Application independent localisation of vehicle plate number using multi-window-size binarisation and semi-hybrid genetic algorithm

Gibrael-Elamin Abo-Samra

Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia
E-mail: gabosamra@kau.edu.sa

Published in The Journal of Engineering; Received on 1st November 2017; Revised on 25th December 2017; Accepted on 9th January 2018

Abstract: This study introduces an enhanced version of a previously published genetic algorithm based technique to allow fast and accurate detection of the vehicle plate number independently of the used application. Hence, significant enhancements are introduced to upgrade the genetic algorithm into a semi-hybrid category by preceding it with a sorting and subgrouping process that reduces the search space and following it by a rule-based local search to optimize its output. The updated population structure and its initiation with subsets of the sorted image foreground objects results in a linear relationship between the image complexity represented by the number of foreground objects and the genetic algorithm search space instead of being exponential in previous versions. Moreover, a novel extra variable-size-window adaptive binarisation step is introduced to overcome the problems of attached license symbols that cannot be solved by the skipping ability introduced in previous versions. Various image samples with a wide range in scale, orientation, and symbol connectivity have been experimented to verify the effects of the new improvements. Encouraging results with 99.2% detection accuracy have been reported using a public dataset outperforming the author of the dataset by more than 5.5% and the state of the art systems by 2%.

1 Introduction

Automatic license plate recognition (ALPR) is the most important module of Intelligent Transportation Systems. ALPR is used separately to perform vehicle identification for the purpose of toll payment, parking fee payment, freeway and arterial monitoring systems, detection of stolen vehicles, among others [1–3]. ALPR applications can be split into three categories (access control (AC), traffic law enforcement (LE), and road patrol (RP)) based on the problem variables such as camera viewpoint, plate size in the image, and illumination conditions [3]. An application-oriented system has been introduced in [3]. Although most ALPR systems are performed in three steps (detection, segmentation, and recognition) [1–3], the genetic algorithm (GA)-based system, introduced initially in [4] and enhanced in [5] combines the first two steps in one step. In this paper, I am going to evolve the enhanced GA-based solution in [5] to allow detection and segmentation of a variety of LPs in different application categories without resorting to select different system parameters for each category as in [3]. In ALPR systems based on the connected component analysis technique, the accuracy and performance are significantly dependent on the binarisation stage where each pixel in the image is decided to belong either to the background or to the foreground objects. In [4], an adaptive local binarisation stage has been adopted to overcome illumination changes in addition to the skipping ability introduced in the GA to overcome physical problems affecting the CCA techniques. The concept of skipping that was introduced in [4] and enhanced in [5] supports partial detection of the LP based on the detection of a few number of license symbols less than the defined number of symbols in the LP (L). The number of skipped symbols (SN) and their positions are then determined based on the relevant geometrical relationship matrix (GRM) that defines the geometrical relationships between the symbols in the LP layout. Applying the previous GA-based system on a different dataset such as in [3], raises new concerns related to the CCA such as connected symbols that their large number results in ambiguous situations that cannot be handled by the skipping ability introduced in the previous systems [4, 5]. To overcome these problems, extra processing and enhancements are introduced in this research at different stages to get an application independent system. In addition to the enhancement introduced in the binarisation stage, the GA itself is upgraded to a higher category such as hybrid algorithms by preceding it with a sorting stage that reduces the search space and hence the time required in the global search, enhancing the effect of the mutation by limiting its search space in a narrower set, and finally a local search is carried out after completion of each GA run to optimise the skipping locations based on smartly designed logical and heuristic rules. A detailed survey of ALPR solutions based on different representations of the LP problem and using different techniques can be found in [1, 2]. In [4], the system was experimented based on a public dataset [6] while in [5], a proprietary dataset was built to prove the orientation independence using the full angle range from −85° to 85°, which was not available in any published dataset. Although the research that built the dataset in [3] has been cited by more than 50 papers (according to Google scholar at the time of writing this paper), only two of them [7, 8] are considered because they have detailed and well-defined methods and had reported results based on this dataset which has a main complexity inherited in the tiny symbols that are mostly connected to each other as will be shown in this research. Another recent paper [9] is considered because it has its implemented powerful code publicly published on the Web. Other related work may be considered in the experiments section. A main difficulty that perhaps prevented most of the researchers from using this dataset is the large scope of the LP rotation angle, which has been solved, in my previous system [5]. In this research, I am going to enhance the adaptive binarisation step to minimise the errors due to connected symbols by trying to zoom in (using a window size less than the CCA box size) through a variable-window-size binarisation step if the default one fails. If the zoom-in trial fails, a zoom-out (using a larger fixed-size window than the default one) trial is performed to correctly segment symbols that cannot be segmented in the first two trials. In addition, the system accuracy will be increased by mapping the inclined LP symbols to the new reduced-in-size GRM. The significant change that will be introduced in the GA is the use of subgrouping technique to speed up the GA but in our case, each subgroup will use a subset of the available objects, which are sorted based on a hashing function (HF) that brings LP objects near each other. This enhancement in the population architecture and its initiation and the limitation of the search space for the mutation operator upgrades the GA to the level of hybrid GAs.
especially after following the GA by a local search that optimises the skipping positions as will be shown later in the following sections. A complete overview of the enhanced system will be introduced in Section 2. The image-processing phase is summarised in Section 3 with detailed implementation of the binarisation steps. In Section 4, GA modifications are described in detail. Finally, the results are presented and discussed in Section 5 and the conclusion is given in Section 6.

