

# Comparison of Real-Time DSP-Based Edge Detection Techniques for License Plate Detection

Zuvena Musoromy, Faycal Bensaali, Soodamani Ramalingam, and Georgios Pissanidis

Department of Engineering and Technology

University of Hertfordshire

Hatfield, UK

{z.musoromy3, f.bensaali, s.ramalingam, g.l.pissanidis}@herts.ac.uk

## Abstract

In this paper, edge detection techniques and their performance are compared when applied in license plate detection using an embedded digital signal processor. License plate detection remains to be the crucial part of a vehicle's license plate recognition process. The edge detection algorithms compared in this work are those reported capable of delivering real-time performance. These are Canny-Deriche-FGL, Haar and Daubechies-4 wavelet transform and the classic Sobel. These particular algorithms are chosen and compared due to their good performance on digital signal processors. The comparison is drawn in terms of speed and detection success of a license plate. The results show Haar wavelet-based edge detector performs better on a DSP with LP detection speed of 7.32 ms and 98.6% success using 45,032 UK images containing license plates at 768X288 resolutions.

**Keywords:** Digital signal processor (DSP), Real-time, Edge detection, License plate (LP)

## I. INTRODUCTION

The traditional automatic number plate recognition (ANPR) systems involving a camera capturing vehicles and a separate nearby computer to process the images would benefit immensely in terms of cost by having an embedded standalone system capable of capturing and processing a license plate (LP) on board. Powerful processors such as Digital Signal Processors (DSPs) and Field Programmable Gate Arrays (FPGAs) are normally used for this task. An example of such system is presented in [1]. The ANPR as part of Intelligent Transportation Systems (ITSs) has an important role in recognizing license plates, which are used to identify vehicles for security reasons.

In most systems, ANPR have three parts that involve license plate detection, character segmentation and character recognition. In LP detection, the region in an image that contains the LP is extracted. There are a number of techniques reported in the literature to detect an LP [2].

The edge detection method, which is simply looking at object boundaries in an image, remains the most popular and efficient way to initially determine the features of an LP. This step is normally followed by joining the edges and comparing the resulted feature to that of an LP. The edge detection is also very useful in enhancing image features.

This work is comparing and deducing the best edge detection algorithm for LP detection performing in real-time at 25 frames per second (under 40 milliseconds) using a DSP. This requirement reflects busy roads traffic scenarios such as the highway where a vehicle is travelling up to 80 miles per hour. A real-time literature review on edge detection and LP algorithms using DSPs is also presented and compared to this work.

DSPs are widely used in applications such as audio and speech processing, image and video processing, and wireless signal processing. In this paper, the investigation is focusing on image processing using DSPs in particular area of edge detection and LP detection.

Previous use of DSPs in this area is reported in applications of video encoding and decoding, surveillance, and image and video objects tracking and detection. Most of these applications use DSPs due to the offered processing power for real-time capability and portability to create custom embedded architectures depending on application's needs. In addition to DSPs, FPGAs have been used in LP detection and recognition. A real-time LP recognition using FPGAs are reported by [3] and [4]. In [3] the fastest time for LP detection and recognition in literature is reported at 9.25 ms with recognition rate of 87%. The shortcoming of the methods is the low recognition rate. However, the authors have demonstrated the power of the FPGAs.

The organization of this paper is as follows. The next section reviews previous work on real-time edge detection on DSP. Section (3) discusses DSP algorithms optimization. Section (4) presents the LP detection algorithm. Section (5) presents experimental results of edge detection and LP detection obtained and finally Section (6) gives discussion and conclusion.

## II. RELATED WORK

A review of edge detection algorithms as well as the LP detection algorithms capable of achieving real-time in DSPs is discussed in this section. A real-time Canny-Deriché-FGL optimal edge detector on DSPs and RISC is reported by [5]. This edge detector originated from Canny [6] and is regarded as the optimal edge detector where Deriché [7] improved the performance index of Canny by 25% followed by optimization to lower computational complexity of Canny-Deriché by García-Lorca (FGL) in [8].

