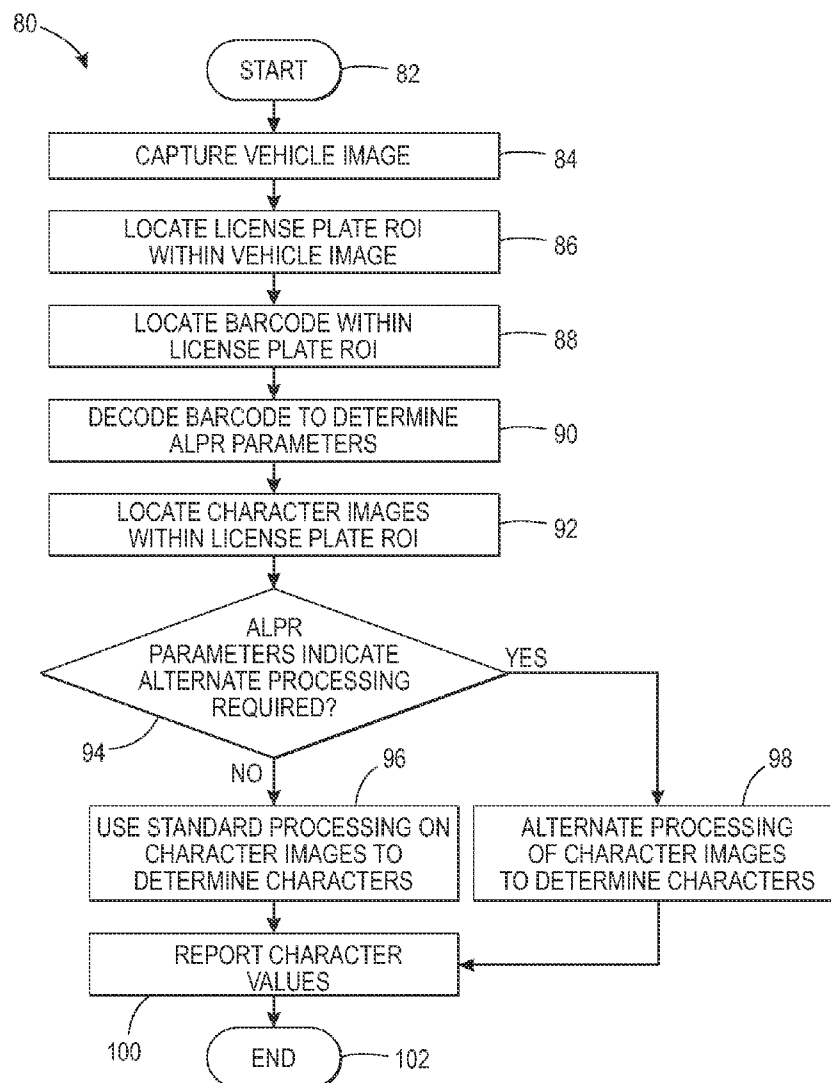




US 20140301606A1

(19) **United States**(12) **Patent Application Publication****Paul et al.**(10) **Pub. No.: US 2014/0301606 A1**(43) **Pub. Date: Oct. 9, 2014**(54) **DATA AUGMENTATION METHOD AND
SYSTEM FOR IMPROVED AUTOMATIC
LICENSE PLATE RECOGNITION****Publication Classification**(51) **Int. Cl.**
G06K 9/18 (2006.01)(52) **U.S. Cl.**
CPC **G06K 9/18** (2013.01)
USPC **382/105**(71) Applicant: **XEROX CORPORATION**, Norwalk,
CT (US)(72) Inventors: **Peter Paul**, Webster, NY (US);
Vladimir Kozitsky, Rochester, NY
(US); **Aaron Michael Burry**, Ontario,
NY (US); **Martin Edward Hoover**,
Rochester, NY (US); **Mark Cantelli**,
Germantown, MD (US)(73) Assignee: **Xerox Corporation**, Norwalk, CT (US)(21) Appl. No.: **13/857,657**(22) Filed: **Apr. 5, 2013**(57) **ABSTRACT**

Methods, systems, and processor-readable media for data augmentation utilized in an automatic license plate recognition engine. A machine-readable code can be associated with an automatic license plate recognition engine. The machine-readable code can be configured to define parameters that drive processing within the automatic license plate recognition engine to produce recognition results thereof and enhance a machine readability of a license plate recognized and analyzed via the automatic license plate recognition engine.



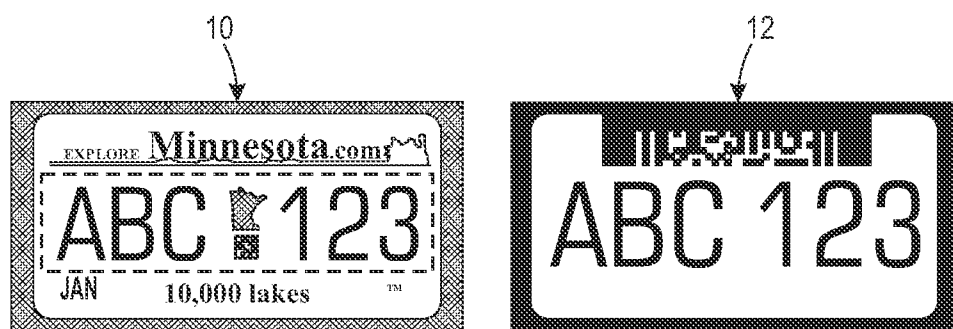


FIG. 1

14

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H	J	K	L	M	N	P	R	S	T	V	W	X	Y	Z
0	1181	0	0	0	0	0	0	0	1	0	0	1	1	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1190	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	7	1	0	0	0	0
2	0	0	1198	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
3	0	2	0	1192	0	0	0	0	0	0	0	1	0	0	0	0	0	0	4	0	0	0	0	0	0	1	0	0	0	0	0	0
4	0	7	0	0	1179	0	0	1	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	1	0	0	1
5	0	1	0	0	0	1197	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	1194	0	0	2	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	1196	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0	0	1184	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
9	0	0	0	0	0	0	2	4	0	1189	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A	0	0	0	0	0	0	1	0	0	0	1197	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0	2	0	0	1194	0	1	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0
C	0	197	0	0	0	0	0	0	0	0	0	0	1197	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0
D	1	0	0	0	0	0	0	0	0	0	0	0	0	1199	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1194	1	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1184	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G	9	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	1188	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
H	0	2	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	1194	0	0	0	1	0	0	0	0	0	0	1	0	0	0
J	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1199	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1198	0	0	0	0	0	0	0	0	0	0	0	0
L	0	2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1196	0	0	0	0	0	0	0	0	0	0	0
M	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1198	0	0	0	0	0	0	0	0	0	0
N	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1196	0	0	0	0	2	0	0	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1193	1	0	0	0	0	0	0	0
R	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1199	0	0	0	0	0	0	0	0
S	0	0	0	3	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1195	0	0	0	0	0	0	0
T	0	8	0	0	0	0	0	3	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	1187	0	0	0	0	
V	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1190	0	0	5	0	
W	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1196	0	1	0	
X	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1191	0	
Y	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	6	0	1	1189	
Z	0	0	3	0	1	0	0	5	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1190