2 System overview

In this research, the system will have three trials to overcome the image processing problems related to the CCA technique, in addition to the enhancement of the skipping ability introduced in the previous systems. In the first trial, the maximum skipping number MAXSN will be two or three based on \( L \) (\( \text{MAXSN} = L - 4 \)). In the second and third trials, MAXSN will be increased by one. The purpose of the second trial is to separate the attached license symbols by applying a variable-size window local binarisation technique if the first trial fails in finding a solution with an acceptable horizontal objective distance (HOD) as defined in [5]. The third trial is done if the second trial fails to find a solution with acceptable HOD, where in this case a fixed-size window having double the default size is used in the binarisation stage. The separation of trials into three is intentionally done to determine the impact of each trial on the accuracy of the system and to search for different solutions for complex LPs. The complete system flow chart is shown in Fig. 1. The system is composed of three phases and each phase is composed of many stages as in [5]. Most of these phases and their stages will be enhanced and supported with new additions to get higher accuracy and performance as described in detail in the following sections.

3 Image processing phase

In this phase, an input colour image is exposed to a sequence of four stages to extract a number \( N \) of the relevant two-dimensional objects that may represent the symbols constituting the LP. These four stages [colour-to-grey conversion, adaptive binarisation, morphological operations, and connected component analysis (CCA: MATLAB version is used although there are many efficient implementations [10])] are the same in the first trial as in [5]. The change is done in the adaptive binarisation stage that will include two extra versions that one of them is executed in the second trial and the other in the third trial when the system fails to localise the LP in a former trial. The third trial is a fixed-size window adaptive binarisation as the first one but with double the default size while the second stage is a variable-size-window local binarisation stage that will be demonstrated in the following section. The default window size value (DWSZ) is selected as half of the image width divided by the number of LP symbols \( L \), assuming that the LP occupies \( \sim 60\% \) of the image width

\[
\text{DWSZ} = 0.5 \times \text{ImageWidth} / L
\]  

(1)

3.1 Multi-window-size local adaptive stage

In the first trial, the system performs binarisation using a fixed-size window. The binarisation is based on a local threshold value (LT) which is evaluated for each pixel \((x_0, y_0)\) based on the mean of the intensity value in the square surrounding the pixel and having a side length WS (=DWSZ) as follows:

\[
\text{LT}(x_0, y_0) = \text{BC} \times \left( \frac{\sum_{x=0}^{x=\text{WS}/2} \sum_{y=0}^{y=\text{WS}/2} \text{IM}(x, y)}{\text{WS}^2} \right)
\]

where \( \text{BC} \) is a binarisation constant = 0.9

(2)

The above equation has been efficiently computed by the integral images [11]. WS is selected according to (1) which is sufficient for normal conditions. In some cases when the distances between the license symbols are very small, especially when the capturing device resolution is low or the camera-to-car distance is long (which depends on the application), the resultant binarised symbols are attached to each other in the LP image. This attachment problem is solved partially by the previous systems [4, 5] if the number of attached symbols is small relative to the total number of the license symbols \( L \) by the skipping technique. In case of seven license symbols as for the Saudi and Greece LPs [4], a maximum of three symbols can be skipped and their locations can be estimated based on the prototype GRM [5]. If the number of attached symbols is more than 3, the GA may fail in the localisation process and even if partial detection happens, the expected locations of the skipped symbols will be ambiguous. For higher accuracy, the skipping in the first trial is limited to two skipping symbols \( L = 6 \) in [3] and if the GA fails, the localisation process is restarted by the new local adaptive stage. The new local adaptive stage will use a variable-size window, unlike the first trial which uses a default fixed-size window with side length (DWSZ). The variable window size VWS at any pixel \((x, y)\) will be calculated based on the size of the CCA object (detected in the first trial) that contains this pixel. My concept is based on the assumption that: if the attached symbols constitute a rectangle whose shorter

Fig. 1  System’s overall flowchart for the localisation of an L-symbols LP where yellow or shaded blocks indicate enhancement locations

J. Eng., 2018, Vol. 2018, Iss. 2, pp. 104–116  doi:10.1049/joe.2017.0815  This is an open access article published by the IET under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0/)
length represents the character height \((H)\), then a window which has approximately the character width \((i.e. \text{VWS}=H)\) multiplied by the default aspect ratio \(\text{ASPR}\) will do better segmentation of the current object. To get a better understanding of the best window size for the binarisation problem the reader is referred to \cite{12} where document binarisation is considered. Although \cite{12} has been cited by more than 30 papers (as reported by Google Scholar at the time of writing this paper), only two of them \cite{13,14} have considered the license plate problem. In \cite{13}, a different technique has been proposed to correctly segment symbols inside an already detected LP. Also, \cite{14} suggested an iterative loop method to segment the LP after detection. Both methods used prior knowledge of the detected LP characteristics before doing the segmentation process. None of them introduced an enhancement of the technique proposed in \cite{12}. The difference in my implementation compared to \cite{12} lies in the application of the multi-window-size concept (without using the variance) in a scenario of three trials instead of one for experimentation issues. In addition, a simplification of the calculation of the zoom-in window size, which is based on the character thickness and height in \cite{12}, is done by considering it as the character width in the proposed solution. Moreover, if the rotation is considered, the solution in \cite{12} should be modified because it was proposed for almost horizontal text lines in normal documents. The rotation independency is handled in the proposed solution by selecting the minimum of the object dimensions \((H \text{ and } W)\) and then multiply it by the aspect ratio and the bounding box extension \((\text{BExt})\) parameter which represents the percentage of the foreground pixels in the rectangle surrounding the character (or block of connected characters \(\text{BCC}\)). To justify the proposed approximation, the general case is considered when a \(\text{BCC}\) is detected in the first trial with dimensions \(h \times w\) and rotated by an angle \(\Theta\) as shown in Fig. 2a. The detected height \((H)\) and width \((W)\) of the surrounding rectangle, in conjunction with the \(\text{BExt}\) are already given after performing the first CCA step.

