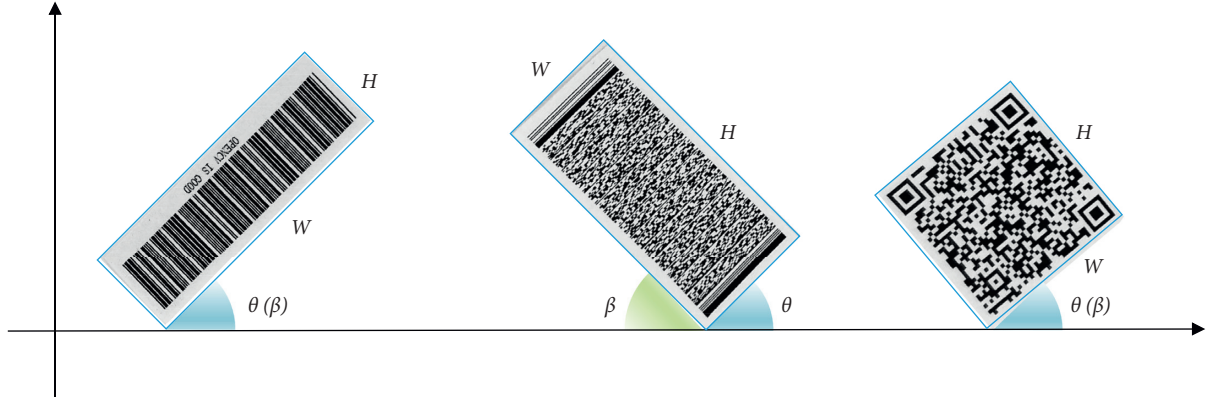
FIGURE 6: Schematic diagram of the width (W), height (H), tilt angle θ, and rotation angle β of the smallest outer rectangle.

the barcode is horizontal and the short side of the barcode is vertical after rotating β with any tilt barcode.

2.3.2. Improved Affine Transformation.

The tilted barcode area is rotated to a horizontal state around the center point of the barcode area by the affine transformation, where the rotation matrix A and translation vector B required for the affine transformation are determined by the rotation angle β and the center of rotation. The rotation relationship is as follows:

$$
\begin{bmatrix} x\prime \\ y\prime \end{bmatrix} = A \begin{bmatrix} x \\ y \end{bmatrix} + B,
$$

$$
B = \begin{bmatrix} (1 - \cos \beta) \times C_x - \sin \beta \times C_y \\ \sin \beta \times C_x + (1 - \cos \beta) \times C_y \end{bmatrix}. \tag{8}
$$

In this formula, $A = \begin{bmatrix} \cos \beta & \sin \beta \\ -\sin \beta & \cos \beta \end{bmatrix}$, C_x and C_y are the horizontal and vertical coordinates of the center point of the barcode area, (x, y) are the horizontal and vertical coordinates of the pixel point to be rotated, and (x', y') are the horizontal and vertical coordinates of the pixel point after rotation.

However, as the size and the center of the barcode area are changed after rotation, the traditional affine transformation for rotation will result in missing or incomplete edges of the barcode area. To address this shortcoming, the algorithm is improved in this paper to recalculate the barcode area size and the new translation vector after the barcode area rotation by the following equations:

$$
\begin{cases} W' = W|\cos \beta| + H|\sin \beta|, \\ H' = W|\sin \beta| + H|\mathrm{con}\,\beta|, \end{cases} \tag{9}
$$

$$
B = \begin{bmatrix} (1 - \cos \beta) \times C_x \sin \beta \times C_y \\ \sin \beta \times C_x + (1 - \cos \beta) \times C_y \end{bmatrix} + \begin{bmatrix} \dfrac{W' - W}{2} \\ \dfrac{H' - H}{2} \end{bmatrix}, \tag{10}
$$

where W and H are the width and the height of the barcode area before rotation and W' and H' are the width and the

height of the barcode area after rotation. The results of barcode region extraction are shown in Figure 7, where the red and green boxes in Figure 7(a) are the results of fitting the maximum outer rectangle and the minimum outer rectangle, respectively. The results show that the proposed algorithm can extract arbitrarily skewed multitype barcode regions in high-resolution images and rotate them to a horizontal state, providing good barcode images for decoders.