2. GLE

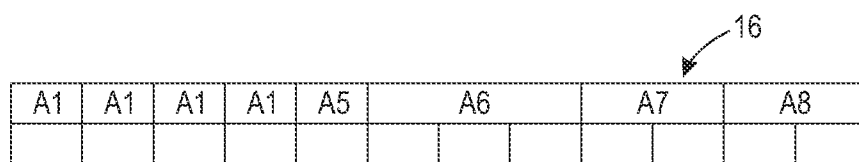


FIG. 3

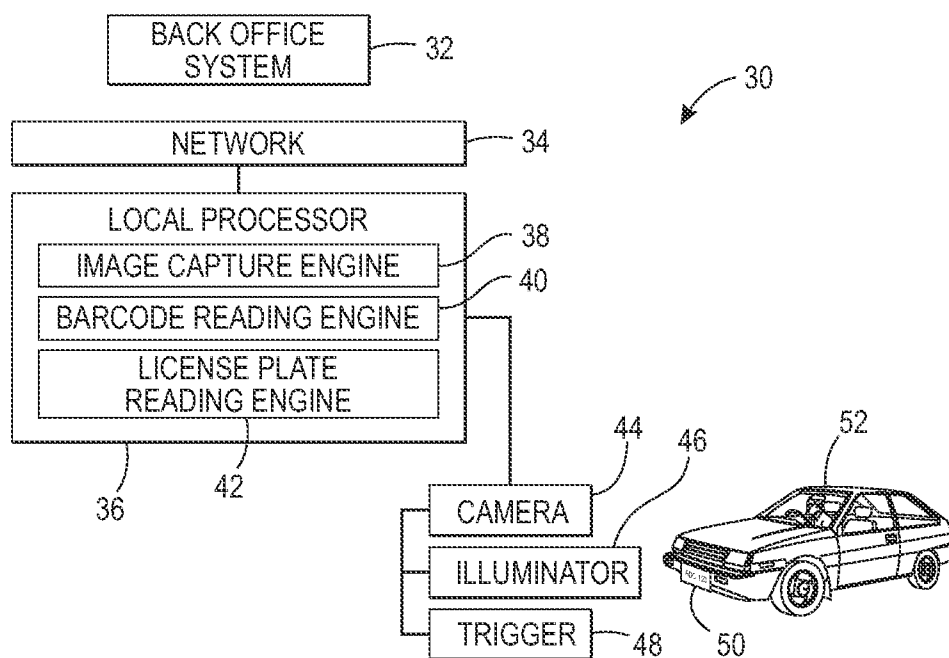


FIG. 4

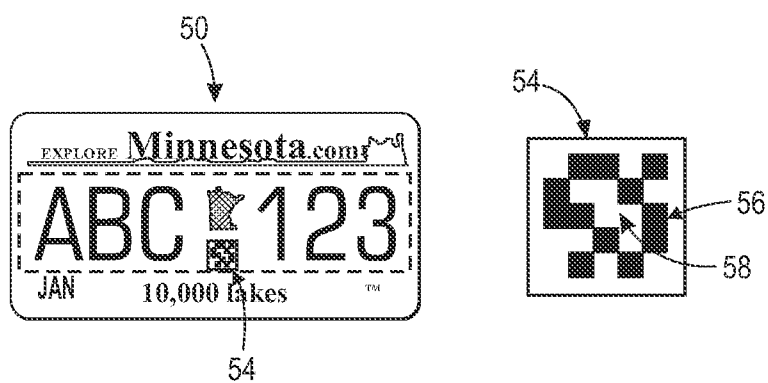


FIG. 5

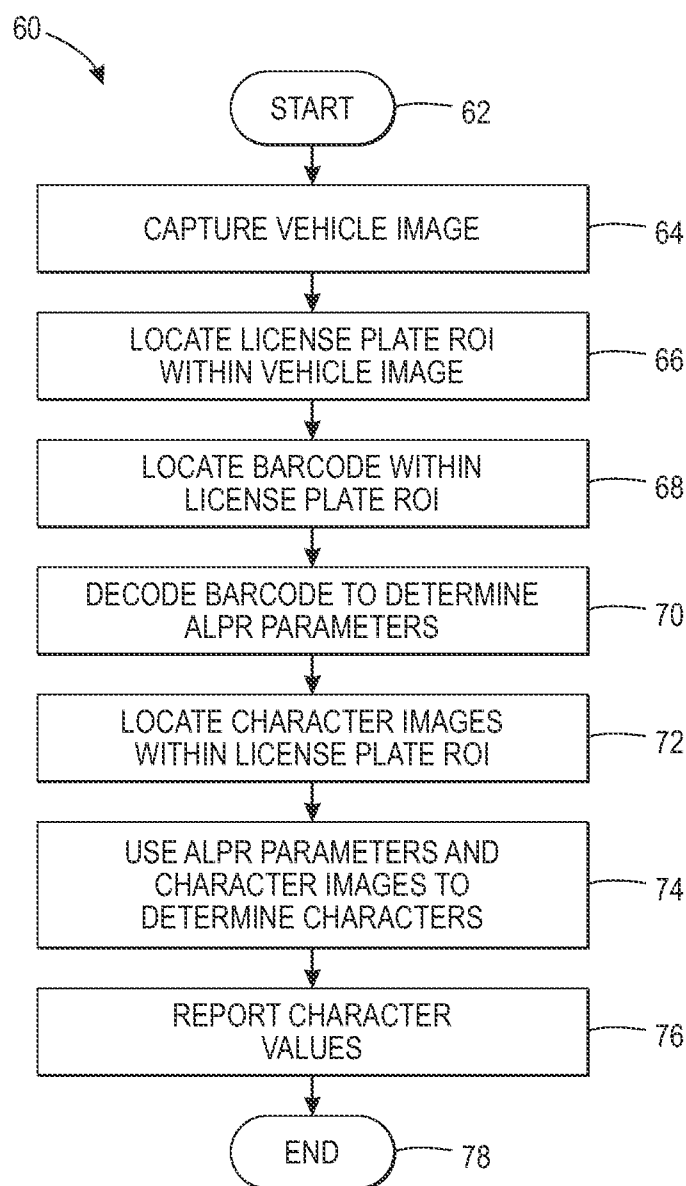


FIG. 6

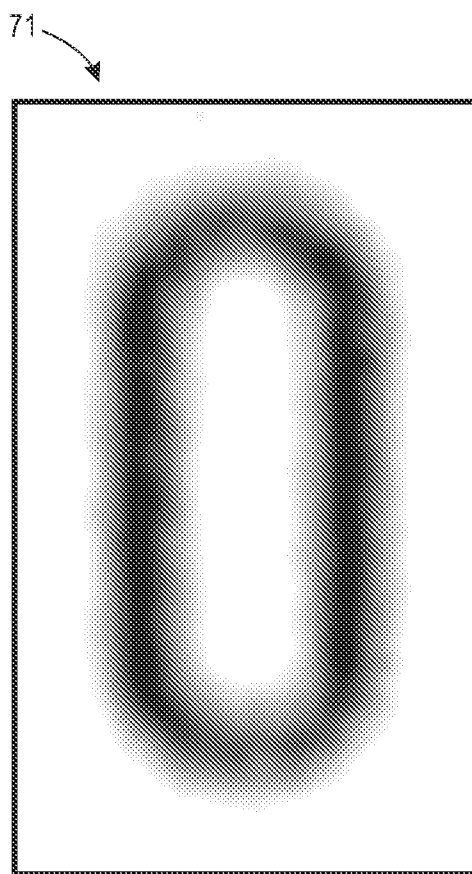


FIG. 7

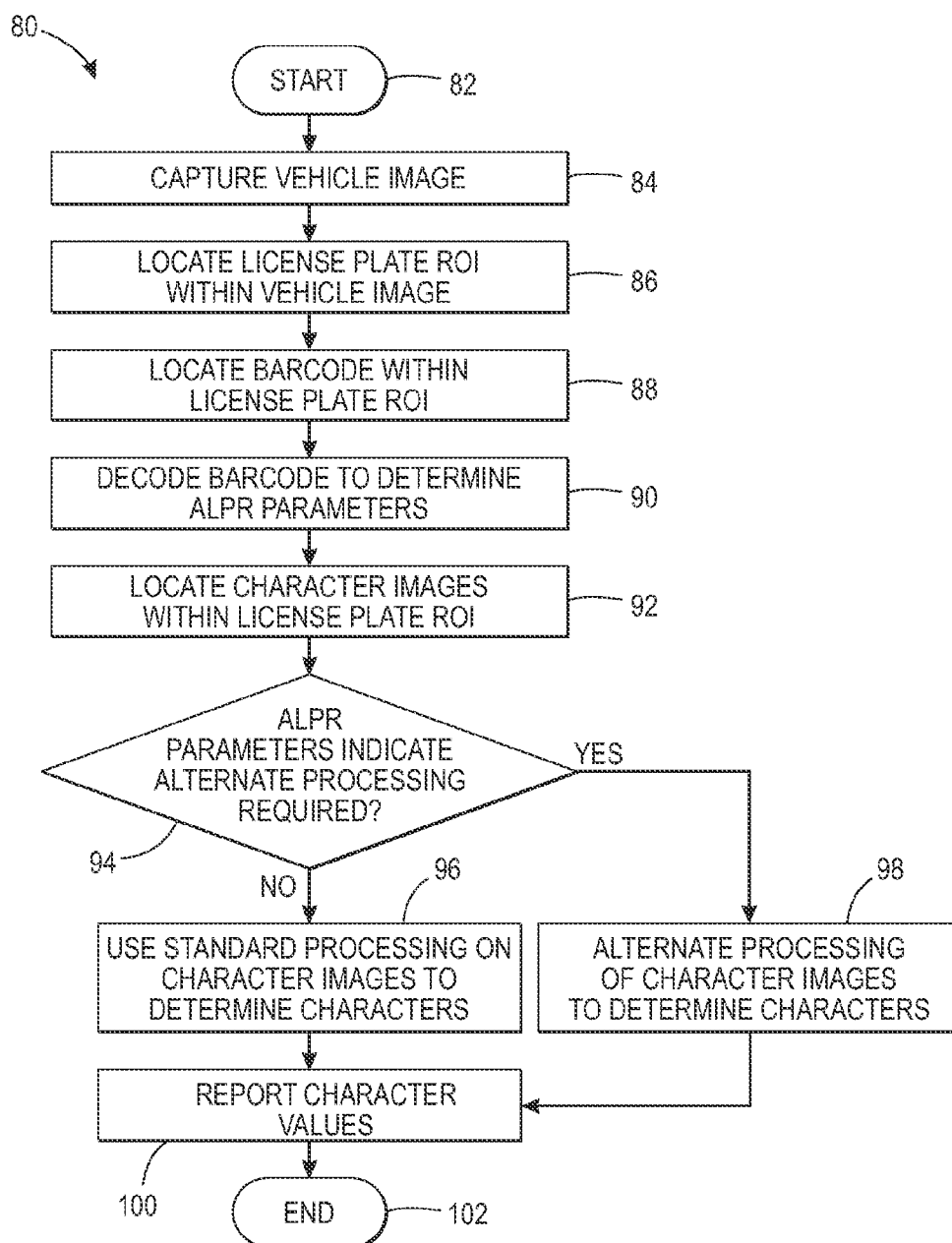


FIG. 8

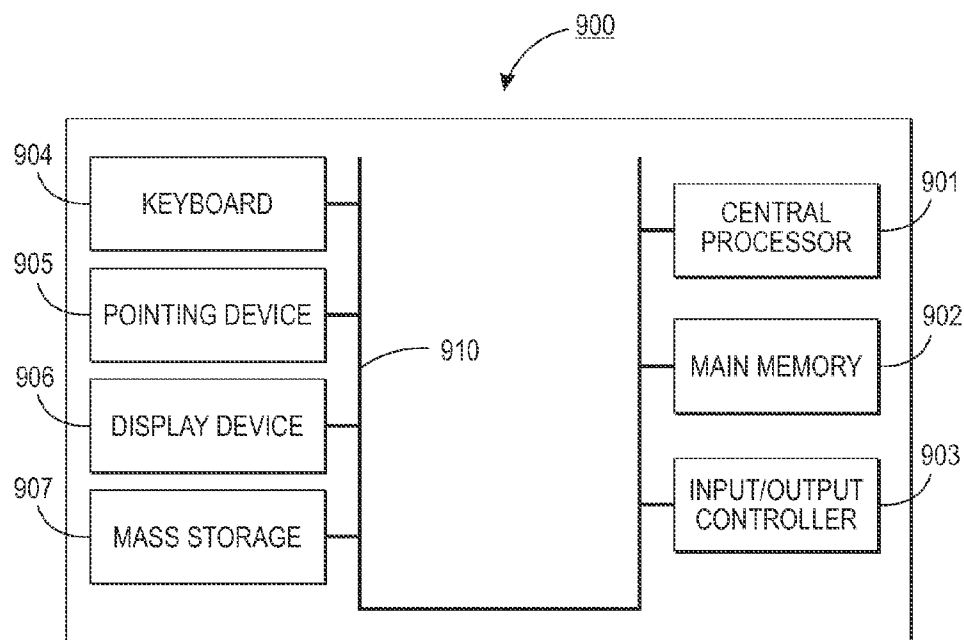


FIG. 9

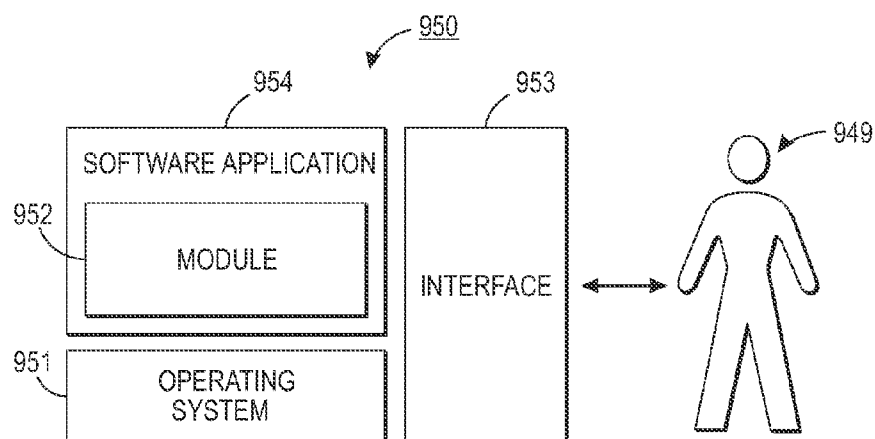


FIG. 10

DATA AUGMENTATION METHOD AND SYSTEM FOR IMPROVED AUTOMATIC LICENSE PLATE RECOGNITION

FIELD OF THE INVENTION

[0001] Embodiments are generally related to image-processing methods and systems. Embodiments are further related to video-based surveillance methods and systems. Embodiments are additionally related to the field of ALPR (Automatic License Plate Recognition).

BACKGROUND

[0002] ALPR systems generally include various image processing and computer vision techniques for: (1) locating a license plate within a scene, (2) segmenting characters on the plate, and (3) performing optical character recognition to determine the character sequence.

[0003] In addition, determining the jurisdiction such as the state and plate type is often required. ALPR is used in electronic tolling, photo enforcement, parking, and other transportation applications where the vehicle identity is required. Performance requirements for ALPR systems are very high and are ever increasing in this competitive market space. Performance errors can be categorized into common failure modes. One prevalent failure mode is referred to as “confused characters”. Characters are confused with each other due to image noise. For example, “B” is often confused with “8” and “D” is often confused with “0” in the presence of image blur. In another common failure mode, the license plate frame partially covers the license plate obscuring character strokes such as making a “T” look like a “I” and an “E” look like a “F”. Other common failure modes and desired augmenting data are known a priori, as well.

SUMMARY