The dimensions \(h\), \(w\), and the angle \(\Theta\) are unknown but can be deduced based on the following equations:

\[
H = h \times \cos(\Theta) + w \times \sin(\Theta) \tag{3}
\]

\[
W = h \times \sin(\Theta) + w \times \cos(\Theta) \tag{4}
\]

\[
\text{BExt} = (h \times w) / (H \times W) \tag{5}
\]

Solving these three non-linear equations will lead to the intermediate solution of \(h\) in terms of \(\Theta\) as follows:

\[
h = \text{BExt} \times H \times \text{FNC}(\Theta) \tag{6}
\]

where \(\text{FNC}(\Theta)\) is given by the following formula:

\[
\text{FNC}(\Theta) = (\cos^2(\Theta) - \sin^2(\Theta)) / (\cos(\Theta) - (H/W)\sin(\Theta)) \tag{7}
\]

The changes of \(\text{FNC}(\Theta)\) versus \(\Theta\) using different rectangle lengths (two to six solid rectangular characters) are shown in Fig. 2b.

Assuming \(\text{FNC}(\Theta)\) is equal to 1 at any angle \(\Theta\), the average width of any symbol inside the detected block will be

\[
w = \text{BExt} \times H \times \text{ASPR} \tag{8}
\]

If the \(\text{BCC}\) is completely solid, \(\text{(8)}\) will be correct but in real LPs, the \(\text{BCC}\) will have its own real extension \(\text{RBExt}\) which will linearly affect the measured extension \((\text{MExt}=\text{BExt} \times \text{RBExt})\). \(\text{RBExt}\) depends on the symbols extensions inside the \(\text{BCC}\) and the space

---

**Fig. 2** Calculation of the character width of a block of connected characters based on the algebraic solution and the approximated function

\(a\) Solid rectangle \((h \times w)\) rotated by \(\Theta\) inside its surrounding rectangle \((H \times W)\)

\(b\) Variation of \(\text{FNC}\) versus \(\Theta\) for different rectangle lengths (in terms of the number of connected characters)

\(c\) Detected character width versus the block angle based on the algebraic solution

\(d\) Detected character width versus the block angle based on the approximated expression

This is an open access article published by the IET under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0/)

J. Eng., 2018, Vol. 2018, Iss. 2, pp. 104–116

doi: 10.1049/joe.2017.0815
between symbols. Hence, the approximate width can be given in general as follows:

\[ w = M_{\text{Ext}} \times H \times \text{ASPR} / \text{RB}_{\text{Ext}} \]  

(9)

Based on the experimented dataset [3] that has symbols with \( \text{ASPR} \approx 0.5 \) and assuming \( \text{RB}_{\text{Ext}} \) to be 0.45 for a BCC (this value is higher than the normal extension because we know that the symbols are connected inside the BCC), (9) can be approximated to

\[ w = 1.1 M_{\text{Ext}} \times H \]  

(10)

The incorporation of \( M_{\text{Ext}} \) in (10) replaces the time-consuming method used in the calculation of the character thickness in [12] and at the same time approximates the width estimation at different angles that supports the independency on the concerned application. Figs. 2c and d illustrate the quality of the proposed approximation by displaying the detected widths for different character block widths versus the block angle based on the algebraic solution of (2)–(4) (Fig. 2c) and based on the proposed approximation (Fig. 2d).

The inexact results of the algebraic solution are due to the approximated value of \( \text{RB}_{\text{Ext}} \) that is needed to reach an exact solution. Since \( \text{RB}_{\text{Ext}} \) is not constant for different combinations of symbols and its measured value is affected by the previous inexact binarisation stage, then it is expected for the algebraic solution to suffer from non-perfect results as the approximated one. Moreover, if we consider the execution time we will be satisfied by the proposed approximation. The implementation algorithm will be based on the output of the first trial and will be explained as follows. The inputs to the proposed binarisation algorithm are the previous label matrix (\( LBM \)) output from the previous CCA stage (in trial 1), 3 arrays: \( WA, HA, \) and \( MEXTA \) containing \( W, H, \) and \( M_{\text{Ext}} \) of every CC object, respectively, the input grey image \( GIMG \), the previous integral image of the input grey image \( \text{IntGIMG} \). Based on these inputs, the binarisation algorithm will be as shown in (Fig. 3).

Based on the system flowchart in Fig. 1 and the work done in [5], we can show the intermediate and final outputs of an LP image as in Fig. 4. The output sequence starts in trial 1 at the fixed size window binarisation stage (Fig. 4a) and then passes through the objects filtering phase (Fig. 4b) and then designated as unacceptable LP with high HOD in the three runs of the GA with three different skipping values \( SN = 0,1,2 \) as in (Figs. 4c–e) respectively. Since trial 1 failed, trial 2 continues the work by the variable-size binarisation stage that separates the two previous BCCs (119 and 160) into correctly segmented symbols (Fig. 4f), performs size filtering (Fig. 4g), and finally detects the LP symbols by the GA (Fig. 4h).

Since the variable size binarisation converts only foreground pixels into background ones, it has the effect of segmenting larger objects into smaller ones or clearing small objects as in Fig. 4f. Fig. 5 displays 16 original LP images (columns 1, 4, 7, and 10) and their binary output using the default fixed-size window (columns 2, 5, 8, and 11) in trial 1 and the binary output using the new variable-size-window method (columns 3, 6, 9,
the following one gave excellent results:

from values for non-LP objects. Many HFs have been tried where
they give LP symbols hashing values near each other while being far
from each other subgroups. A HF can be proposed to
affect other subgroups.

fast convergence, subgrouping supports convergence to global
number of iterations to reach the optimal solution. In addition to
result in a better performance because the GA will require smaller
where each subgroup gets a subset of the
object

Due to the large scope of both scale and angle of our GA-based
system, partitioning the whole population into G subgroups
where each subgroup gets a subset of the M objects (output from
the objects filtering phase) based on their X or Y coordinates will
result in a better performance because the GA will require smaller
number of iterations to reach the optimal solution. In addition to
fast convergence, subgrouping supports convergence to global
optimum because a local minimum in one subgroup will not
affect other subgroups.