3. Results and Discussion

3.1. Experimental Environment and Datasets.

In this paper, the proposed algorithm is simulated using the OpenCV framework and C++, and the barcode images are captured using a high-resolution camera. The barcode images are divided into three data sets as follows:

(A) Dataset A contains 900 low-resolution images (600 images of a single type of barcode and 300 images of multiple types of barcodes), where the barcode image resolution is 1024×768 pixels, and each image has an average of 12 barcodes.

(B) Dataset B contains 1000 high-resolution images (600 images of a single type of barcode and 400 images of multiple types of barcodes), where the barcode image resolution is 4208×3120 pixels, and each image has an average of 14 barcodes.

(C) Dataset C contains 400 barcode images of multiple types, where the resolution and number of barcodes are shown in Table 1.

To verify the performance of the proposed algorithm, this paper compares the accuracy and time cost of barcode localization with those of the traditional localization algorithms.

3.2. Accuracy of Positioning.

The accuracy of barcode positioning includes single type barcodes and multiple type barcodes in both low-resolution images and high-resolution images. The details are as follows:

First, the localization accuracy of the proposed algorithm for barcodes in low-resolution images is tested in Dataset A. Tables 2–4 show the accuracy of the proposed algorithm
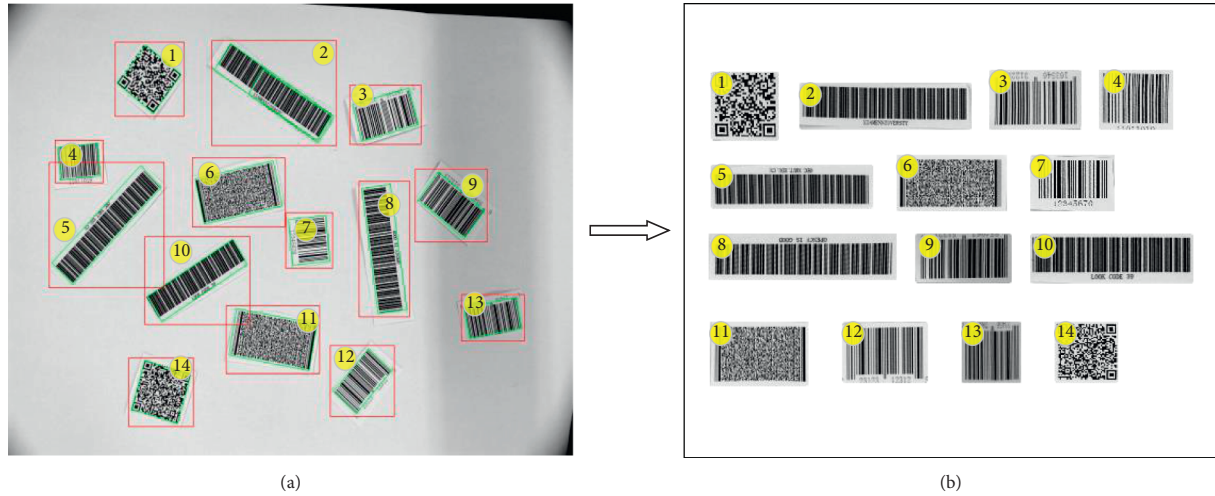
(a)

(b)

FIGURE 7: Extraction results: (a) fitting results of barcode area and (b) horizontal status barcode area.

TABLE 1: Time cost for locating multiple types of barcodes in low-resolution images.

Methods	Resolution	Number of barcodes	Time cost (ms)	Proposed (ms)
Chen et al. [11]	300 × 400	3–5	230	98
D. T. Lin and C. L. Lin [7]	640 × 480	3–5	256	165
Katona and Nyúl [8]	720×480	3–5	310	176
Katona et al. [19]	1024 × 768	3–5	550	233

TABLE 2: Localization accuracy of 1D barcodes in low-resolution images.