A DSP-based edge detection comparison is explained in [9]. Three edge detection algorithms performance on DSP are compared using Canny, Prewitt and Haar wavelet-based. The reported outcome is that the Haar wavelet-based edge detector performed best in terms of SNR in noisy images. The authors recommended post-processing of the output edges to make them more optimal.

In [1], an embedded DSP-based system for surveillance purposes is presented. The system is also used for LP recognition task. The hardware main components are one DSP for image processing and one FPGA for transferring video between the video input processor and the DSP. This work demonstrates that DSP processors can be expanded to work with other processors for more complex applications. The AdaBoost algorithm by Viola and Jones is used to detect and recognise LP. The LP recognition success was 96% running at 141.62 ms.

An extension work of [1] using the same DSP system is reported in [10] for LP recognition in real-time. The LP detection algorithm used is combination of Kalman tracker and RealBoost based on AdaBoost by Viola and Jones. This work produced the fastest time in the DSP literature for LP detection and recognition. A summary of the DSP literature review is shown in table I.

The results in table I and conclusions by the authors indicate that the processor's clock speed, the size of the image processed and choice of algorithms are the contributing factors in achieving real-time performance.

## III. DSP ALGORITHMS OPTIMIZATION TECHNIQUES

The DSP algorithms need to be highly optimized in order to meet the real-time requirement. In this section, the standard DSP-based optimization techniques to accelerate the algorithm performance are presented.

The first method is DSP memory optimization explained in [11] and [12]. The DSP used here has two-level (L1 and L2) Cache-based memory architecture [11]. L1 has a program memory section known as L1P and a data memory section known as L1D while L2 has memory section for both program and data. Two types of Cache optimization techniques explained in [11] are applied to the algorithms using Code Composer Studio (CCS) version 3.3. These are:

- Data Cache Optimization, which includes application level, procedure level and exploiting types of Cache access whether it misses or hits and read or write.
- Program Optimization, which explores the program Cache by analyzing and eliminating conflict misses.

The second method is the linear assembly optimization. This is the interface between programming languages such as C and assembly programming language. In linear assembly, code are automatically converted to assembly using the compiling optimizer.

The third method is optimizing C source code. The following is a list of the techniques used:

- Inline function (intrinsic)
- Wide memory data access and data packet
- Software pipelining
- Optimization of function implementation

TABLE I. DSP-BASED EDGE DETECTION AND LP DETECTION LITERATURE REVIEW

Ref.	Year	DSP processor name	Processor Clock speed (MHz)	Edge detection algorithm	Image size (pixels)	Edge detection time (ms)	LP detection algorithm	LP detection time (ms)	LP detection success (%)
[5]	1998	C80	60	Canny-Deriché-FGL	512x512	20	*Non	*Non	*Non
[9]	2005	C6711	100	Prewitt, Canny and Haar Wavelet	640x480	20, 1200 and 790	*Non	*Non	*Non
[5]	1998	C62	200	Canny-Deriché-FGL	512x512	13	*Non	*Non	*Non
[1]	2006	C6414	600	*Non	352x288	*Non	Adaboost	141.62	96
[10]	2007	C6414	600	*Non	352x288	*Non	Real boost	< 50	*Non
**	2010	C64plus	600	Sobel	768x288	1.22	CCA	4.86	96.1
**	2010	C64plus	600	Daubechies-4 wavelet	768x288	2.63	CCA	7.72	98.1
**	2010	C64plus	600	Haar (Daubechies-1) wavelet	768x288	2.55	CCA	7.32	98.6
**	2010	C64plus	600	Canny-Deriché-FGL	768x288	24.27	CCA	27.84	98.5

\* Indicate that the information is not available, \*\* indicate the results in this paper, CCA stands for Connected Component Analysis

#### IV. LICENSE PLATE DETECTION ALGORITHM

The algorithm proposed in this paper is Haar wavelet based edge detector, which is used to find LP region. The literature has shown that optimal edge detectors such as Canny would require a powerful processor to reach real-time performance of less than 40 ms. It is reported in [5] that an alternative optimized version of Canny (Canny-Deriche-FGL) is capable of producing good edges and delivering real-time performance using DSP.

