ISSN 2347 - 3983
Volume 3, No.4, April 2015
International Journal of Emerging Trends in Engineering Research
Available Online at http://www.warse.org/ijeter/static/pdf/file/ijeter02252015.pdf

License Plate Character Recognition using Advanced Image Processing Techniques and Genetic Algorithm

Dr. L.M. Varalakshmi[1], Radha Ramalingam[2]
[1]Dept. of ECE, Sri Manakula Vinayagar Engg. College, Madagadipet, Puducherry, India – 605107.
Email: varalakshmi-1@yahoo.co.in
[2] Dept. of ECE, Sri Manakula Vinayagar Engg. College, Madagadipet, Puducherry, India – 605107.
Email: radha.vipsha@gmail.com

ABSTRACT

Dynamic image processing techniques coupled with Genetic Algorithm are used for recognizing the license plate numbers from an image containing it. Recognition of license plate in a picture which is prone to illumination problems is done using this process. The proposed system finds multi style number plates in an image by using Multistyle License Plate Number Using genetic algorithm and Dynamic Image Processing techniques. The license plates detection stage is the most is the most critical step in the automatic license plate identification system. Many researches have been done to overcome all the challenges faced in this area, but no general method is best suitable for detection of license plates models from different countries or places. This is because each country has different plate style and design. All existing techniques or algorithms available can be classified based on the features used for the detection. Different existing algorithms that have been researched are Color-based techniques that use specific fixed color coding used by different countries. Another widely used technique is External-shape based detection which detects plates based on its shape ratio as prescribed by different countries.

Key words: Image processing; genetic algorithm; license plate; CCAT

1. INTRODUCTION

The term digital image refers to processing of a two dimensional picture by a digital computer. In a broader context, it implies digital processing of any two dimensional data. A digital image is an array of real or complex numbers represented by a finite number of bits. An image given in the form of a transparency, slide, photograph or an X-ray is first digitized and stored as a matrix of binary digits in computer memory. This digitized image can then be processed and/or displayed on a high-resolution television monitor. For display, the image is stored in a rapid-access buffer memory, which refreshes the monitor at a rate of 25 frames per second to produce a visually continuous display.

2. LITERATURE SURVEY

In earlier researches the idea of using edge geometrical features is used in detecting these license plates. The edge part is obtained using the Difference of Gaussian operation followed by Sobel vertical edge mask. Before doing that, the gamma correction is applied to increase chances of detecting the edges. After this the morphological operations are applied to get the plate region candidates. Using these regions, together with the edge image identified previously, we calculate geometrical features of these regions and use rule-based classifier to identify the actual plate region [1]. In this paper we present a proposal to solve the problem of license plate recognition using a three layer fuzzy neural network. In the first stage the plate is detected inside the digital image using rectangular perimeter detection and the finding of a pattern by pattern matching, after that, the characters are extracted from the plate by means of horizontal and vertical projections [2]. An algorithm for detecting license plates that can detect multiple license plates having different sizes in very much unfamiliar and complex background. Detecting the License plates is an important processing step in recognizing the license plates which has many applications in transportation systems. Vertical edges and edge density are utilized to find candidate regions [3]. The crossover operator, Sequential Constructive crossover (SCX) technique, for genetic algorithm is the technique that generates solution of very high quality to the Traveling Salesman Problem (TSP). The constructive crossover uses a pair of parents to create an offspring using better edges based on their values that may be present in the parents' structure maintaining the sequence of nodes in the parent chromosomes. The efficiency of the Sequential constructive crossover is compared with some existing crossover operators; like, generalized N-point crossover (GNX) and edge recombination crossover (ERX) for some benchmark TSPLIB instances [4]. In ITS, car license plate detection is a basic task. A Maximally Stable External Region (MSER) license plate detection method is proposed in the paper constrained by some prior knowledge of license plate. Initially the input image is processed and converted to grayscale using gray scaling and gray stretching, etc., and then, the candidate MSER license plate regions are chosen according to the pixel sum, length-width ratio of the license plate character region, after that, similar single-character regions are removed and the upper and lower borders of the license plate are determined, using horizontal projection of candidate license plate character region and the grey level jump constraints, finally, the right and left borders of license plate are determined by vertical projection [5].