In case of slightly inclined plates, distributing objects inside sub-
groups based on the Y-coordinates of the objects will enhance the
GA performance. In case of fixed size symbols (which is true for
most of the recent LPs in the world), a HF can be proposed to
give LP symbols hashing values near each other while being far
from values for non-LP objects. Many HFs have been tried where
the following one gave excellent results:

| HF(obj) = (ImageWidth + ImageHeight) |
| | × Wj × Hj + Xj/Wj + Yj/Hj |

Applying this many-to-one HF clusters objects based on their areas,
which brings LP objects near each other and orders them based on
their normalised coordinates, which simplifies the job of the GA
and results in getting global optimum in almost all cases and
hence increases the system speed and accuracy. After experimenta-
tion for almost all of the plates in the dataset used in [3], we found
that the LP objects would be found in a range (last LP symbol index
– first LP symbol index + 1) of at most 3L (=18) objects in the sorted
sequence of M objects based on the HF function even for large M
(≥320) as shown in Fig. 6.

To prove the significance of this result, we will roughly compare
the search space size (SSZ) in each case. In case of a single group
having M objects and searching for a sequence of L objects that
gives the minimum objective distance, the SSZ is M/(M−L)!,
that can be approximated in case of large M into (M−L)/αM. By par-
titioning the population (Z) into G subgroups where G equals
(M/L−3) and limiting each subgroup to only 4L objects (three
objects are shared between every two successive subgroups) as in
Fig. 7, the SSZ for each subgroup will be 4L!/3L!, that can be
roughly approximated to (4L−L/2)² or (3.5L²²). Having approxi-
mately M/L subgroups, we can deduce that the total SSZ in case of
subgrouping is M/L × (3.5L²² or simply M × (3.5L²²−1)² which is a
linear function of M in comparison to the exponential function in
case of a single group. Hence, we can deduce that the required de-
tection time of a blind search algorithm will increase linearly with
the number of objects found in the image instead of being expo-
ential in case of a single group which is a great improvement although
it may not be observable in case of small M (≤4L).

Another advantage of the introduced subgrouping is that each
subgroup has its own optimal solution which can be used to
support detection of multiple plates per image under the condition
of preventing interlaced solutions. Unfortunately, the number of
multi-plate images in the dataset under consideration is very
small and cannot be used to evaluate this feature because many
experiments should be done to assess the performance of this

---

**Fig. 5** 16 original LP images (columns 1, 4, 7, and 10) and their binary output after using trial 1 (columns 2, 5, 8, and 11) and trial 2 (columns 3, 6, 9, and 12).

---

**Fig. 6** LP symbol objects range lengths versus the number of objects in dif-
ferent LP images in case of sorting based on the object’s X-position (X) and
in case of sorting based on the HF.
4.2 Compact multi-dimensional GRM layout

In [5], the GRM was upgraded to support highly inclined plates by the inclusion of the geometrical relationship values of the concerned prototype LP in the full range of the angles $\Theta$ $[-90^\circ, 90^\circ]$. Since the relationship values are concerned with the differences between the coordinates and dimensions of the consecutive (and non-consecutive in case of skipping) symbols, then by noticing that the absolute values of these differences are repeated four times, each in one of the four $45^\circ$ sectors, we reach to the conclusion that through mapping, the $180^\circ$ directions can be represented by only one of the four $45^\circ$ sectors, we reach to the conclusion that through mapping, the $180^\circ$ directions can be represented by only one of the $45^\circ$ sectors, reducing the GRM size to one-fourth of its original size. According to Figs. 8(a–d), in which symbols are represented as rectangles with fixed dimensions, we have

$$\Delta X_a(=X_{a2} - X_{a1}) = \Delta Y_b(=Y_{b2} - Y_{b1}), |\Delta X_a| = |\Delta X_c|,$$

$$|\Delta Y_b| = |\Delta Y_d|, \Delta Y_a(=Y_{a2} - Y_{a1}) = \Delta X_b(=X_{b2} - X_{b1}),$$

$$|\Delta Y_a| = |\Delta Y_c|, |\Delta X_b| = |\Delta X_d|,$$

$$H_a = W_b = H_c = W_d \text{ and } W_a = H_b = W_c = H_d$$

\[(14)\]

Then the mapping will be as follows:

$$\begin{align*}
\Theta_m &= \Theta & 0 \leq \Theta \leq 45^\circ \text{ sector a} \\
\Theta_m &= 90^\circ - \Theta & 45^\circ < \Theta < 90^\circ \text{ sector b} \\
\Theta_m &= -\Theta & -45^\circ < \Theta < 0 \text{ sector c} \\
\Theta_m &= 90^\circ + \Theta & -90^\circ < \Theta < -45^\circ \text{ sector d}
\end{align*}$$

\[(15)\]

After mapping, the GRM values can be calculated by substituting $\Theta$ with $\Theta_m$ in the new compact GRM as follows:

$$\text{GRM}(\Theta_m, R, i_1, i_2)$$

\[(16)\]

where $\Theta_m$ represents the unknown mapped sequence direction angle, $R$ represents the relation type ($1$: RX, $2$: RY, $3$: RH, $4$: RW), $i_1$ represents the first gene index, and finally, $i_2$ represents the second gene index where $i_2 > i_1$.

Before evaluation of the objective distances using the mapped angle in the GRM, the relationships in the four sectors should be redefined based on the geometric rules that appear from Figs. 8 (a–d) as follows.

Sector a: all relationships are defined as in [4] as follows:

$$RX_{1j2} = (X_{j2} - X_{j1})/H_{j1} \quad (17)$$

$$RY_{1j2} = (Y_{j2} - Y_{j1})/H_{j1} \quad (18)$$

$$RH_{1j2} = (H_{j2} - H_{j1})/H_{j1} \quad (19)$$

Sector b: the relationships will be redefined as follows:

$$RX_{1j2} = (Y_{j2} - Y_{j1})/W_{j1} \quad (20)$$

$$RY_{1j2} = (X_{j2} - X_{j1})/W_{j1} \quad (21)$$

$$RH_{1j2} = (W_{j2} - W_{j1})/W_{j1} \quad (22)$$

The relationships for sectors c and d can be similarly deduced as for sector b.