[0004] The following summary is provided to facilitate an understanding of some of the innovative features unique to the disclosed embodiments and is not intended to be a full description. A full appreciation of the various aspects of the embodiments disclosed herein can be gained by taking the entire specification, claims, drawings, and abstract as a whole.

[0005] It is, therefore, an aspect of the disclosed embodiments to provide for an improved automatic license plate recognition engine.

[0006] It is another aspect of the disclosed embodiments to provide for data augmentation for an automatic license plate recognition engine.

[0007] The aforementioned aspects and other objectives and advantages can now be achieved as described herein. Methods, systems, and processor-readable media for data augmentation utilized in an automatic license plate recognition engine are disclosed. A machine-readable code can be associated with an automatic license plate recognition engine. The machine-readable code can be configured to define parameters that drive processing within the automatic license plate recognition engine to produce recognition results thereof and enhance a machine readability of a license plate recognized and analyzed via the automatic license plate recognition engine.

[0008] In another embodiment, a machine-readable code (e.g., a barcode) can be associated with an automatic license plate recognition engine, wherein the machine-readable code

is encoded with information based on a hierarchy, having hierarchy levels with higher levels that are most robust to image noises and lower levels that are less robust with respect to the image noises, but provide more information. The hierarchy is arranged such that the higher levels and the lower levels define parameters that drive processing within the automatic license plate recognition engine to produce recognition results thereof and enhance a machine readability of a license plate recognized and analyzed via the automatic license plate recognition engine.