Barcode type	Chen et al. [11]	Katona and Nyúl [8] (%)	D. T. Lin and C. L. Lin [7] (%)	Proposed (%)
Code 128	91.07	84.8	94.64	96.30
Code 39	96.43	90.5	100	97.41
UPC-A	90.7	95.4	95.35	99.24
EAN 13	94.55	99.02	98.18	99.36
EAN 18	90	95.9	92.43	98.05
Average	92.55	93.16	96.12	98.07

TABLE 3: Localization accuracy of 2D barcodes in low-resolution images.

Barcode type	Ren et al. [20]	Tian et al. [21] (%)	D. T. Lin and C. L. Lin [7] (%)	Proposed (%)
QR code	92.8	93.3	96.52	98.28
PDF 417	93	96	91.04	97.25
Data matrix	93.2	94.4	97.67	98.32
Average	93	94.57	95.08	97.95

TABLE 4: Localization accuracy of multitype barcodes in low-resolution images.

Methods	Katona and Nyúl [8]	Chen et al. [11] (%)	Tian et al. [21] (%)	D. T. Lin and C. L. Lin [7]	Proposed (%)
Accuracy	92.97	94.49	94.66	96.52	97.82

for 1D, 2D, and multitype barcode localization and compare the experimental results of such a method with those of D. T. Lin and C. L. Lin, Katona and Nyúl, Chen, Ren et al. and Tian et al. [7, 8, 11, 20, 21], respectively. The experimental results show that the proposed algorithm has an average accuracy of 98.07% in 1D barcode localization, 97.95% in 2D barcode localization, and 97.82% in multitype barcode localization. From the data in the tables, it can be seen that the proposed algorithm can effectively locate both single type and multitype barcodes in low-resolution barcode images and that its

TABLE 5: Localization accuracy of 1D barcodes in high-resolution images.

Barcode type	Number of barcodes	Number of positioning	Locate rate (%)
Code 128	884	872	98.64
Code 39	735	725	98.64
UPC-A	916	910	99.34
EAN 13	940	931	98.04
EAN 18	1230	1223	99.42
Total	4705	4661	99.06

TABLE 6: Localization accuracy of 2D barcodes in high-resolution images.

Barcode type	Number of barcodes	Number of positioning	Locate rate (%)
QR code	873	871	99.77
PDF 417	690	686	99.42
Dara matrix	772	771	99.87
Total	2335	2328	99.70

TABLE 7: Localization accuracy of multitype barcodes in high-resolution images.

Barcode type	Number of barcodes	Number of positioning	Locate rate (%)
Multi-type	3056	2996	98.04

localization performance is better than that of the existing barcode localization algorithms.

Second, the localization accuracy of the proposed algorithm for barcodes in high-resolution images is tested in Dataset B. Tables 5–7 show the accuracy of the proposed algorithm for 1D, 2D, and multitype barcode localization. From the data in the table, it can be seen that the proposed algorithm can effectively locate both single type and multiple type barcodes in high-resolution barcode images with high accuracy. Specifically, the average positioning accuracy of 1D, 2D, and multitype barcodes was 99.06%, 99.70%, and 98.04%, respectively.

3.3. Time Cost of Positioning.
Effectively reducing the time cost of barcode positioning is another focus of the proposed algorithm. The time cost of barcode positioning includes multiple types of barcodes in low-resolution images and high-resolution images, respectively. The details are as follows:

Firstly, the time cost of the proposed algorithm for multiple types of barcode localization in low-resolution images is tested in Dataset C, where the barcode image resolution and the number of barcodes correspond to those provided by Lin, Katona, and Chen. As shown in Table 1, the time costs of the proposed algorithm are 98 ms, 165 ms, 176 ms, and 233 ms, respectively, which are lower than the time costs of the conventional localization algorithms, i.e., 230 ms, 256 ms, 310 ms, and 550 ms, respectively. It can be seen from the data in the table that the proposed algorithm can effectively reduce the time cost of multitype barcode positioning in low-resolution images by 36%–57% compared to the traditional method.