In addition to Haar wavelet, Canny-Deriche-FGL, Daubechies-4 wavelet and Sobel edge detectors are also compared in the application of LP detection. Canny-Deriche-FGL and Daubechies-4 edge detectors are reported for the first time in LP detection in this paper. An example of Sobel edge detector for LP detection is reported in [13]. A Pentium III PC was used for image processing where the authors did not report the LP detection time. The LP detection success reported is 96.22%.

The following is a list of algorithm's steps which are shown in Fig.1:

- Normalization of an input gray scale image
- Sharpening filter
- Edge detection
- Post-processing
- Adding edges back to an input gray scale image
- Connected Component Analysis (CCA) on gray scale image
- Return best LP candidate

The first step is normalizing the images. The fact that images are captured at different lighting conditions, the testing database used in this work contain images with flat (weak) edges and sharp edges. Image normalization is performed by setting a threshold value based on the difference between the maximum and minimum intensity levels.

It is observed that some LP were not detected due to the images containing flat edges. The edge enhancement is introduced as a second step. In this process, flat edges are benefited by enhancement using a standard 3x3 sharpening kernel.

The third step is edge detection using Canny-Deriche-FGL, Daubechies-4 wavelet [14], Haar wavelet [14] and the classic Sobel edge detector. The standard Sobel operator (3x3) mask is used for edge detection. Canny-Deriche-FGL edge detector similar to [5] is also investigated.

The Haar wavelet transform to detect LP is reported by [15], where a reference line is drawn in a horizontal edge image above the LP followed by searching the exact location of the plate. The method struggles when the LP is too close or too far away from the camera as reported by the authors with 92.4% success.

By definition, Haar transform is the first and simplest type of wavelet. It is discontinuous with a property of a step function and is the same wavelet as Daubechies order one (orthonormal wavelet). In addition, Daubechies-4 is that of order four where the algorithm principle remains the same compared to that of Haar apart from the transform order.

In this paper, a 1D wavelet transform kernel is used, which also perform the decomposition of 2D wavelet transform. More details on the theory of wavelet transform can be found in [9, 14, 15]. The application of this kernel to the rows (r) and columns (c) of an input image result in low and high frequency data. The output images  $E_{(r,c)}$  due to one scale transform contain horizontal ( $E_{(h)}$ ) and vertical edges ( $E_{(v)}$ ). The interest in this paper is on the absolute sum of edges, which is computed to produced the resultant edge image given as,

$$|E_{(r,c)}| = |E_{(h)}| + |E_{(v)}|. \quad (1)$$

The resultant edges are not well visible due to the effect of wavelet transform suppressing image pixels intensity. Post-processing is introduced at this stage to enhance the edges. The contrast of the image pixels is stretched and a variable threshold value is set to reduce the noises and make the edges more stronger.