[0009] To address a priori known failure modes and to include augmenting data, a small barcode can be employed, which encodes information that efficiently augments the ALPR system (e.g., an ALPR engine) in performing its task. This can be thought of as adding “differential information” that enhances the machine readability of the overall plate information. Such a barcode can be read using existing license plate reading cameras using either visible light or infrared illumination.

[0010] The highest level in the hierarchy can include data indicative of, for example, (1) whether the plate is a standard plate or non-standard plate; (2) whether the plate contains a potential for confusing characters; (3) whether the plate contains characters subject to plate frame occlusion; (4) the State jurisdiction; (5) whether the plate has a logo; (6) whether the plate has stacked characters; (7) the Font used; (8) the best RGB-to-Gray mapping to use; (9) the number of characters on the license plate; and/or (10) whether each character is a letter or a number. Such conditions can define parameters, which drive various processing within the ALPR engine to produce improved recognition results.

BRIEF DESCRIPTION OF THE FIGURES

[0011] The accompanying figures, in which like reference numerals refer to identical or functionally-similar elements throughout the separate views and which are incorporated in and form a part of the specification, further illustrate the present invention and, together with the detailed description of the invention, serve to explain the principles of the present invention.

[0012] FIG. 1 illustrates a pictorial view of a prior art barcode system readable under infrared illumination, with the left license plate subject to visible light and the right license plate subject to IR light;

[0013] FIG. 2 illustrates a sample ALPR confusion matrix in accordance with a preferred embodiment;

[0014] FIG. 3 illustrates a sample ALPR augmentation data bit format in accordance with a preferred embodiment;

[0015] FIG. 4 illustrates a high-level block diagram of an ALPR system that can be implemented in accordance with a preferred embodiment;

[0016] FIG. 5 illustrates a pictorial view of a license plate with a license plate sticker that can include a barcode and ALPR augmentation data, in accordance with a preferred embodiment;

[0017] FIG. 6 illustrates a high-level flow chart of operations depicting logical operational steps of a method for providing ALPR augmentation data, in accordance with the disclosed embodiments;

[0018] FIG. 7 illustrates an example character image in accordance with a preferred embodiment;

[0019] FIG. 8 illustrates a high-level chart of operations depicting logical operational steps of an alternative method for providing ALPR augmentation data, in accordance with an alternative embodiment;

[0020] FIG. 9 illustrates a schematic view of a computer system, which can be implemented in accordance with one or more of the disclosed embodiments; and

[0021] FIG. 10 illustrates a schematic view of a software system including an ALPR module using augmented data, an operating system, and a user interface, in accordance with the disclosed embodiments.

DETAILED DESCRIPTION

[0022] The particular values and configurations discussed in these non-limiting examples can be varied and are cited merely to illustrate at least one embodiment and are not intended to limit the scope thereof.

[0023] The embodiments will now be described more fully hereinafter with reference to the accompanying drawings, in which illustrative embodiments of the invention are shown. The embodiments disclosed herein can be embodied in many different forms and should not be construed as limited to the embodiments set forth herein; rather, these embodiments are provided so that this disclosure will be thorough and complete, and will fully convey the scope of the invention to those skilled in the art. Like numbers refer to like elements throughout. As used herein, the term “and/or” includes any and all combinations of one or more of the associated listed items.

[0024] The terminology used herein is for the purpose of describing particular embodiments only and is not intended to be limiting of the invention. As used herein, the singular forms “a”, “an” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will be further understood that the terms “comprises” and/or “comprising,” when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof.

[0025] Automatic License Plate Recognition (ALPR) systems are generally composed of (A) cameras, illuminators, and vehicle triggers which capture images of vehicles; and (B) image processing and computer vision systems which perform processing which: (1) locates the license plate in a scene; (2) segments the characters within the license plate; and (3) performs optical character recognition (OCR) to determine the character sequence on the license plate. In addition, the State jurisdiction, the plate type, and other information is often required to be determined. ALPR systems are used extensively in transportation systems such as electronic tolling systems and photo enforcement systems (speed enforcement or red light enforcement) where vehicle identity is required.

[0026] The performance requirements for ALPR systems are ever increasing. ALPR mistakes can lead to unidentified vehicles causing lost tolling revenue, for example, or worse, can lead to erroneous violations which cause frustration with transportation system patrons who need to go through a process to have the violation vacated. Requirements of 99.5% accuracy and beyond are typically required for current request for proposals put out by transportation agencies.

[0027] One approach to improving the machine readability of license plates is to install RFID transponders on the license plates themselves. Some countries are exploring this option,

but it is seen as a much longer-term solution and may be superseded by other technologies as they are developed (Dedicated Short Range Communications, DSCR, for example). Another barrier to RFID transponder adoption on license plates is public acceptance. Even in electronic tolling systems using RFID transponder tags (E-ZPass® in New York, for example), which have clear patron benefit (quicker entrance and exit to the toll road way) adoption is not universal. It is estimated that RFID transponders for tolling would never exceed 70%-80% without government mandate.

[0028] Other barriers include RFID transponder life and cost, with transponders requiring replacement at some periodic interval. Since different manufacturers of RFID systems use different frequency bands and different protocols, interoperability problems exist in electronic tolling systems that use RFID transponders as a mechanism to identify vehicles. While standardization could potentially alleviate this problem for license plate mounted RFID transponders, significant federal government action would be required in a domain (license plates) that has traditionally been a responsibility of the states. Even current electronic tolling systems are moving away from using RFID transponders for all of these reasons.

[0029] Another approach to improving the machine readability of license plates has been proposed: the use of barcodes. Such a system employs a large barcode that covers existing text on the license plate but is only readable under infrared light. FIG. 1 depicts such a prior art system. On the left side of FIG. 1, a license plate 10 is shown under visible light illumination. License plate 10 looks like a standard license plate to a human observer, such as a police officer who might need to examine the plate to execute their duties. On the right side of the figure is the same license plate 12 illuminated under infrared illumination. Note that a large barcode is prominently visible on the plate.

[0030] While this approach seems promising, it has some deficiencies. Most of all is the size of the barcode. In order to fit the amount of information to encode all the license plate information in the barcode, the barcode needs to contain roughly 12-24 bytes of information depending on the level of error correction in the code and the length of the character sequence on the license plate. This is for US plates. For plates that can have much larger character sets, the amount of information would be much higher. In order to contain this amount of information on a barcode that is robustly readable at highway speeds with typical image noises seen by license plate reading cameras and maintaining the use of typical license plate reading cameras (not increasing the cost by increasing the resolution of the cameras, for example), the barcode is required to be physically large.

[0031] In the example shown in FIG. 1, the barcode is so large that it obscures the State jurisdiction designation on the plate in the IR image. This is important since for many electronic tolling applications, only one image is taken per plate. The image is taken using IR light so the only image available is the image on the right side of FIG. 1. However, the requirements of the electronic tolling system are that a human may review the imagery to verify a tolling transaction or other violation (e.g., red light violation). A human, viewing only the right side of FIG. 1, would not be able to determine the State jurisdiction of the license plate because the large barcode occludes the State designation. Further, the imagery must provide evidence to assess a toll or issue a violation. This evidence is usually provided to the patron in printed form or

as a web link in order for the patron to see that they did indeed make the violation or is correctly paying the toll. Given the image on the right side of FIG. 1, a patron may argue that it is not their vehicle since the State jurisdiction cannot be determined by a person viewing the IR image.

[0032] While legislation could possibly be written to alleviate this problem or the plate could be completely redesigned to accommodate the large barcode, these solutions are not straightforward. For example, States use their license plate designs as revenue sources from patrons who are willing to pay a higher fee for a designation such as being a university supporter (for example, a "Texas A&M" license plate with a school mascot on the plate) or supporting some cause ("State Park Supporter" license plate with a special tree logo on the plate). Putting constraints on plate design may impede this revenue source and may not be acceptable.

[0033] Another problem with this approach is that it requires infrared cameras and illumination systems. While many existing electronic tolling and other license plate reading systems are IR systems, trends seem to indicate using visible light color based systems to gather enhanced evidence for toll and violation enforcement (color of car, etc.) for new system rollouts. Others have also proposed license plate barcode systems [2], [3]. However, no system we know of describes encoding the kind of information proposed in our invention: information used to aid an ALPR system in reading the characters on the plate and determining other license plate information.