Secondly, the multitype barcode images in Dataset B are converted to five different resolutions, which are 526×390 pixels, 1052×780 pixels 2104×1560 pixels, 3208×2120 pixels, and 4208×3120 pixels, respectively. Meanwhile, the time cost
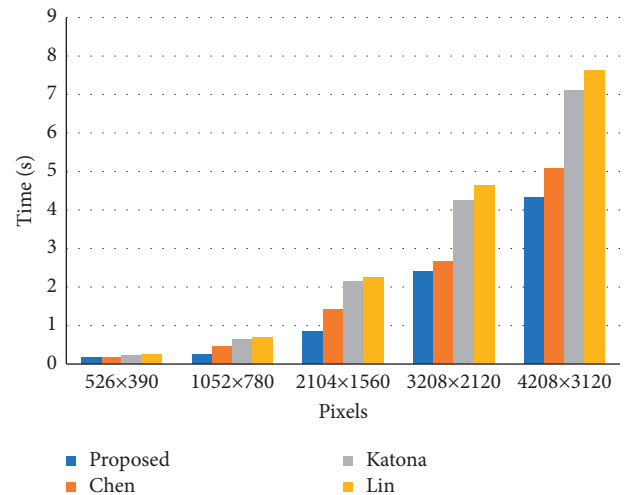

FIGURE 8: Time cost of barcode positioning at different resolutions.

of the proposed algorithm to locate the multitype barcodes in the high-resolution images is being tested. The test results are shown in Figure 8. It is clear that the time cost of barcode positioning is greatly influenced by the image resolution and the number of barcodes to be positioned, but the increase in the time cost of the proposed algorithm is lower than that of existing methods. On the other hand, the time cost of existing algorithms is higher than that of the proposed algorithm at different resolutions. The experimental results show that the proposed algorithm can effectively reduce the time cost of barcode positioning and improve the efficiency of barcode positioning, especially in high-resolution images.