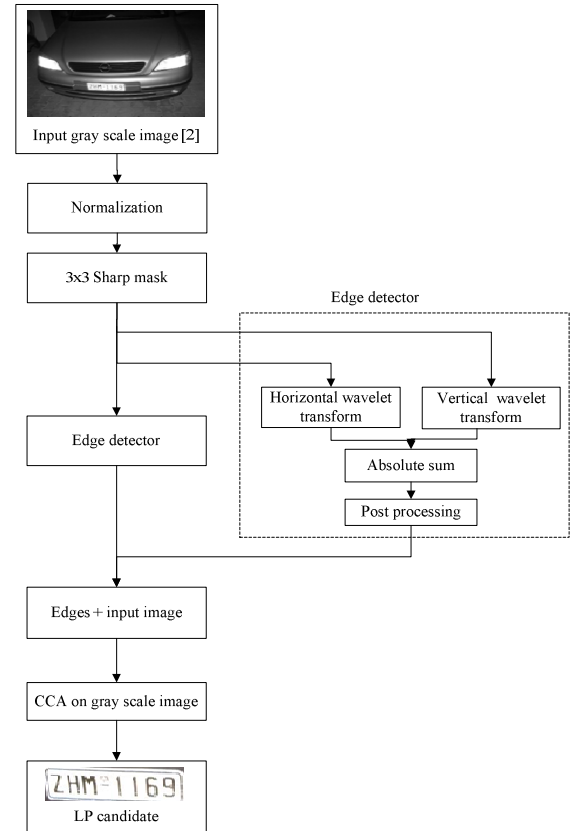


Figure 1. License Plate Detection Algorithm

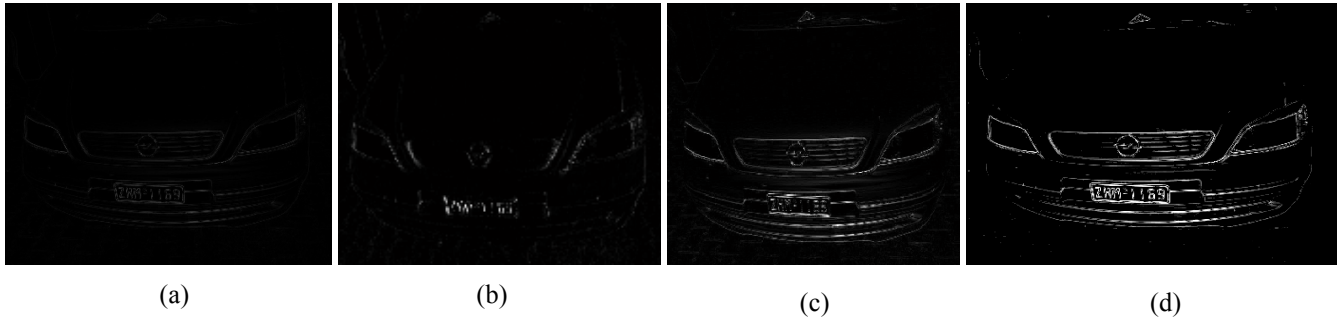


Figure 2. Haar edges resulted from input image in Fig.1. (a) Absolute edges before post-processing (b) Enhanced vertical edges (c) Enhanced horizontal edges (d) Enhanced absolute edges

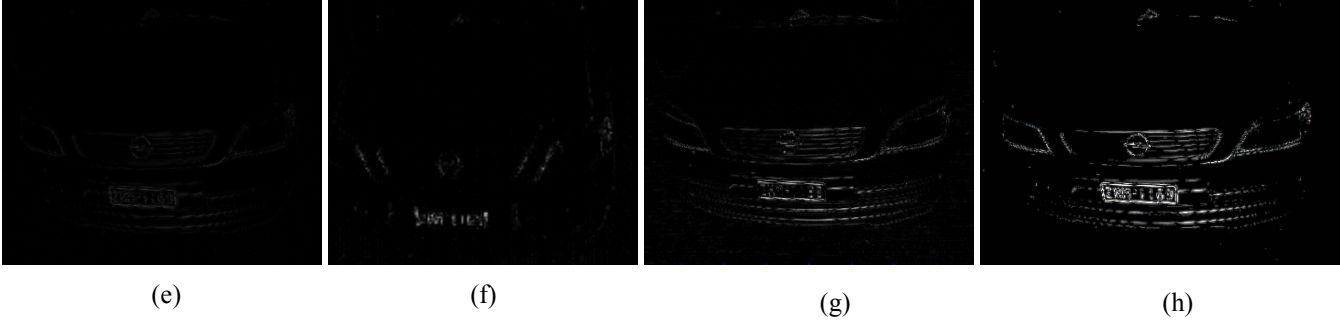


Figure 3. The Daubechies-4 edges resulted from input image in Fig.1. (e) Absolute edges before post-processing (f) Enhanced vertical edges (g) Enhanced horizontal edges (h) Enhanced absolute edges