[0034] The disclosed embodiments are generally directed toward a technique for determining a small extra set of information that works in conjunction with an ALPR system to provide improved overall machine readability of the plates, and for affixing this small amount of extra information on the license plate using a barcode on a small sticker placed in an unused portion of the plate. Additionally, such embodiments allow the barcode sticker to be read under either visible light or IR illumination.

[0035] Since the present inventors have designed and constructed an ALPR system, they are in a position to be aware of the types of extra information most valuable in order to augment ALPR systems. The disclosed approach, however, does not seek to redundantly encode license plate information. Instead, such an approach, as will be discussed in greater detail herein, seeks to augment an ALPR system with key information that will improve its performance.

[0036] The same information that is visually observable on the license plate is not simply repeated into the barcode. Instead, a smaller amount of key information is placed in the barcode to allow for dual reading of the barcode and license plate characters and to provide for the improved accuracy of character recognition while using a small footprint barcode. Some characters are almost always read correctly by the ALPR system; why waste valuable space by encoding them into the barcode? The barcode and ALPR system can work together, not as two completely separate systems redundantly encoding the same information.

[0037] Note that in the transportation business, clients are typically state and local governmental agencies that manage roadway systems or parking lots or spaces. Thus, any stickers that would be mounted to license plates would be fully authorized by the governmental agency with full state approval and typically not be subject to any state restriction of sticker placement on license plates. In a scenario in which the disclosed embodiments may be implemented, a governmental

agency (e.g., such as E-ZPass® in New York), would physically provide the stickers to their patrons just as they currently provide RF transponders to their patrons. The stickers would simply be an extension of the RFID-based system for agencies that want improved accuracy and to perhaps slowly phase out costly RFID transponders.

[0038] The dimensions of a license plate with its barcode can be used to determine the information density while maintaining robust barcode reading using license plate reading cameras at highways speeds in license plate reading geometries. The information density of the license plate barcode system as depicted in the example shown in FIG. 1 can be derived as follows:

[0039] A. Barcode Area:

License Plate Area=6 inches×12 inches=72 sq inches

Barcode Area (including border)=1/6 License Plate→12 sq inches

Barcode Area (without border)=1.1 in ×4.6 in→5 sq inches

[0040] B. Information Required:

26 Letters+10 Numbers=36 Characters→ $(\log 36)/(\log 2)=5.17$ bits

50 States→ $(\log 50)/(\log 2)=5.64$ bits

32 Plate Types→ $(\log 32)/(\log 2)=5$ bits

7 Characters+1 State+1 Plate Type→ $(7 \times 5.17 \text{ bits}) + 5.64 \text{ bits} + 5 \text{ bits} = \text{Approx } 50 \text{ bits}$

w/Error Detection & Correction→Factor of 2

[0041] →Approx 100 bits

[0042] C. Information Contained in 3M Barcode

Analysis of Barcode Image→5 rows×24 cols→120 bits

[0043] D. Information Density

Including Border: 120 bits/12 sq inches=10 bits/sq inch

Without Border: 120 bits/5 sq inches=24 bits/sq inch

[0044] Note that a simple enumeration of the data required to be in the barcode shows that approximately 100 bits is required to be in the barcode as shown in section B above. Examination of the barcode reveals 120 bits as shown in section C above, which is roughly consistent with the simple enumeration calculation. From FIG. 1, the barcode seems to have very large border regions including what looks like barcode docking lines to the left and to the right of the barcode. Optimization of the border region and barcode docking lines may reduce the non-information carrying border region. With this in mind the information density for robust barcode reading on license plates seems to be in the range of 12-24 bits/sq inch. This range of barcode spatial information density is used in the remainder of this invention disclosure. To avoid any potential confusion, note that the current invention is aimed at reducing the total required information contained in the barcode, but for reference will assume the level of information density within the barcode.

[0045] Information can be encoded in the barcode based on specific known failure modes of an ALPR system. In this way a small amount of information is provided on the barcode sticker, which will aid the ALPR system in its operation. The small amount of additional information can be stored in a relatively small barcode. The size of the required barcode will be determined using the amount of information stored and the data density for robust reading on license plates. The specific failure modes or added information can include, for example:

A1: Plate Contains Confusing Characters

A2: Plate Contains Characters Susceptible to Frame Occlusion

A3: Standard or Non-Standard Plate

A4: Plate Contains Logo or Not

A5: Plate Contains Stacked Characters or Not

A6: State Jurisdiction

A7: Font Used

A8: Best RGB-to-Gray Mapping to use

[0046] Confusing Characters are those characters which can easily be confused with each other, especially under typical license plate reading noises such as blur and geometric distortions. These include mistaking a “B” for an “8”, a “D” for a “0”, a “Y” mistaken for a “V”, and a “4” for an “A”. Of course, this is very font dependent. Note that a “B” is very similar to an “8” plus some higher spatial frequency terms. If these higher spatial frequency terms are removed due to blur, then a “B” is confused with an “8”. Note that typically an “8” is not mistaken for a “B” and a “0” is not typically mistaken for a “D” (i.e., the confusion is really “one-way”).

[0047] FIG. 2 illustrates a “Confusion Matrix”, which is a matrix showing which characters get mistaken for which other characters in the ALPR system. Such a Confusion Matrix can be obtained from a License Plate Reading system on real license plate data taken from a tolling system. The top row in the example Confusion Matrix indicated in FIG. 2 labels the correct character, while the first column labels the recognized characters. Note that the matrix is strongly diagonal which means that the ALPR system does not make very many mistakes. The Confusion Matrix very explicitly shows the failure modes and challenging characters of the ALPR system. By simply providing 1-bit of information to the ALPR system, which encodes whether the plate has characters that are susceptible to the confusing characters failure mode, the ALPR system can reduce these specific mistakes. In one embodiment, for plates that have more than one confusing character, more than 1-bit is used to encode the locations of these characters as well.

[0048] Characters susceptible to frame occlusion include a “T” whose top stroke may be occluded by the plate frame and then may be mistaken for a “1”. 1-bit is required for this purpose.

[0049] A Non-Standard plate is a plate that does not conform to a general standard for a particular state. This includes, for example, vanity plates. Vanity plates are often not required to be read by license plate reading systems. This is often written in the RFP and into the eventual contract with the client agency. 1-bit is required for this purpose.

[0050] Logos are typically used to delimit letters and/or number on the left side of the plate from those on the right side of the plate for aesthetic purposes. A state logo, for example, is depicted in the middle of the license plate 10 of FIG. 1. Since Logos are in the same plate region as characters to be read, they can cause confusion in character segmentation and character recognition. 1-bit is required for this purpose.

[0051] Stacked characters are a set of smaller characters stacked on top of each other where the stack is located in a

typical location where the license plate characters are expected to be. Since the stack is in this location, they can cause confusion in character segmentation and in character recognition. 1-bit is required for this purpose.

[0052] State jurisdiction is the state that the plate is from. Typically at most, 6-7 states are required to be identified. Any other state outside of this 6-7 state set can be grouped together. Thus 8 values are possible, which requires 3-bits.

[0053] The font used is the font used to write the characters on the license plate. Since the state is encoded, a smaller number of fonts need to be encoded. Assume 4 fonts (which is really 4 per state, since state is being encoded), which requires 2-bits.

[0054] Best RGB-to-Gray mapping is a hint as to how to process the RGB image in order to use the ALPR engine. While the full color image can be used, for some cases one color plane provides the best contrast. However, that one color plane may be different for different states or even different plate types within a state. Thus 4 values are required to state whether the best encoding is R-to-Gray, G-to-Gray, B-to-Gray, or (R+G+B)-to-Gray. This requires 2-bits.

[0055] FIG. 3 depicts the ALPR augmentation data bit format to encode the extra information used to improve the performance of the ALPR engine. For example, a bit sequence of 01010-010-01-10 would indicate: No Confusing Characters, Plate Occlusion Susceptibility, Standard Plate, Logo Present, No Stacked Characters, State Number 2, Font Number 1, and use the B-to-Gray mapping for color images. The ALPR engine can use these hints to set processing parameters in the processing engine in order to improve the ALPR accuracy.