In [5], it was proved that the variation in the measured symbol dimensions is increased with the increase of the inclined angle $\Theta$, hence by performing this mapping, not only the GRM size will be reduced, but also more accurate objective distances will be gained since calculations will be performed in a limited and more stable range which will increase the rigidity of the geometric features and hence maximises the true positive rate. On the other hand, an important enhancement is gained by this modification, if we consider the maximum acceptable objective distance threshold, which was defined in [5] as follows:

$$OD_{\text{max}} = 4 + 0.1 |\Theta_k|$$

\[(23)\]
Where in [5], $\Theta_d$ was changing from 0° to 85° (~90) but now it will be substituted by $\Theta_m$ which changes from 0 to 45°. Hence, OD$_{max}$ will be less in the two sectors b and d, which should minimise the false positives also. For example, if a chromosome (k) has $\Theta_d$ equals 85°, then $\Theta_m$ will be 5° and OD$_{max}$ will be 4.5 instead of 12.5 which surely will reduce the false positives significantly and increase the accuracy of the detected plates at this angle direction.

### 4.3 Objective distance and fitness modifications

The objective distance is calculated as defined in [5], but based on the experiments on the new public dataset [3], some refinements appear to be necessary to increase the system accuracy. Unlike the previous systems where the concentration was on the general architecture of the system, in this paper a deeper solution is introduced by exploring systems where the concentration was on the general architecture of the concerned LP as stated in [5].

Hence, the three relationships (17) will be redefined as follows:

\[
RX_{j1,2} = 2(X_{2} - X_{j1})/(H_{j1} + H_{j2})
\]

\[
RY_{j1,2} = 2(Y_{2} - Y_{j1})/(H_{j1} + H_{j2})
\]

\[
RH_{j1,2} = 2(H_{j2} - H_{j1})/(H_{j1} + H_{j2})
\]

This modification should be taken into consideration in the mapping defined in the previous section when performing calculations in the other three sectors.

Another modification regards the weighting parameters, which should be chosen based on the stability of the geometric features of the concerned LP as stated in [4]. In case of the public dataset in [3], the $X$ and $Y$ distances between the centres of the symbols are more fixed than the distances between the coordinates of the corners of these symbols. Hence, by using the centre coordinates instead of the corner coordinates, the $X$ and $Y$ coordinate based distances will be nearly stable as the height based distances in case of semi-horizontal plates. Consequently, the same weighting value ‘3’ can be used for all the three distances in the objective function that defines the distance between a prototype $p$ and any chromosome $k$ as follows:

\[
OD_{k,p} = 3\Delta RX_{k,p} + 3\Delta RY_{k,p} + 3\Delta RH_{k,p}
\]

where $\Delta RX_{k,p}$, $\Delta RY_{k,p}$, and $\Delta RH_{k,p}$ are calculated as in [5].

Hence, the fitness of a chromosome $k$ (Fit$_k$) will be given as follows:

\[
Fit_{k} = -3\Delta RX_{k,p} - 3\Delta RY_{k,p} - 3\Delta RH_{k,p}
\]

### 4.4 Mutation and crossover operators

In [4, 5], two types of interchangeably used mutation operators have been implemented: substitution operator and swap operator. Unlike what is done in [4, 5] and almost all GA based systems mutation will be performed before crossover to prevent disordering of the unevaluated offspring produced by the crossover and, at the same time, allow the crossover to benefit from the done changes by the mutation operator. After experimenting the benefits of the mutation operators and finding that, the work done (objects ordering) by the swap operator is performed more accurately by the sorting crossover a decision was made to use only the substitution operator. The crossover operators are left intact as in [5].

#### 4.4.1 Substitution operator:

In this type of operators, a random position in the chromosome is selected and a new random object from the $M$ available objects replaces the corresponding allele if the random position is in the non-skipped region. Due to the mentioned subgrouping and the sorting of objects based on HF defined in (13), the range of objects from which the random object is selected can now be limited to what we can call the near set. The near set, used to replace an object ($j$), has the range $[j-L, j+L]$, which significantly enhances the effectiveness of the mutation operator in comparison to the previous systems which uses the full range $[1; M]$. If the random position is located in the skipped region, a new random number from one to $L$ substitutes the current value at this location as in Fig. 9.

#### 4.5 Fake detection and skipping optimisation stage

In the previous version [5], a fake detection stage was introduced to detect false positive outputs from the GA by mapping the resultant sequence to the horizontal direction where the rigidity of the plate becomes maximum (or the variance of the different symbol dimensions becomes minimum) and the objective distance represents the actual difference between the output and the horizontal prototype template. In this paper, another enhancement will be done to optimise the output in case of skipping, where the positions of the skipped symbols can have different arrangements that are all accepted from the perspective of having a final objective distance less than OD$_{max}$. The lowness of the OD than OD$_{max}$ criteria may result in accepting many cases with incorrect positions of the skipped symbols because the width relationship is not included in the objective distance due to the variation of the symbol widths of most LPs. To correct this, a type of exhaustive search is performed by testing all combinations of possible skipping positions in the horizontal version [instead of doing it in every chromosome in the population (Z) for speed issues] to reach to the pattern having the minimum objective distance taking into consideration the following heuristic rules:

(a) Select the most in-phase sequence where an out-of-phase penalty is measured as follows:

\[
\text{(i) Count the number of positive values (XPOS) for the } r \text{th } x\text{-distance for chromosome } k \text{ which is defined as follows:}
\]

\[
\left(\Delta RX_{k,p}\right)^+ = \left(RX_{k+1,p} - RX_{k+1,p}\right)
\]

when $i$ goes from 1 to $L - SN - 1$ and $|\Delta RX_{k,p}| > 0.1$

(ii) Count the number of negative values (XNEG) when $|\Delta RX_{k,p}|$ is less than $-0.1$

(iii) Multiply the current HOD by the out-of-phase penalty $(1 + |\text{XPOS + XNEG})/(1 + |\text{XPOS - XNEG})$ to get the synchroised HOD (SHOD).