3. EXISTING SYSTEM

Usually the car plates appear in different types of character styles, either single or double row, different sizes, spacing and character counts. Due to such kind of variations even localizing or detecting these plates a difficult problem. The problem of localization will be much harder during night time due to poor lighting conditions. In this paper, edge geometrical feature is being used for detecting the license plates. The edge part is got using Difference of Gaussian operation followed by Sobel vertical edge mask. Before that, gamma correction is applied to the image to increase the chances of detection of edges. After this we apply morphological operations to get the plate region candidates. Using these regions, along with the edge image, we calculate geometrical features of these regions and use rule-based classifier to identify the true plate region exactly.

Disadvantages of Existing System

- There are Color-based systems developed that detect specific plates having fixed colors or color coding.
- The drawback of this existing method is, it is sensitive to any model identification text or any other texts or objects that are present above or below the license plates that can disturb the texture histogram.
- The main drawback of these existing segmentation techniques was their huge computational demand and also they are sensitive to the presence of other texts such as model identification texts and bumper stickers.

4. PROPOSED SYSTEM

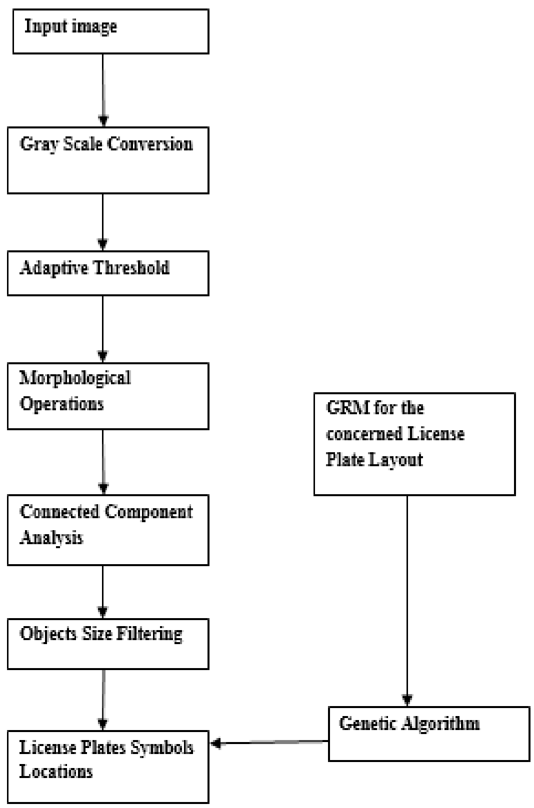
Use of a new genetic algorithm (GA) approach is introduced to detect the locations of license plate (LP) symbols. An adaptive threshold method is being used to overcome the errors happening due to changes of illumination conditions while converting the image into binary. To detect the candidate object inside the unknown image an image processing technique, connected component analysis technique (CCAT) is used. To simplify the system adaptability when applied in different countries, a scale-invariant geometric relationship matrix is introduced to model the layout of symbols in any LP. Along with that, two new crossover operators, based on sorting, are being used, which improve the convergence speed of the system very much. Most of the problems faced by CCAT techniques such as broken bodies, are minimized by modifying the Genetic Algorithm used to perform partial match until reaching an acceptable fitness value. The proposed system is implemented using MATLAB and different samples of license plate image are experimented with to verify the accuracy of the proposed system.

The most critical step in the process is the detection/identification stage of the license plate. Lot of research has been done to overcome the problems faced in this area, but there is no successful general method which could be used for detecting the license plates of different countries or places because each country or regions use different plate style or design. All the developed existing techniques can be classified according to the selected features upon which the detection algorithm was based. For plates having fixed colours or color coding, color-based systems have been built to detect specific plates. External-shape based techniques are being used to detect the plate based on its rectangular shape. With advancement in artificial intelligence and computer science, intelligent systems such as intelligent transportation systems play more and more important role in modern society. Among these systems license plate recognition is used in many applications including automatic toll payment, identification of stolen vehicles, border control, and traffic law enforcement. A license plate recognition system generally consists of three processing steps: license plate detection, character segmentation, and character recognition. There are many factors to be taken into account when developing license plate detection method. License plate standards vary from country to country. Images can be captured in different illumination conditions and may contain other objects such as buildings, people, trees, fences etc. Also the number of vehicles and the distance between the vehicle and the camera can vary. This makes license plate detection to be the most important and challenging step. For many years, moving object detection and location is a focus in the field of image processing, and its key technology is feature extraction and description. The most frequently used features are texture, geometrical configuration and gray feature. However, these features are often subject to different environment such as view point and distance. The MSER feature is affine-invariant to scale transformation, rotation transformation and transformation of the view-point, so it has great advantages over the common features in robustness, repetition rate, discrimination.