Through the above experimental analysis, the accuracy of the barcode localization algorithm proposed in this paper in locating 1D barcodes, 2D barcodes, and multitype barcodes in
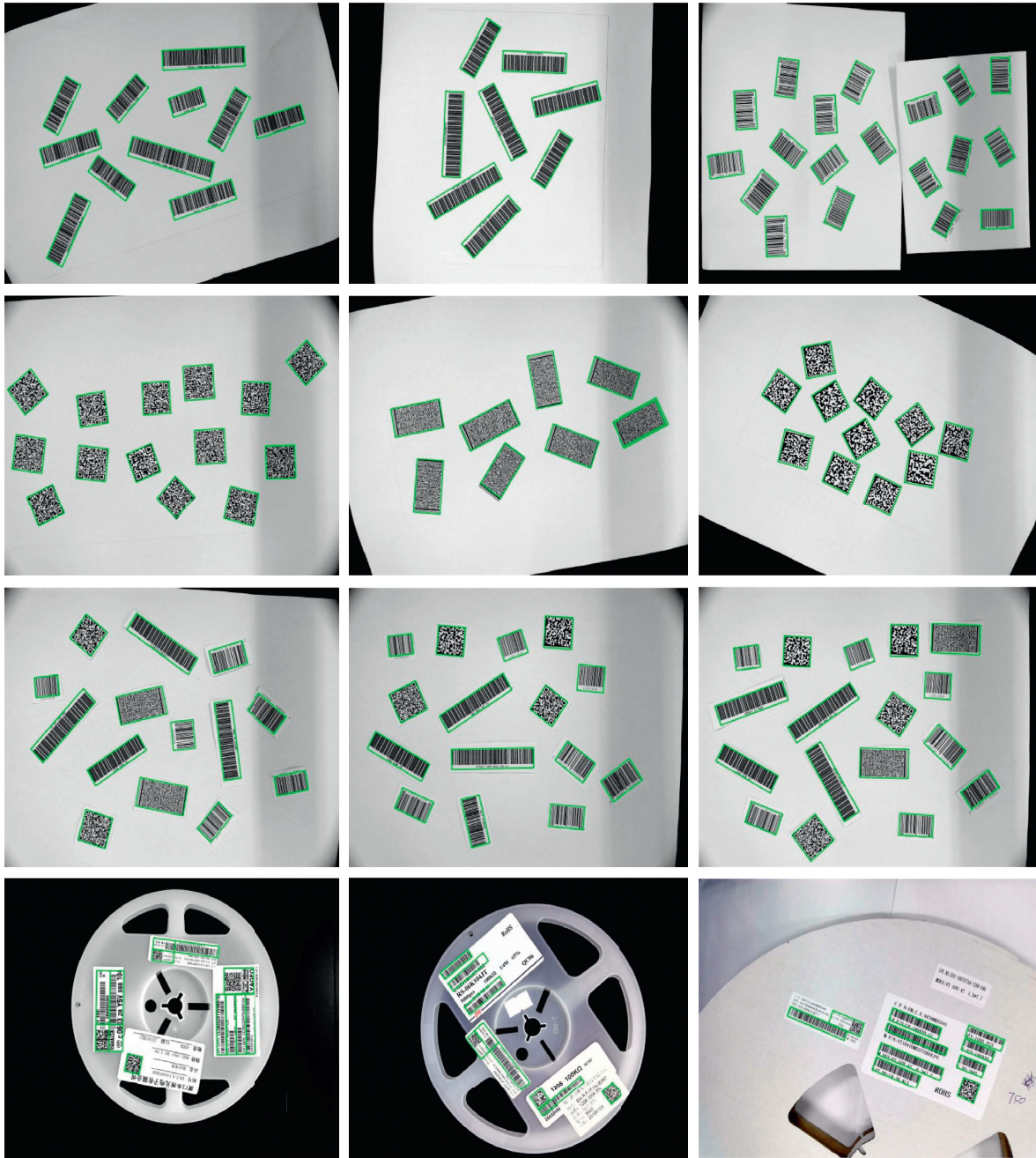
FIGURE 9: Barcode positioning effect in high-resolution images.

low-resolution images is higher than that of existing algorithms. Secondly, it can effectively locate single-type barcodes and multitype barcodes in high-resolution images, and some of the localization effects are shown in Figure 9. Finally, the time cost of the proposed algorithm is lower than that of traditional algorithms for multitype barcode localization in both low-resolution images and high-resolution images, which can effectively reduce time cost and improve localization efficiency.

4. Conclusions

In this study, a new reliable localization method is proposed to extract arbitrary tilting real barcodes from one image,

especially when 1D, 2D, or multitype barcodes are present in high-resolution images. The proposed method mainly comprises three steps: first, detection of barcode edge features; second, marking of barcode regions; and finally, extraction of barcode regions. In the proposed method, the edge features of arbitrary barcodes are quickly extracted by a joint edge detection algorithm, and potential barcode regions are efficiently marked by using a bidirectional contour marking method. Finally, the problem of tilted barcode extraction is solved by an improved affine transformation. The experimental results show that the proposed method can effectively locate multitype barcodes with higher accuracy in both low-resolution and high-resolution images. On the

other hand, the proposed method can effectively reduce the time cost of barcode localization and further improve its efficiency, especially for multitype barcodes in high-resolution images.

Data Availability

The datasets used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest.

Acknowledgments

References

[1] S. M. Youssef and R. M. Salem, "Automated barcode recognition for smart identification and inspection automation," *Expert Systems with Applications*, vol. 33, no. 4, pp. 968–977, 2007.

[2] D. H. Chen and H. Liu, *Barcode Technology and Applications*, pp. 1–10, Chemical Industry Press, Beijing, China, 2015.

[3] O. Gallo and R. Manduchi, "Reading 1D barcodes with mobile phones using deformable templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1834–1843, 2011.