Fig. 9 Example of the substitution operator in case of skipping where the dashed line separates the non-skipped and skipped parts and the shaded area represents part of the near set of a seven-symbol LP

---

This is an open access article published by the IET under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0/).
The above optimisation is carried out because the definition of the objective distance is concerned with absolute values and neglects the signs of the distances of the consecutive symbols w.r.t. the corresponding prototype template symbols represented in the GRM. The importance of the signs of distances appears mostly in the RX relationship especially when there are skipped symbols or two-style matching problem (two prototype templates with differences in the RX relationships only). Examples of this horizontal synchronisation are shown in Figs. 10a–d where dotted arrows point to skipping positions based on the normal HOD and solid ones point to new positions based on the SHOD. The shown examples include a two-style matching problem where the best matches were for style 1 in cases (a) and (b), and style 2 in cases (c) and (d).

(b) Support skipping in positions that follow wide symbols (connected symbols) where a symbol is considered wide if its aspect ratio is more than 0.7(=0.2 + ASPR(=0.5)) for horizontal plates or (0.7 + tan(Θα))/1 + 0.7tan(Θα) for inclined plates. This will support splitting connected or attached symbols while at the same time of suggesting these positions as good skipping positions, the widths of the wide symbols are reduced to their normal values. The support is achieved by introducing a negative penalty of 5 to reduce the final objective distance when skipping is matched with each suggested position. This rule allows splitting the remaining attached symbols (counted as ATS), that may be due to physical reasons other than illumination conditions, at the end of each trial level in the output sequence of the GA phase as shown in Figs. 10e–h.

(c) Prevent skipping positions that do not have enough space left between the neighbours of the skipped symbol. When the skipping position faces this condition, a penalty of 5 is added to a skipping error accumulator parameter PrvSk that is added to the final objective distance.

Hence, the optimised horizontal distance OHOD will be given by the following formula:

\[ \text{OHOD} = \text{SHOD} - 5 \times \text{ATS} + \text{PrvSk} \]  

OHOD is calculated at every possible skipping pattern (maximum \( L!/(L-3)! \times 3! = 20 \) for \( L = 6 \) and \( SN = 3 \)) and the pattern that leads to the minimum is selected and its HOD is recalculated and checked as in [5] to decide stopping or going to the next run or trial.

With the above mentioned local optimisation, the subgrouping fed by the HF-based sorted objects and the near-set based mutation operator we do a kind of hybridisation although not traditional as defined in the literature [15], it has the same effect of reducing the search time and supporting optimal solution. Fig. 11 displays the number of generations (NumGen) needed to reach the optimal solution versus the number of objects (M) in case of a single group (circle-markers curve) and in case of subgrouping (diamond-marker curve). Since the population size (Z) increases with the number of objects (M) as defined in [5], it is better to get an indication of the processing overhead, independently of the platform and the implementation, in each case by plotting (Z × NumGen) versus \( M \). Hence, from the (Z × NumGen) curves (double lines), we can deduce that there is a great enhancement in the execution time in case of skipping will also be reduced significantly because the maximum number of generations in the stopping conditions will be 10 instead of 20 in the previous systems [4, 5].

5 Experimental results and discussion

The evaluation of this semi-hybrid GA (SHGA) based system will be based on the benchmark dataset in [3]. This dataset has three categories of images (AC, LE, RP) based on the application in which these images were collected. The binarisation parameters have been...
defined for the three categories as follows (the mentioned constants are used mainly in the object filtering stage in addition to the ASPR which is used in different stages): ASPR = \text{width/height} = 0.5, minimum symbol width (MNW) = 2 pixels, minimum symbol height (MNH) = 11 pixels, maximum symbol width (MXW) = ImageWidth/L, maximum symbol height = MXW/ASPR, minimum number of symbol foreground pixels (MNSFGP) = 6.

Although the dataset [3] has all LPs with six symbols, it has two styles. These two styles are detected in the GA using a single default style template due to the small difference between the two styles that happens only in the arrangement of the X-coordinates of the symbols inside the two prototype sequences. The effect of this simplification causes a problem when there are skipped symbols where one style will not be enough to detect the correct positions of the skipped symbols for the other style. To correct this, two GRMs (one for each style) are taken into consideration in the fake detection and skipping optimisation stage by performing the exhaustive search two times; one for each style, which corrects cases having the non-default style (see Figs. 10a–d).

Since the output of the enhanced system is in the form of segmented license plate symbols, the scores are defined in the two steps (detection and segmentation) in a fair way to be well compared with the results in [3]. First, as in [7], a plate is considered detected if the intersection of the detected area and the ground truth area divided by the union of both areas is larger or equal to 0.5. Since in [3] each LP has six symbols, then if four or more symbols are detected out of six, the detection is considered a true positive otherwise it is counted as a false positive. We will perform two experiments, Exp1 and Exp2. Exp1 will be performed based on the described scenario for the three trials. On the other hand, in Exp2, the scenario will be changed by performing the three binarisation steps in one trial and combining their labelled objects into one array after eliminating repeated objects. Also, the role of each trail in Exp2 is changed by using a different binarisation. Hence, in this study, the combined accuracy rates, leading to the conclusion that SHGA outperforms the system in [7] by more than 6% in terms of false negatives. This is an open access article published by the IET under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0/).