[0056] This data and format requires 12 bits. Assuming that error detection and correction coding doubles the number of bits required yields a total of 24 bits to be encoded in the barcode. With the barcode data density determined in Section 2 of 12-24 bits/sq in, this augmentation data requires 1-2 sq in of plate area for the barcode sticker. Note that this is much smaller than the barcode depicted in the example of FIG. 1, which requires up to 12 sq in of plate area. Furthermore, since this information is just the right extra information to improve the accuracy of the ALPR system, it may, in fact, provide a better overall machine readability of the plate compared to a system that attempts to redundantly encode all the information contained in the license plate on to the barcode.

[0057] Note that this is simply one embodiment. Of course, more or fewer bits could be allocated to the different fields of the Augmentation Data in order to encode more or less specific information. For example, the sequence or location of multiple confusing characters could be encoded, or more or fewer bits could be allocated to encode more or fewer states. Furthermore, additional fields could be defined to encode, for example, the plate type, positive or negative text, or other information in order to aid the ALPR system. An additional field for the number of characters on the plate may be very useful. For 1 to 8 characters, this would require 3 bits. However, to further reduce the number of bits required for encoding the number of characters on the plate, a baseline number of characters plus offset can be used. If the baseline number of characters is 4, then only 2 bits can be used to encode a value from 4 to 7+ for the number of characters on the plate. Another additional field could be used to encode whether a letter or a number is at a particular location in the character sequence. For an 8 character field this would require 3 bits.

[0058] In another embodiment, the augmentation data depicted in FIG. 3 would be at the highest level in a data hierarchy. More details would be at a lower level in a data hierarchy and that information could be stored in the same barcode areas, but require higher resolution cameras in order to read the data at the lower levels in the hierarchy. In this way, standard license plate reading cameras as currently deployed in the field can read the standard density information at the highest level in the data hierarchy (such as the data depicted in FIG. 3.), while the more detailed information would require higher resolution cameras. The barcode would thus have some level of “future-proof” as higher resolution cameras get rolled out into transportation applications. The more detailed information could include, for example, character locations, a larger number of state designations, and plate type designations.

[0059] In another embodiment, the augmentation data can be updated over time based on the current operation of the overall ALPR system and real statistics of the deployment of the ALPR system. New stickers would then be mailed out to the patrons to affix to their license plates. Note that in many states, inspection or registration stickers are put on license plates on a periodic basis, often yearly, so patrons are familiar with affixing new stickers on their license plates periodically.

[0060] FIG. 4 illustrates a high-level system diagram of an ALPR system 30 that can be implemented in accordance with a preferred embodiment. The system 30 depicted in FIG. 4 includes a vehicle 52 with a license plate 50, which has a barcode sticker containing augmentation data for an automatic license plate reading engine, along with a trigger 48, a camera 44, and an illuminator 46 for capturing an image. System 30 further includes a local processor 36 that includes an image capture engine 38, a barcode reading engine 40, and a license plate reading engine 42. System 30 can further include a network 34 (e.g., a local wireless network, the Internet, cellular communications network, other data network, etc.), and a back office system 32 for processing transactions and managing patron accounts. The local processor 36 can communicate with the back office system 32 via the network 34.

[0061] In FIG. 4, the license plate 50 is depicted located on the front of the vehicle 52, but the license plate 50 can be located on the rear of the vehicle 52 as well. In one scenario, the vehicle 52 enters a license plate reading zone which contains a trigger device 48 which controls an illuminator 46 which illuminates the license plate region of the vehicle 52, and a camera 44 which images the license plate 50 on the vehicle 52. The camera 44 is connected to a local processor unit 38, which contains an image capture engine 38, a barcode reading engine 40, and the license plate reading engine 42.

[0062] The image capture engine 38 controls the trigger 48, illuminator 46, and camera 44 in order to properly image the vehicle 52 and the license plate 50. The barcode reading engine 40 locates the barcode region in the captured image and decodes the barcode in order to determine a parameter set which is sent to the license plate reading engine 38. The license plate reading engine 38 uses the parameter set determined by the barcode reading engine 40 to determine the correct character sequence and state jurisdiction indicated on the license plate 50 in order to determine a unique identifier for the vehicle 52.

[0063] The license plate 50 and state jurisdiction data can then be sent by the local processor 36 over the network 34 to the back office system 32. The back office system 32 pro-

cesses the license plate and state jurisdiction data and assesses a toll or otherwise interacts with a patron account or takes other transportation application specific actions.

[0064] FIG. 5 illustrates a pictorial view of a license plate 50 with a license plate sticker 54 that can include a barcode 56 and ALPR augmentation data 58, in accordance with a preferred embodiment. The left side of FIG. 5 depicts the sticker 54, which contains the barcode 56 on the license plate 50 in a location that does not detract from the visual observation of the license plate 50. The right side of FIG. 5 depicts the actual sticker 54 with the barcode 56. The ALPR augmentation data 58 is depicted as the data contained by the barcode 56. The configuration shown in FIG. 5 thus uses data within the barcode 56 that is derived from that information which best helps the ALPR improve its performance.

[0065] Note that the disclosed embodiments can be implemented in different scenarios. For example, the license plate sticker 54 does not necessarily need to be directly located on the license plate 50. In some situations, the sticker 54 can be still be located on the rear of the vehicle 52 somewhere, for example, on the back bumper, and remain within the field of view of the ALPR camera 44. An alternate location for the sticker 54 would be on a license plate frame/holder. An alternative to the sticker is encoding of the barcode 56 on a license plate frame/holder through painting or other marking or through etching. Tolling agencies, for example, could provide plate frames rather than stickers as a patron option.

[0066] FIG. 6 illustrates a high-level flow chart of operations depicting logical operational steps of a method 60 for providing ALPR augmentation data, in accordance with the disclosed embodiments. As indicated in block 62, the process can begin. Then, as shown at block 64, an operation can be implemented for capturing an image of, for example, vehicle 52 shown in FIG. 4. This can be performed in a standard manner. That is, a vehicle enters the image capture zone. The vehicle triggers a sensor, which could be a ground loop, or other sensor located on or in the roadway surface, or it could be a laser triggering mechanism, or it could be a video based triggering mechanism. The trigger sensor then activates an illuminator unit, which illuminates the license plate on the vehicle. The trigger sensor also activates a camera, which takes a digital picture of the vehicle. Other standard methods of capturing an image of a vehicle and its license plate are also possible, including using continuous video capture and extracting video frames that contain vehicles and license plates.

[0067] As depicted next at block 66, image processing and computer vision techniques can be employed to locate the license plate region of interest (ROI) within the vehicle image. This can be performed in ALPR systems, for example, using a standard shape and size of the license plate as cues.

[0068] Thereafter, as shown at block 68, the barcode image can be located within the license plate ROI. This can be accomplished using a priori knowledge of the location of the barcode within the license plate ROI using edge references and known pixel-to-edge distance relationships. The location of the barcode can be further determined using correlation template matching and/or more advanced machine learning techniques.

[0069] Next, as shown at block 70, the barcode data can be decoded. The barcode encoding scheme can be a known 1D or 2D scheme such as Code 39, Code 128, Data Matrix, QR

Codes, PDF417, Data Glyphs, or others. The barcode can be decoded to produce a set of ALPR parameters, as shown in FIG. 3, for example.

[0070] Thereafter, as depicted at block 72, the character images can be segmented from the license plate ROI image. Next, as illustrated at block 74, the ALPR parameters and the character images can be employed to determine the characters on the license plate. For example, if the image 71 shown in FIG. 7 is captured as one of the character images and the “Confused Character” Bit, A1, was set in the ALPR parameter set as read from the barcode, then the character would be declared a “D”, if the A1 bit is not set then the character would be declared a “O”. Note that in an alternate embodiment the confidence outputs of the character recognition engine could be used in conjunction with the ALPR parameters in the declaration of character values.

[0071] Thereafter, as depicted at block 76, the character values are reported. The method then processes the next vehicle image and/or the process can terminate, as shown at block 78.