The proposed fitness is taken as the inverse of the objective distance calculated between the prototype chromosome and the existing chromosome.

In genetic algorithm, a population of candidate solutions to an optimization problem is evolved toward better solutions. In Every candidate solution some properties can be altered and mutated; traditionally, solutions are represented using binary strings i.e., 0s and 1s. But we can also represent the solutions using other encodings.

The evolution in genetic algorithm normally starts from a population of randomly generated candidates. This happens in an iterative process. The population of candidates in each iteration is called as generation. In each generation, every candidate's fitness will be evaluated; the fitness is normally the value of the objective function in the optimization problem being solved. The most fit individuals are selected from the current population, and each the genome of the individual is modified (mutated and recombined randomly) to form a new generation. The newer generation of candidates will be used in the next iterations. The iteration will come to an end when either a satisfactory fitness level has been reached or maximum number of generations has been produced for the population.



A genetic algorithm requires the following:

1. A fitness function to evaluate the solution domain.
2. A genetic representation of the solution domain,

A candidate solution is usually represented using as an array of bits. Arrays other types and structures can also be used in the same way. The most important property that makes these genetic representations most convenient is that their parts are aligned easily because of their fixed size, which makes simple crossover operations easier. Even variable length representations can be used but in this case the cross over implementation will be more complex. In genetic programming, Tree-like representations are explored and in evolutionary programming graph-form representations are explored; a combination of both linear chromosomes and trees is explored in gene expression programming. Once the genetic representation and the fitness function are defined, a Genetic Algorithm proceeds to initialize a population of solutions and iterate it thru repeated application of mutation, inversion, crossover and selection operations.

Usually a simple form of GA will represent each chromosome as binary string. Usually numeric parameters are represented using integer values. It is also possible to represent them using floating point representation. The floating point representation is natural to evolutionary programming and evolution strategies. Based on theoretical and experimental results (see below). The basic GA algorithm performs mutation and crossover at the bit level. Other kinds of algorithms treat the chromosome as any imaginable data structures like a list of numbers which are indexes into an instruction table, nodes in a linked list, hashes, objects, or any other imaginable data structure. Crossover and mutation are performed so as to respect data element boundaries. For most data types, variation operators can be designed. Different chromosomal data types might work better or become worse for different specific problem.

Gray coding is often employed when bit-string representations of integers are used. By using grey coding small changes in the integer can be readily affected through mutations or crossovers. This method helps to prevent premature convergence at so called Hamming walls, where too many simultaneous mutations must occur in order to change the chromosome to a better solution.

Using arrays of real-valued numbers instead of bit strings are also used to represent chromosomes. Results from the theory of schemata suggest that in general the smaller the alphabet, the better will be the performance, but good results were also obtained from using real valued chromosomes. This was described as the set of real values in a finite population of chromosomes as forming a virtual alphabet with a much lower cardinality than would be expected from a floating point representation.

Binarization

Converting the input license plate image into a binary image is one of the most important and sensitive stages in localizing the license plates. The main challenges faced during the conversion are caused due to temporal and spatial variations encountered in the plate itself and the environment around it which causes several illumination problems. Due to these issues, binarization of the image using a fixed global threshold method is not suitable to overcome these problems. In this system a local adaptive method has been used to determine the threshold at each pixel dynamically based on the average amount gray levels in the neighborhood pixels.

The AT technique is just an extension of Bradley and Roth's and Wellner's methods. The idea used in Wellner's algorithm is a pixel is compared with an average of neighboring pixels. Specifically, an approximate moving average of the last S pixels seen is calculated while traversing the image. If the value of the current pixel is T percent lower than the average, then it is set to black; otherwise, it is set to white. This technique best suited because comparing a pixel to the neighboring average pixels will keep hard contrast lines and ignore soft gradient changes. The advantage of this method is that only a single pass through the image is required. Wellner uses one eighth of the image width for the value of S and 0.15 for the value of T to yield the best results for a variety of images. The value of T might be a little bit modified from the proposed value by Wellner depending on the used images; whereas it should be in the range $0.1 < T < 0.2$ in our method.

Noise Objects Elimination

Dilation and erosion are the Morphological operations used in this method. These are vital processes that are required in most pattern recognition systems to eliminate noise and retain only objects that are expected to represent the targeted patterns. In License Plate detection, closing operation i.e., dilation followed by erosion is performed to fill holes with noise inside candidate objects and to connect broken characters/symbols.