Table 1 Number and percentage of detected samples at each trial for different categories

| Category | NOS | Tri1 | Tri2 | Tri3 | FPs | FNs |
|----------|-----|------|------|------|-----|-----|
| AC       | 681 | 574/662 | 78/7 | 148 | 6/3 | 9/1 |
| LE       | 757 | 687/745 | 43/2 | 10/6 | 11/2 | 6/2 |
| RP       | 611 | 464/591 | 116/9 | 8/3 | 12/6 | 11/2 |
| ALL      | 2049 | 1725/1998 | 237/18 | 32/17 | 29/11 | 26/5 |
| percentage | 100% | 84.2%/97.51% | 11.5%/0.87% | 1.6%/0.83% | 1.4%/0.54% | 1.3%/0.25% |

Table 2 Number and percentage of detected samples at each skipping level for different categories

| Category | NOS | SN(0) | SN(1) | SN(2) | SN(3) |
|----------|-----|-------|-------|-------|-------|
| AC       | 681 | 318/414 | 212/172 | 123/66 | 13/25 |
| LE       | 757 | 516/382 | 136/110 | 75/40 | 13/21 |
| RP       | 611 | 234/341 | 207/156 | 132/61 | 15/45 |
| ALL      | 2049 | 1068/1337 | 555/438 | 330/167 | 41/91 |
| percentage | 100% | 52.1%/65.25 | 27.1%/21.37 | 16.1%/8.15 | 2%/4.44 |
The only change is the enhancement of the role of trials 2 and 3 in the development and execution phases. An important shortcoming
which requires an average of 0.12 s (for 640 × 480 resolution) and
0.06 s (for 320 × 240 resolution) with platform as in [3].

Another enhancement has been gained in this test which regards narrow symbols such as 1 and 1, which enforce the system to
process small non-LP objects, by giving an expression to estimate
the values of the minimum symbol width (MNW) and its related
MNH as follows:

\[ w = \text{MIN}(1.1 \times \text{ME} \times H, \text{ASPR} \times H) \] (32)

The previous estimates have the effect of filtering non-LP
symbols and speeding up the GA iterations. The other modification
related to narrow symbols regards their hashing values based on
(13) which places them far away from their related neighbours
having normal dimensions. The HF is modified to correct this situ-
ation and place thin symbols in their related subgroups with their
related symbols as follows:

\[ \text{zero} \] (see (34))

In case of LPs having unequal symbol heights such as that of the
Saudi LP where the first four digits have heights approximately
double of the remaining three letters, we can replace the heights
and/or widths in the above equation with their base-2 logarithmic
values. The system with the mentioned enhancements has been
tested again on the dataset in [3] giving same accuracies as reported
earlier, in addition to the improvements in the results when tested on
the dataset in [6] in comparison with [4] as shown in Table 5.

From Table 5, we notice that there is a 1% enhancement added by
SHGA system because the dataset in [6] is not suffering from illu-
mination conditions that result in symbols being connected to each
other. Nevertheless, we got better results with a dataset that is cate-
gorised based on the image complexity and not based on real oper-
ating applications. Considering the system in [9] which has
 overcome the speed problems in [7] and based his system on a pro-
prietary deep CNN, we will base our comparison on a sample of test
images from [3] with extra modified samples from the same dataset

\[
\text{HF}(\text{obj}) = \begin{cases} 
(\text{ImageWidth} + \text{ImageHeight}) \times W_j \times H_j + X_j/W_j + Y_j/H_j & \text{for } W_j/H_j \geq 0.6 \times \text{ASPR} \\
(\text{ImageWidth} + \text{ImageHeight}) \times H_j \times H_j \times \text{ASPR} + X_j/(H_j \times \text{ASPR}) + Y_j/H_j & \text{for } W_j/H_j < 0.6 \times \text{ASPR}
\end{cases}
\] (34)
### Table 5  Comparison of detection results between [4] and SHGA (Exp2)

| Category          | Day (large sample) | Day (close view) | Day (shadow in plate) | Difficult (shadows) | All |
|-------------------|--------------------|------------------|-----------------------|---------------------|-----|
| NOS               | 136 (100%)         | 122              | 52                    | 26                  | 336 |
| Det. [4]          | 136 (100%)         | 119 (97.5%) [3FN]| 49 (94.2%) [2FN + 1FP]| 24 (92.3%) [1FN + 1FP]| 328 (97.61%) [6FN + 2FP]|
| SHGA              | 136 (100%)         | 120 (98.4%) [2FN]| 49 (94.2%) [2FN + 1FP]| 26 (100%)           | 331 (98.5%) [4FN + 1FP]|

Fig. 12  Detected samples in different categories at different angles with trial number and SN under each
[3]. After running the published code on the prepared samples, we conclude the following: (i) the system is rotation invariant but with limits ($\Theta_k$<45) in comparison to SHGA ($\Theta_k$<90), (ii) the system is scale invariant but to some extent, while SHGA is completely scale invariant, and finally (iii) it may be using the vehicle data (logo position, car boundary, and headlights positions) to prevent FPs because it gives no information when these parts are occluded. Fig. 12 shows a variety of images detected in different categories at different angles and having different illumination conditions. Under each image four attributes are given, the category (AC, LE, or RP), the trial number, the SN, and finally the LP rotation angle. All the shown images are detected by SHGA. The first six rows are detected by Masood et al.[9] while the last two rows contain images that are not detected by Masood et al. [9] except image H3 which has only four symbols detected and images H4 and H5 which have their apparent symbols detected without indication of hidden or missing ones as indicated by SHGA. I think there are many machine learning techniques that may claim high accuracy on the same dataset by doing tricky ideas as using part of the dataset to train the algorithm and do testing on all the dataset or the remaining parts forgetting the fact that there are many LPs common between the different segments of the dataset [3]. Others use the characteristic of the LP shape or colour to filter out non-LP objects which is not our case where only the geometrical layout of the symbols is considered. As a final demonstration of the capability of SHGA, we present in Fig. 13 three images with multi-plates at different scales and orientations that are manually constructed in addition to two original images from [3] to prove the ability mentioned in Section 4.1. Future work may consider more covering and evaluation of the accuracy and performance of the system when faced with multi-plates and even multi-style LPs. Also, CNN can be used to filter out non-text objects to completely eliminate FPs. Experimentally, SHGA proved more ability than previous versions in detecting plates at a variety of conditions that makes it more independent on the type of the application.