The next step is to add the edges and the threshold set in the previous step back to the original gray scale image and then apply CCA widely used in image processing to scan image pixels and labels them to identify connected pixel regions that share similar pixel intensity value. Haar Wavelet (Daubechie-1) edge detector shown in Fig.2 produced good (thinner) edges compared to Daubechie-4 shown in Fig.3, which makes them easier to connect when using CCA. The good edges produce rectangles that can easily be distinguished when detecting LP. This gives less false identification of plates. The final stage is to filter the rectangles by comparing them to match the defined characteristics (plate resolution, pixels and number of edges) of an LP and return the best candidate.

## V. DSP IMPLEMENTATION AND RESULTS

The proposed algorithms are optimized using the techniques listed in section 3. The implementation of the algorithm in DSP requires the following tools;

- A Windows host PC with Code Composer Studio and a monitor
- A Texas Instrument's C64plus DSP (fixed-point DSP based on an enhanced version of the second generation high-performance, advanced Very-Long-Instruction-Word (VLIW)) with minimum of 600MHz clock speed with of 1MB of RAM

- DSP host board (The exact details of the DSP board used is beyond the scope of this paper)
- JTAG interface debugger to provide interface between the DSP and the host PC during debugging DSP algorithm
- Testing database of 45,032 images of 768X288 resolutions provided by CitySync Ltd.

The results in table II show the time taken to execute a single image using Canny-Deriche-FGL, Daubechies-4 wavelet, Haar wavelet and Sobel edge detectors. The individual time for each algorithm achieved is less than 40 ms (real-time). However, the average time presented in table II when a video stream (real ANPR setting) is used may be affected at least twice as much. This is because firstly, the algorithm would process every frame given even the ones that do not contain LP. This effect can be reduced by filtering these images before they are passed to the algorithm for processing.

## VI. DISCUSSION AND CONCLUSION

It is noted that the introduction of sharpening kernel increases the LP detection rate in Sobel, Haar and Daubechies-4 wavelet edge detectors by 1.7% of which 0.5 are false positives. It is also observed that the false positives increases when a sharpening kernel is used. To minimise this effect, the unwanted edges formed after edge detection are removed using a threshold before they are passed to CCA. Further reduction of false positives is applied when the unwanted rectangles formed after applying CCA on the edges are filtered out using a defined rectangle resolution threshold.

TABLE II. DSP-BASED EDGE AND LP DETECTION PERFORMANCE

Algorithm	DSP Edge time (ms)	DSP LP detection time (ms)	LP Detection success using 45,035 images (%)	Sharpening images LP time (ms)	Sharpening LP using 45,035 images detection rate (%)
Sobel	1.22	3.92	94.9	4.86	96.1
Daubechies-4 wavelet	2.63	6.61	97.2	7.72	98.1
Haar (Daubechies-1) wavelet	2.55	6.26	97.4	<b>*7.32</b>	<b>*98.6</b>
Canny-Derliche-FGL	24.27	26.91	98.2	27.84	98.5

\* Indicates the fastest time and the recognition success achieved using Haar wavelet

Initially, Canny-Derliche-FGL edge detector produced high LP detection success. When a sharpening filter is introduced, the Haar wavelet-based edge detector produced the fastest time and higher LP detection rate. This is because Canny-Derliche-FGL is already a good edge detector so sharpening the image has very small effect on its overall performance.

The application of sharpening kernel has small improvement on Canny-Derliche-FGL LP detection rate compared to Sobel, Haar and Daubechies-4 wavelet where the improvement is observed to be significant. The performance of Canny-Derliche-FGL edge detector was slow compared to the rest of the algorithms. This is because, the algorithm was not fully optimised as suggested by [5]. Therefore further optimization of this algorithm using methods such as Enhanced Direct Memory Access (EDMA) during calculations and data transfer would produce faster times.