Image morphology is a way of analyzing images based on shapes. In this study, we assume that the blood vessels are a tubule-like structure running along the length of the face. The operators used in this experiment are opening and top-hat segmentation, which are explained next. The opening operation is done to preserve foreground areas that have a similar shape to the structuring element or that can completely contain the structuring element, while eliminating all other regions of foreground pixels. The top-hat segmentation has two versions; but for our process, we use one of the version known as white top-hat segmentation as this process enhances the bright objects in the image. This operation can be defined as the difference between the input image and its opening. The selection of the top-hat segmentation is based on the fact that we desire to segment the regions associated with those of higher intensity, which demark the facial thermal signature. The task in this step is to enhance the maxima in the image. The top-hat segmented image is thus given by the basic idea in binary morphology is to look at an image with a simple, pre-defined shape, drawing conclusions on how this shape misses or fits the shapes in the image. This is called the structuring element, and is itself a binary image (i.e., a subset of the space or grid).

Encoding and Recognizing

For a complex object such as license plate, encoding is accomplished based on the objects contained within it. Since after plate detection the next step is to recognize the license plate number, the number of main symbols identifying the plate numbers should be minimum.

The simplest algorithm represents every chromosome in binary i.e., bit string. Typically, numeric parameters are represented using integers, even though it can be represented using floating points. The floating point representation is natural to evolution strategies and evolutionary programming. The real-valued genetic algorithm is really a contradictory because it does not really represent the building block theory that was proposed by John Henry Holland. This theory also has some support based on theoretical and experimental results (see below). The basic algorithm performs crossover and mutation at the bit level. The other variants treat the chromosome as a list of numbers which indexes into anything such as nodes in a linked list, instruction table, hashes, objects, or any other imaginable data structure. Crossover and mutation are done to respect data element boundaries. For most data types, specific variation operators can be designed. For different problem domains, each chromosomal data types seem to work better or worse.

When bit-string representations of integers are being used, usually gray coding is being used. Due to this, small changes in the integer can be readily affected through mutations or crossovers. This prevents premature convergence at so called Hamming walls, in which too many simultaneous mutations must occur to change the chromosome to a better solution.

Other approaches involve using list or array of real-valued numbers instead of using binary or bit strings to represent chromosomes. Results from the theory of schemata suggest that usually when the alphabets are smaller the performance gets better. Initially it was surprising to researchers that good results were obtained from using real-valued chromosomes. This was explained as the set of real values in a finite population of chromosomes as forming a virtual alphabet with a much lower cardinality than would be expected from a floating point representation.

1. **Position relationship:** This will be represented using the relative distances between the bounding boxes of the two objects in both X and Y directions.
2. **Size relationship:** This will be represented as the relative differences in their bounding boxes' widths and heights. In the above relationships, relativity is achieved by dividing on the height or width of the first object depending on which is more stable for practical reasons. Although it is logical to divide differences in heights using height and differences in widths using width to compensate for scale changes in the general case.

Fitness Functions

The proposed fitness is taken as the inverse of the calculated objective distance between the current chromosome and the prototype chromosome. Before looking up on how the objective distance is calculated, first we can contemplate how the geometric relationships between the objects inside a compound object are represented, followed by a discussion of parameter adaption in the case of various LP detection layouts. The Genetic algorithm stops if one of the following conditions is met.

1) The best chromosome's objective distance (OD) is less than 5.

2) If the average objective distance (AOD) is not improved for 6 successive generations. Then chromosomes having minimum objective distance can be accepted if it is less than 8. This maximum limit will affect the allowable angle range for the detected license numbers. For most License plates, the alphabets and digits have same heights while some symbols might have different widths than others. Hence, standardized relationships between any two objects can be based on the height of the first object.

In this system, the Stochastic Universal Sampling (SUS) method has been used for the selection of offspring in the new generation. In this method (SUS) each individual or people is assigned to a continuous segment of a line equal in size to its fitness as in roulette-wheel selection. Then, a number of equally spaced pointers are placed over the line depending on the percentage of individuals to be selected. In our system, individuals of ninety percent of the population size are selected that will exposed to mutation and crossover operators.