[0072] FIG. 8 illustrates a high-level chart of operations depicting logical operational steps of an alternative method 80 for providing ALPR augmentation data, in accordance with an alternative embodiment. The steps or logical operations of method 80 shown at blocks 84, 86, 88, 90, and 92 in FIG. 8 are respectively analogous to the steps/logical operations of blocks 64, 66, 68, 70, and 72 shown in FIG. 6. FIG. 8 depicts an alternate embodiment where separate character image processing algorithms can be selected based on information located in the barcode data payload.

[0073] Following processing of the operation shown at block 92, a test can be performed as depicted at block 94 to determine if ALPR parameters indicate that alternate processing is required. If so, then the operation described at block 98 is processed wherein alternate processing of character images can be performed to determine characters. If not, standard processing of character images to determine characters can be performed, as depicted at block 96. The operation illustrated at block 100 is analogous to the operation of block 76 shown in FIG. 6. The method 80 of FIG. 8 can then terminate, as indicated at block 102.

[0074] An example of how the method 80 shown in FIG. 8 can be implemented is where, for example, a bit in the barcode data format indicates that the license plate is susceptible to plate frame occlusion or not. If the bit indicates that the license plate is not susceptible to plate frame occlusion, then “standard processing” as depicted at block 96 is used to determine the characters; if the bit indicates that the license plate is susceptible to plate frame occlusion, then the “alternate processing” indicated at block 98 can be used.

[0075] This “alternate processing” may be implemented in a form where more detailed analysis is performed at the border regions of the character image including applying a separately trained classifier for characters partially occluded by the plate frame. If the more detailed processing were performed on every plate, then overall system computational burden would be very high. If the more detailed analysis is only performed when the license plate is susceptible to plate frame occlusions, then more efficient processing takes place since you only do the more computationally burdensome processing when potentially needed. Note that “one bit in the data format” may result in many bits in the eventual barcode due to the application of barcode standards and standard error control coding.

[0076] Another example of embodiment 8 includes the case where a bit in the data format includes whether the plate is a “vanity plate” or not. If it is not a vanity plate, then “standard processing” for character images would occur. If it is a vanity plate, then the “alternate processing” may include a completely different license plate identification system such as using “image signature” or “image hashes” and a database look up for known plate-hash pairs.

[0077] As will be appreciated by one skilled in the art, the disclosed embodiments can be implemented as a method, data-processing system, or computer program product. Accordingly, the embodiments may take the form of an entire hardware implementation, an entire software embodiment or an embodiment combining software and hardware aspects all generally referred to as a “circuit” or “module” or “engine”. For example, an “engine” may be a software module.

[0078] Furthermore, the disclosed approach may take the form of a computer program product on a computer-usable storage medium having computer-usable program code embodied in the medium. Any suitable computer readable medium may be utilized including hard disks, USB flash drives, DVDs, CD-ROMs, optical storage devices, magnetic storage devices, etc.

[0079] Computer program code for carrying out operations of the present invention may be written in an object oriented programming language (e.g., JAVA, C++, etc.). The computer program code, however, for carrying out operations of the present invention may also be written in conventional procedural programming languages such as the “C” programming language or in a visually oriented programming environment such as, for example, Visual Basic.

[0080] The program code may execute entirely on the user’s computer or mobile device, partly on the user’s computer, as a stand-alone software package, partly on the user’s computer and partly on a remote computer or entirely on the remote computer. In the latter scenario, the remote computer may be connected to a user’s computer through a local area network (LAN) or a wide area network (WAN), wireless data network e.g., WiFi, WiMax, 802.11x, and cellular network or the connection can be made to an external computer via most third party supported networks (e.g., through the Internet via an internet service provider).

[0081] The embodiments are described at least in part herein with reference to flowchart illustrations and/or block diagrams of methods, systems, and computer program products and data structures according to embodiments of the invention. It will be understood that each block of the illustrations, and combinations of blocks, can be implemented by computer program instructions. These computer program instructions may be provided to a processor of a general-purpose computer, special purpose computer, or other programmable data-processing apparatus to produce a machine such that the instructions, which execute via the processor of the computer or other programmable data-processing apparatus, create means for implementing the functions/acts specified in the block or blocks discussed herein such as, for example, the various instructions shown with respect to the methods 60, 80 respectively illustrated in FIGS. 6 and 8.

[0082] These computer program instructions may also be stored in a computer-readable memory that can direct a computer or other programmable data-processing apparatus to function in a particular manner such that the instructions stored in the computer-readable memory produce an article of

manufacture including instruction means which implement the function/act specified in the block or blocks.

[0083] The computer program instructions may also be loaded onto a computer or other programmable data-processing apparatus to cause a series of operational steps to be performed on the computer or other programmable apparatus to produce a computer implemented process such that the instructions which execute on the computer or other programmable apparatus provide steps for implementing the functions/acts specified in the block or blocks.

[0084] FIGS. 9-10 are provided as exemplary diagrams of data-processing environments in which embodiments of the present invention may be implemented. It should be appreciated that FIG. 9-10 are only exemplary and are not intended to assert or imply any limitation with regard to the environments in which aspects or embodiments of the disclosed embodiments may be implemented. Many modifications to the depicted environments may be made without departing from the spirit and scope of the disclosed embodiments.

[0085] As illustrated in FIG. 9, the disclosed embodiments may be implemented in the context of a data-processing system 900 that includes, for example, a central processor 901 (or other processors), a main memory 902, an input/output controller 903, and in some embodiments, a USB (Universal Serial Bus) or other appropriate peripheral connection. System 900 can also include a keyboard 904, an input device 905 (e.g., a pointing device such as a mouse, track ball, pen device, etc.), a display device 906, and a mass storage 907 (e.g., a hard disk). As illustrated, the various components of data-processing system 900 can communicate electronically through a system bus 910 or similar architecture. The system bus 910 may be, for example, a subsystem that transfers data between, for example, computer components within data-processing system 900 or to and from other data-processing devices, components, computers, etc.

[0086] It can be appreciated that in some embodiments, the processor 901 may process instructions from the image capture engine 38, the barcode reading engine 40, and the license plate reading engine 42 shown in FIG. 4, and that in fact, the data-processing system 900 may function as the local processor 36, or, for example, the back office system 32 shown in FIG. 4 and further can communicate with, for example, the camera 44, the illuminator 46, the trigger 48, etc., depicted in FIG. 4.

[0087] FIG. 10 illustrates a computer software system 950, which may be employed for directing the operation of the data-processing system 900 depicted in FIG. 9. Software application 954, stored in main memory 902 and on mass storage 907, generally can include and/or can be associated with a kernel or operating system 951 and a shell or interface 953. One or more application programs, such as module(s) 952, may be “loaded” (i.e., transferred from mass storage 907 into the main memory 902) for execution by the data-processing system 900. In the example shown in FIG. 10, module 952 can be implemented as, for example, a module that performs the logical instructions or operations of methods 60, 80 respectively illustrated in FIGS. 6 and 8, or other appropriate methods/processes. Module 952 can in some embodiments be implemented as an ALPR module that uses augmented data as discussed herein.

[0088] The data-processing system 900 can receive user commands and data through user interface 953 accessible by a user 949. These inputs may then be acted upon by the data-processing system 900 in accordance with instructions

from operating system 951 and/or software application 954 and any software module(s) 952 thereof.

[0089] The discussion herein is thus intended to provide a brief, general description of suitable computing environments in which the system and method may be implemented. Although not required, the disclosed embodiments will be described in the general context of computer-executable instructions such as program modules being executed by a single computer. In most instances, a “module” constitutes a software application.

[0090] Generally, program modules (e.g., module 952) can include, but are not limited to, routines, subroutines, software applications, programs, objects, components, data structures, etc., that perform particular tasks or implement particular abstract data types and instructions. Moreover, those skilled in the art will appreciate that the disclosed method and system may be practiced with other computer system configurations such as, for example, hand-held devices, multi-processor systems, data networks, microprocessor-based or programmable consumer electronics, networked personal computers, mini-computers, mainframe computers, servers, and the like.

[0091] Note that the term module as utilized herein may refer to a collection of routines and data structures that perform a particular task or implements a particular abstract data type. Modules may be composed of two parts: an interface, which lists the constants, data types, variable, and routines that can be accessed by other modules or routines, and an implementation, which is typically private (accessible only to that module) and which includes source code that actually implements the routines in the module. The term module may also simply refer to an application, such as a computer program designed to assist in the performance of a specific task such as word processing, accounting, inventory management, etc.