The LP detection rate depends on the quality of an input image. As a result, it is very difficult to find an algorithm with 100% LP detection rate. The quality of an image captured can be improved at higher resolution, therefore, the future work will focus on images at higher resolution.

The overall results have shown that it is possible to use DSP processors to accelerate LP detection process and achieve the real-time performance at higher LP detection rate using edge detection methods. The increase in DSP's processing power creates an opportunity to produce better algorithms that require a real-time performance. The advantage of real-time performance is that an embedded standalone automatic number plate recognition system can be created for automotive security applications such as traffic monitoring and access control, which are very useful in law enforcement.

#### ACKNOWLEDGMENT

This work is funded by Engineering and Physical Sciences Research Council (EPSRC) and CitySync Ltd. Many thanks to Carol McStravick, Nico Bekooy, Prof. Aladdin Ariyaenia, Prof Reza Sotudeh and Bob Andrews for their support on this work.

#### REFERENCES

- [1] C. Arth, C. Leistner, and H. Bischof. TRICam: An Embedded Platform for Remote Traffic Surveillance. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (Embedded Computer Vision Workshop), 2006.
- [2] C. E. Anagnostopoulos, I. E. Anagnostopoulos, I. D. Psoroulas, V. Loumos, and E. Kayafas, "License plate recognition from still images and video sequences: Asurvey," IEEE transactions on Intelligent Transportation System, vol. 9, no. 3, pp. 377–391, 2008.
- [3] Kanamori, T., Amano, H., Arai, M., Konno, D., Nanba, T., and Ajioka, Y. 2007, "Implementation and Evaluation of a High Speed License Plate Recognition System on an FPGA". In Proceedings of the 7th IEEE international Conference on Computer and information Technology (October 16 - 19, 2007). CIT. IEEE Computer Society, Washington, DC, 567-572.
- [4] N. Bellas, S. M. Chai, M. Dwyer, D. Linzmeier, "FPGA implementation of a licence plate recognition SoC using automatically generated streaming accelerators," Proc. of the IPDPS, April, 2006 (ISBN: 1-4244-0054-6).
- [5] L. Lacassagne, F. Lohier. Real time execution of optimal edge detectors on RISC and DSP processors, 23rd International conference on Acoustics, Speech and Signal Processing, 1998.
- [6] Canny, J. F., A computational approach to edge detection. IEEE Transactions. On Pattern Analysis and Machine Intelligence, 8, 1986, pp. 679-714.
- [7] R. Deriche. Fast Algorithms for low level-vision, IEEE Trans. on PAMI, vol 12, 1 (1990).
- [8] F. Garcia-Lorca. Filtres recursifs temps reel pour la detection de contours : optimisations algorithmiques et architecturales. These Universite d'Orsay (1996).
- [9] Abdel-Qader, I. M. and Maddix, M. E. 2005. Edge detection: wavelets versus conventional methods on DSP processors. MG&V 14, 1 (Jan. 2005), 83-101.

- [10] C. Arth, F. Limberger, and H. Bischof, "Real-time license plate recognition on an embedded DSP-platform," in Proc. IEEE Conf. CVPR, Jun. 17–22, 2007, pp. 1–8.
- [11] Texas Instruments, "TMS320C64x+ DSP Cache User's Guide", Literature number: spru862a, October 2006.
- [12] C. Peng et al. "The Optimization of H.264 Encoder Based on TI TMS320DM642", Future Generation Communication and Networking (FGCN 2007), Vol. 1, pp. 38-42, December 2007.
- [13] Muhammad Sarfraz, Mohammed Jameel Ahmed, Syed A. Ghazi, "Saudi Arabian License Plate Recognition System," gmag, pp.36, 2003 International Conference on Geometric Modeling and Graphics (GMAG'03), 2003.
- [14] S G Mallat, A Wavelet Tour of Signal Processing, Academic Press, 1998.
- [15] C.-T. Hsieh, Y.-S. Juan, and K.-M. Hung, "Multiple license plate detection for complex background," in Proc. Int Conf. AINA, 2005, vol. 2, pp. 389–392.