Mutation is necessary in this method because successive removal of members that not less fit in genetic iterations might alter or remove some characteristics of the genetic material forever. This can be avoided using mutation. By using mutation in the chromosomes, genetic algorithm ensures that new parts of the search space are attained to maintain the mating pool variety. We have employed two types of mutation operators' viz., substitution operator and swap operator as follows:

There are many methods to implement the crossover operator like, single point crossover, two/double point crossover, n-point crossover, three parent crossover, uniform crossover and, alternating crossover, etc. These operators are not suitable for our method since the generated children will not be valid due to repeated genes that may be produced. Even if we prevent repetition, the resultant children's fitness will be improved slowly because of the randomness of these mechanisms. Another solution to this problem is to design a crossover operator that insures of the generated offspring are enhanced. Since, in case of license plate detection problem, genetic algorithm is used to search for a sequence of objects having almost the same 'y' position and placed in an order according to their respective 'x' positions, then the problem can be gradually solved by dividing the recombined objects of the chromosomes' according to their 'y' positions into 2 groups and then sorting each group (constituting a chromosome) according to the 'x' positions.

Extraction

To replace only a portion of the population between generations. The most common strategy is to replace the less fit individuals in the previous generation probabilistically. In elitist strategy the best fit individuals of the previous generation are added to the current population. Feature extraction is a type of reduction of dimensionality that efficiently represents parts of an image as a compact feature vector. This approach is useful when image sizes are huge and a reduced feature representation is required.

Feature detection, extraction, and matching are joined to solve common problems with computer vision such as object recognition and detection, content-based image retrieval, face detection and recognition, and texture classification.

The building blocks of many computer algorithms are the local features and descriptors. Their applications include image registration, object detection, classification, tracking, and motion estimation. Using local features makes these algorithms to better handle rotation, scale changes, and occlusion.

5. CONCLUSION

A conclusion section is not required. Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. A conclusion might elaborate on the importance of the work or suggest applications and extensions.

REFERENCES

1. Y. Qiu, M. Sun, and W. Zhou, "**License plate extraction based on vertical edge detection and mathematical morphology,**" in Proc. Int. Conf. Comput. Intell. Softw. Eng., Dec. 2009, pp. 1–5.
2. A. Ahmadyfard and V. Abolghasemi, "**Detecting license plate using texture and color information,**" in Proc. Int. Symp. Telecommun., 2008, pp. 804–808.
3. G. Li, R. Yuan, Z. Yang, and X. Huang, "**A yellow license plate location method based on RGB model of color image and texture of plate,**" in Proc. 2nd Workshop Digit. Media Its Appl. Museum Heritages, 2007, pp. 42–46.
4. X. Shi, W. Zhao, Y. Shen, and O. Gervasi, "**Automatic license plate recognition system based on color image processing,**" in Lecture Notes on Computer Science, Berlin, Germany: Springer-Verlag, 2005, vol. 3483, pp. 1159–1168.
5. M. Deriche, "**GCC license plates detection and recognition using morphological filtering and neural networks,**" Int J. Comp. Sci. Info Security, vol. 8, no. 8, pp. 263–269, Dec. 2010.
6. O. Villegas, D. Balderrama, H. Dom´inguez, and V. S´anchez,"**License plate recognition using a novel**

fuzzy multilayer neural network," Int. J. Comput., vol. 3, no. 1, pp. 31–40, 2009.