6 Conclusion
An enhancement of the previously designed genetic-based prototype system for localising vehicle plate number inside plane images has been introduced to support fast and accurate detection of LPs having many connected symbols. Hence, a deeper solution of the CCA problems has been introduced at the low level of pre-processing in the binarisation step and at the higher level of post-processing in the fake detection and skipping optimisation stage after the GA module. In the binarisation stage, a novel variable window size based local adaptive step has been introduced to overcome the problem of connected symbols that cannot be solved by the limited skipping ability introduced in the previous systems. At the fake and optimisation stage, optimum positions of the skipped symbols have been guaranteed by the exhaustive search for the optimum objective distance under all combinations of skipped positions and based on three guiding heuristic rules using two GRMs. Moreover, a more compact representation of the GRM is introduced by minimising its size to one-fourth its original size in [5], which at the same time minimises the maximum objective distance in the two sectors having the ranges $[46°, 90°]$ and $[-46°, -90°]$, respectively. A highly significant enhancement has also been introduced by minimising the search space for the GA to guarantee fast and more accurate convergence by introducing the new initial population stage with subgrouping where each subgroup is initialised with a subset of the sorted CCA objects based on a HF that groups LP symbols near each other. The significance of this enhancement has been proven by showing that the search space will increase linearly with the increase of the image complexity represented by the total number of CCA objects instead of growing exponentially in case of a single group. In addition, allowing multiple plates to be detected per image given its symbols are placed in a separate subgroup. Moreover, the effectiveness of the mutation operator has been increased by limiting its mutation space to a narrower set. The introduced enhancements upgrade the original GA to a faster and more stable category as hybrid GAs. An important novelty is provided by maintaining scale and rotation independency in all the introduced enhancements. Finally, the enhancements have been verified by applying the new system on a benchmark dataset, which resulted in a high detection and segmentation accuracy with low FP rate. The accuracy of the new system now competes with the state-of-the-art techniques although maintaining the wide range of angle and scale with independency on the shape or the colour of the LP. The wide range of the different categories of the LP images in the experimented dataset proves the independency of the enhanced system on the intended application.

7 References

[1] Du S., Ibrahim M., Shehata M., et al.: ‘Automatic license plate recognition (ALPR): a state-of-the-art-review’, IEEE Trans. Circuits Syst. Video Technol., 2013, 23, pp. 311–325. doi: 10.1109/TCSTD.2012.2203741

[2] Gilly D., Raimond K.: ‘A survey on license plate recognition systems’, Int. J. Comput. Appl., 2013, 61, (6), pp. 34–40

[3] Hsu G.S., Chen J.C., Chung Y.Z.: ‘Application-oriented license plate recognition’, IEEE Trans. Veh. Technol., 2013, 62, (2), pp. 552–561

[4] Abo Samra G., Khalefah F.: ‘Localisation of license plate number using dynamic image processing techniques and genetic algorithms’, IEEE Trans. Evol. Comput., 2014, 18, pp. 244–257

[5] Abo Samra G.: ‘Genetic algorithms based orientation and scale invariant localization of vehicle plate number’, IJSER, 2016, 7, (4), pp. 817–829

[6] Available at http://www.medialab.ntua.gr/research/LPRdatabase, accessed December 2015

[7] Li H., Shen C.: ‘Reading Car License Plates Using Deep Convolutional Neural Networks and LSTMs’, arXiv preprint arXiv: 1601.05610 (2016)

[8] Soora N.R., Deshpande P.S.: ‘Color, scale, and rotation independent multiple license plates detection in videos and still images’, Math. Probl. Eng., 2016, 2016, 14 pages
[9] Masood S.Z., Shu G., Dehghan A., \textit{et al.}: ‘License Plate Detection and Recognition Using Deeply Learned Convolutional Neural Networks’, arXiv preprint arXiv:1703.07330, March 2017

[10] Cabaret L., Lacassagne L., Oudni L.: ‘A review of world’s fastest connected component labeling algorithms: speed and energy estimation’. Int. Conf. Design & Architectures for Signal & Image Processing, 2014, doi: 10.1109/DASIP.2014.7115641

[11] Facciolo G., Limare N., Meinhardt E.: ‘Integral images for block matching’, \textit{Image Process. On Line}, 2014, 4, pp. 344–369

[12] Kim I.J.: ‘Multi-window binarization of camera image for document recognition’. IWFHR-9, 2004 October 26, pp. 323–327

[13] Kim M.-K.: ‘Adaptive thresholding technique for binarization of license plate images’, \textit{J. Opt. Soc. Korea}, 2010, 14, (4), pp. 368–375

[14] Hong J., Chen J.: ‘Location and binarization of license plate image based on modification’, \textit{Comput. Eng.}, 2007, 33, (8), pp. 175–177

[15] El-Mihoub T.A., Hopgood A.A., Nolle L., \textit{et al.}: ‘Hybrid genetic algorithms: a review’, \textit{Eng. Lett.}, 2006, 13, (2), pp. 124–137

[16] Wong M.L., Wong T.T.: ‘Implementation of parallel genetic algorithms on graphics processing units’, \textit{Intell. Evol. Syst.}, 2009, 187, pp. 197–216.

[17] Rafique M.A., Pedrycz W., Jeon M.: ‘Vehicle license plate detection using region-based convolutional neural networks’, \textit{Soft Comput.}, 2017, pp. 1–2, doi: 10.1007/s00500-017-2696-2

[18] Ren S., He K., Girshick R., \textit{et al.}: ‘Faster r-cnn: towards real-time object detection with region proposal networks’, NIPS, 2015, pp. 91–99

This is an open access article published by the IET under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0/) J. Eng., 2018, Vol. 2018, Iss. 2, pp. 104–116
doi: 10.1049/joe.2017.0815