[0092] The interface 953 (e.g., a graphical user interface) can serve to display results, whereupon a user may supply additional inputs or terminate a particular session. In some embodiments, operating system 951 and interface 953 can be implemented in the context of a “windows” system. It can be appreciated, of course, that other types of systems are possible. For example, rather than a traditional “windows” system, other operation systems such as, for example, a real time operating system (RTOS) more commonly employed in wireless systems may also be employed with respect to operating system 951 and interface 953. The software application 954 can include, for example, user proximity detection module 952, which can include instructions for carrying out steps or logical operations such as those shown in FIGS. 6 and 8.

[0093] FIGS. 9-10 are thus intended as examples and not as architectural limitations of disclosed embodiments. Additionally, such embodiments are not limited to any particular application or computing or data-processing environment. Instead, those skilled in the art will appreciate that the disclosed approach may be advantageously applied to a variety of systems and application software. Moreover, the disclosed embodiments can be embodied on a variety of different computing platforms including Macintosh, Unix, Linux, and the like.

[0094] Based on the foregoing, it can be appreciated that a number of embodiments are disclosed. For example, in one embodiment, a method for data augmentation utilized in an automatic license plate recognition engine can be implemented. Such a method can include, for example, the steps of logical operations of associating a machine-readable code

with an automatic license plate recognition engine; and configuring the machine-readable code to define parameters that drive processing within the automatic license plate recognition engine to produce recognition results thereof and enhance a machine readability of a license plate recognized and analyzed via the automatic license plate recognition engine.

[0095] In another embodiment, a step or logical operation can be provided for decoding the machine-readable code to determine the parameters. In yet another embodiment, steps or logical operations can be implemented for encoding the machine-readable code with information based on a hierarchy having hierarchy levels with levels that are more robust to image noises and lower levels that are less robust with respect to the image noises; and arranging the hierarchy such that the higher levels and the lower levels further define the parameters that drive the processing within the automatic license plate recognition engine to produce the recognition results thereof and enhance a machine readability of the license plate recognized and analyzed via the automatic license plate recognition engine.

[0096] In another embodiment, a system for data augmentation utilized in an automatic license plate recognition engine can be implemented. Such a system can include, for example, a processor, a data bus coupled to the processor, and a computer-usable medium embodying computer program code, the computer-usable medium being coupled to the data bus. Such computer program code can include instructions executable by the processor and configured, for example, for associating a machine-readable code with an automatic license plate recognition engine, and configuring the machine-readable code to define parameters that drive processing within the automatic license plate recognition engine to produce recognition results thereof and enhance a machine readability of a license plate recognized and analyzed via the automatic license plate recognition engine.

[0097] In some embodiments, such instructions can be further configured for decoding the machine-readable code to determine the parameters. In other embodiments, such instructions can be further configured for encoding the machine-readable code with information based on a hierarchy having hierarchy levels with levels that are more robust to image noises and lower levels that are less robust with respect to the image noises; and arranging the hierarchy such that the higher levels and the lower levels further define the parameters that drive the processing within the automatic license plate recognition engine to produce the recognition results thereof and enhance a machine readability of the license plate recognized and analyzed via the automatic license plate recognition engine.

[0098] In still another embodiment, a processor-readable medium storing computer code representing instructions to cause a process for data augmentation utilized in an automatic license plate recognition engine can be implemented. Such computer code can include computer code to, for example, associate a machine-readable code with an automatic license plate recognition engine, and configure the machine-readable code to define parameters that drive processing within the automatic license plate recognition engine to produce recognition results thereof and enhance a machine readability of a license plate recognized and analyzed via the automatic license plate recognition engine.

[0099] It will be appreciated that variations of the above-disclosed and other features and functions, or alternatives

thereof, may be desirably combined into many other different systems or applications. Also, that various presently unforeseen or unanticipated alternatives, modifications, variations or improvements therein may be subsequently made by those skilled in the art which are also intended to be encompassed by the following claims.

1. A method for data augmentation utilized in an automatic license plate recognition engine, said method comprising:

associating a machine-readable code with an automatic license plate recognition engine; and

configuring said machine-readable code to define parameters that drive processing within said automatic license plate recognition engine to produce recognition results thereof and enhance a machine readability of a license plate recognized and analyzed via said automatic license plate recognition engine.

2. The method of claim 1 further comprising decoding said machine-readable code to determine said parameters.

3. The method of claim 1 further comprising:

encoding said machine-readable code with information based on a hierarchy having hierarchy levels with levels that are more robust to image noises and lower levels that are less robust with respect to said image noises; and arranging said hierarchy such that said higher levels and said lower levels further define said parameters that drive said processing within said automatic license plate recognition engine to produce said recognition results thereof and enhance a machine readability of said license plate recognized and analyzed via said automatic license plate recognition engine.

4. The method of claim 1 wherein said parameters include data indicative of whether or not said license plate contains a potential for misreading due to confusing characters.

5. The method of claim 1 wherein said parameters include data indicative of whether or not said license plate comprises a standard plate or a non-standard plate.

6. The method of claim 1 wherein said parameters include data indicative of whether said license plate contains characters subject to plate frame occlusion.

7. The method of claim 1 wherein said parameters include data indicative of a state jurisdiction.

8. The method of claim 1 wherein said parameters include data indicative of whether or not said license plate possesses a logo.

9. The method of claim 1 wherein said parameters include data indicative of whether the license plate has stacked characters or not.

10. The method of claim 1 wherein said parameters include data indicative of a particular font utilized by said license plate.

11. The method of claim 1 wherein said parameters include data indicative of optimal RGB-to-Gray mapping.

12. The method of claim 1 wherein said parameters include data indicative of the number of characters on said license plate.

13. The method of claim 1 wherein said parameters include data indicative of whether each character on said license plate is a letter or a number.

14. A system for data augmentation utilized in an automatic license plate recognition engine, said system comprising:

a processor;

a data bus coupled to said processor; and

a computer-usable medium embodying computer program code, said computer-usable medium being coupled to

said data bus, said computer program code comprising instructions executable by said processor and configured for:

associating a machine-readable code with an automatic license plate recognition engine; and
 configuring said machine-readable code to define parameters that drive processing within said automatic license plate recognition engine to produce recognition results thereof and enhance a machine readability of a license plate recognized and analyzed via said automatic license plate recognition engine.

15. The system of claim **14** wherein said instructions are further configured for decoding said machine-readable code to determine said parameters.

16. The system of claim **14** wherein said instructions are further configured for encoding said machine-readable code with information based on a hierarchy having hierarchy levels with levels that are more robust to image noises and lower levels that are less robust with respect to said image noises; and

arranging said hierarchy such that said higher levels and said lower levels further define said parameters that drive said processing within said automatic license plate recognition engine to produce said recognition results thereof and enhance a machine readability of said license plate recognized and analyzed via said automatic license plate recognition engine.

17. The system of claim **14** wherein said parameters include data indicative of at least one of the following:

whether or not said license plate contains a potential for misreading due to confusing characters;

whether or not said license plate comprises a standard plate or a non-standard plate; or
 whether said license plate contains characters subject to plate frame occlusion.

18. The system of claim **14** wherein said parameters include data indicative of at least one of the following:

a state jurisdiction;
 optimal RGB-to-Gray mapping;
 the number of characters on said license plate;
 whether each character on said license plate is a letter or a number.

whether or not said license plate possesses a logo;
 whether said license plate includes stacked characters; and
 a particular font utilized by said license plate

19. A processor-readable medium storing computer code representing instructions to cause a process for data augmentation utilized in an automatic license plate recognition engine, said computer code comprising computer code to:

associate a machine-readable code with an automatic license plate recognition engine; and
 configure said machine-readable code to define parameters that drive processing within said automatic license plate recognition engine to produce recognition results thereof and enhance a machine readability of a license plate recognized and analyzed via said automatic license plate recognition engine.

20. The processor-readable medium of claim **19** wherein said computer code is further configured to decode said machine-readable code to determine said parameters.

* * * * *