7. S. H. M. Kasaei, S. M. M. Kasaei, and S. A. Monadjemi, "**A novel morphological method for detection and recognition of vehicle license plate,**" Amer. J. Appl. Sci., vol. 6, no. 12, pp. 2066–2070, 2009.
8. A. Theja, S. Jain, A. Aggarwal, and V. Kandanvli, "**License plate extraction using adaptive threshold and line grouping,**" in Proc. ICSPS, Jul. 2010, vol. 1, pp. 211–214.
9. P. Tarabek, "**Fast license plate detection based on edge density and integral edge image,**" in Proc. Int. Conf. Appl. Mach. Intell. Inform, 2012, pp. 37–40.
10. V. Abolghasemi and A. Ahmadyfard, "**A fast algorithm for license plate detection,**" in Proc. Int. Conf. Visual Inform. Syst., 2007, vol. 4781, pp. 468–477.
11. S. Roomi, M. Anitha, and R. Bhargavi, "**Accurate license plate localization,**" in Proc. Int. Conf. Comput. Commun. Electr. Technol., 2011, pp. 92–99.
12. W. Wang, Q. Jiang, X. Zhou, and W. Wan, "**Car license plate detection based on MSER,**" in Proc. Int. Conf. Consumer Electron. Commun. Netw., 2011, pp. 3973–3976.
13. H. W. Lim and Y. H. Tay, "**Detection of license plate characters in natural scene with MSER and SIFT unigram classifier,**" in Proc. IEEE Conf. Sustainable Utilization Develop. Eng. Technol., 2010, pp. 95–98.
14. H. Anoual, S. Fkihi, A. Jilbab, and D. Aboutajdine, "**Vehicle license plate detection in images,**" in Proc. Int. Conf. Multimedia Comput. Syst., 2011, pp. 1–5.
15. X. Zhang and S. Zhang, "**A robust license plate detection algorithm based on multi-features,**" in Proc. Int. Conf. Comput. Autom. Eng., 2010, vol. 5, pp. 598–602.
16. D. Zheng, Y. Zhao, and J. Wang, "**An efficient method of license plate location,**" Pattern Recogn. Lett., vol. 26, no. 15, pp. 2431–2438, 2005.
17. J. Xu, S. Li, and M. Yu, "**Car license plate extraction using color and edge information,**" Mach. Learning Cybern., vol. 6, pp. 3904–3907, Aug. 2004.
18. S. Z. Wang and H. M. Lee, "**Detection and recognition of license plate characters with different appearances,**" in Proc. Conf. Intell. Transp. Syst., 2003, vol. 2, pp. 979–984.
19. L. Carrera, M. Mora, J. Gonzalez, and F. Aravena, "**License plate detection using neural networks,**" in Proc. IWANN, 2009, vol. 2, pp. 1248–1255.
20. M. I. Chacon and A. Zimmerman, "**License plate location based on a dynamic PCNN scheme,**" in Proc. Int. Joint Conf. Neural Netw., 2003, vol. 2, pp. 1195–1200.
21. S.-L. Chang, L.-S. Chen, Y.-C. Chung, and S.-W. Chen, "**Automatic license plate recognition,**" IEEE Trans. Intell. Transp. Syst., vol. 5, no. 1, pp. 42–53, Mar. 2004.
22. S. K. Kim, D. W. Kim, and H. J. Kim, "**A recognition of vehicle license plate using a genetic algorithm based segmentation,**" in Proc. Int. Conf. Image Process., 1996, vol. 1, pp. 661–664.
23. J. Xiong, S. Du, D. Gao, and Q. Shen, "**Locating car license plate under various illumination conditions using genetic algorithm,**" in Proc. ICSP, 2004, vol. 3, pp. 2502–2505.
24. Z. Ji-yin, Z. Rui-rui, L. Min, and L. Yinin, "**License plate recognition based on genetic algorithm,**" in Proc. Int. Conf. Comput. Sci. Software Eng., Dec. 2008, vol. 1, pp. 965–968.
25. V. P. de Ara´ujo, R. D. Maia, M. F. S. V. D'Angelo, and G. N. R. D'Angelo, "**Automatic plate detection using genetic algorithm,**" in Proc. 6th WSEAS Int. Conf. Signal Speech Image Process., Sep. 2006, pp. 43–48.
26. K. I. Kim, K. Jung, J. H. Kim, S.-W. Lee, and A. Verri, "**Color texturebased object detection: An application to license plate localization,**" in Lecture Notes on Computer Science, Berlin, Germany: Springer-Verlag, 2008, vol. 2388, pp. 293–309.
27. J. Cano and J. C. Perez-Cortes, "**Vehicle license plate segmentation in natural images,**" in Lecture Notes on Comput. Sci., Berlin, Germany: Springer-Verlag, 2003, vol. 2652, pp. 142–149.
28. M. Sezgin and B. Sankur, "**Survey over image thresholding techniques and quantitative performance evaluation,**" J. Electron. Imaging, vol. 13, pp. 146–165, Jan. 2004.
29. N. Otsu, "**A threshold selection method from gray level histograms,**" IEEE Trans. Syst., Man, Cybern., vol. 9, no. 1, pp. 62–66, Jan. 1979.
30. S. E. Umbaugh, **Computer Vision and Image Processing**. Englewood Cliffs, NJ, USA: Prentice-Hall, 1998, pp. 133–138.
31. J. E. Baker, "**Reducing bias and inefficiency in the selection algorithm,**" in Proc. Int. Conf. Genet. Algorithms Their Appl., 1987, pp. 14–21.
32. Z. H. Ahmed, "**Genetic algorithm for the traveling salesman problem using sequential constructive crossover,**" Int. J. Biometrics Bioinformatics, vol. 3, no. 6, pp. 96–105, 2010.
33. W. Lenders and C. Baier, "**Genetic algorithms for variable ordering problem of binary decision diagrams,**" in Lecture Notes in Comput. Sci., New York, NY, USA: Springer-Verlag, 2005, vol. 3469, pp. 1–20.