[4] I. Y. Yun and J. Kim, "Vision-based 1D barcode localization method for scale and rotation invariant," in *Proceedings of the of the 2017 IEEE Region 10 Conference (TENCON)*, pp. 2204–2208, Penang, Malaysia, November 2017.

[5] K. C. A. FabijbDska, "Detection of QR-Codes in digital images based on histogram similarity," *Image Processing and Communications*, vol. 20, no. 2, pp. 41–48, 2015.

[6] O. L. Rincon, O. Starostenko, A. A. Vicenteand, and J. C. Galan-Hernandez, "Binary large object-based approach for QR code detection in uncontrolled environments," *Journal of Electrical and Computer Engineering*, vol. 2017, Article ID 4613628, 15 pages, 2017.

[7] D. T. Lin and C. L. Lin, "Automatic location for multi-symbology and multiple 1D and 2D barcodes," *Image Processing and Communications*, vol. 21, no. 6, pp. 663–668, 2013.

[8] M. Katona and L. G. Nyúl, "Efficient 1D and 2D Barcode Detection Using Mathematical morphology," in *Proceedings of the Mathematical Morphology And its Applications To Signal And Image Processing*, pp. 464–475, Uppsala, Sweden, May 2013.

[9] R. O. Duda and P. E. Hart, *Classification and Scene Analysis*, pp. 271-272, Wiley, New York, NY, USA, 1973.

[10] F. Chang, C.-J. Chen, and C.-J. Lu, "A linear-time component-labeling algorithm using contour tracing technique," *Computer Vision and Image Understanding*, vol. 93, no. 2, pp. 206–220, 2004.

[11] Y. Chen, Z. X. Yang, and Z. F. Bai, "Simultaneous Real-Time Segmentation of Diversified Barcode Symbols in Complex Background," in *Proceedings of the First International Conference On Intelligent Networks And Intelligent Systems*, pp. 527–530, Wuhan, China, November 2008.

[12] Q. L. Liu, X. C. Li, M. Zou, and Z. Jun, "The Multi-QR Codes Extraction Method in Illegible Image Based on Contour Tracing," in *Proceedings of the 2011 IEEE International Conference on Anti-Counterfeiting, Security and Identification*, pp. 51–56, Xiamen, China, June 2011.

[13] J. A. Lin and C. S. Fuh, "2D barcode image decoding," *Mathematical Problems in Engineering*, vol. 2013, Article ID 848276, 10 pages, 2016.

[14] M. Dubská, A. Herout, and J. Havel, "Real-time precise detection of regular grids and matrix codes," *Journal of Real-Time Image Processing*, vol. 11, no. 1, pp. 193–200, 2016.

[15] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, 1972.

[16] J. M. Chen, "*One-dimension Barcode Localization Algorithm with Complex Background*," M.S. thesis, Zhejiang University, Hangzhou, China, 2015.

[17] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[18] A. Zamberletti, I. Gallo, and S. Albertini, "Robust Angle Invariant 1D Barcode Detection," in *Proceedings of the 2013 Second IAPR Asian Conference on Pattern Recognition (AVPR)*, pp. 160–164, Naha, Japan, November 2013.

[19] M. Katona, P. Bodnár, and L. G. Nyúl, "Distance transform and template matching based methods for localization of barcodes and QR codes," *Computer Science and Information Systems*, vol. 17, no. 1, pp. 161–179, 2019.

[20] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[21] Y. Tian, Z. H. Che, G. T. Zhai, and Z. Gao, "BAN, A barcode accurate detection network," in *Proceedings of the 2018 IEEE Visual Communications and Image Processing (VCIP)*, pp. 1–5, Taichung, Taiwan, China, December 2018.

[22] J. Jia, G. Zhai, J. Zhang et al., "EMBDN: an efficient multiclass barcode detection network for complicated environments," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9919–9933, 2019.

[23] N. Ostu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[24] D Darriba, G. L Taboada, R Doallo, and D. Posada, "JModelTest 2: more models, new heuristics and parallel computing," *Nature Methods*, vol. 9, no. 8, p. 772, 2012.