Implementation of Smart Parking System Using Image Processing

P.M.D.S. Amarasooriya[1], M.P.P.L. Peiris[1], H.M.D.S. Herath[2]

[1]Faculty of science, University of Kelaniya, Sri Lanka
[2]School of Computing, University of Colombo, Sri Lanka

ABSTRACT

In recent years, the number of vehicles in use has shown a steady increase, leading to a clear demand for larger parking areas. However, the traditional methods for detecting occupancy of slots in smart vehicle parking areas are no longer feasible due to the high cost of sensors and the need to monitor larger areas. In response to this challenge, the present study aims to propose a cost-effective, fast, and accurate solution for updating and indicating the real-time number of free parking slots in a parking area. Specifically, the proposed solution utilizes video footage from a camera as the input device and applies the YOLO v3 object detection algorithm for image processing to detect the coordinates of both parking lots and parked vehicles separately. To train and evaluate the model, we used the PKLot database as the dataset and tested the model's performance under different weather conditions. The proposed model achieved an average performance of 88.01%, with the highest performance demonstrated on sunny days and the lowest performance recorded on rainy days.

KEYWORDS: *Convolutional Neural Network, image processing, parking space detection, shortest path algorithm, smart parking system.*

1 INTRODUCTION

With the ever-increasing urban population and the improvement in living standards, the usage of vehicles has increased. According to the past statistics of the Department of motor traffic Sri Lanka usage of vehicles is continuously increasing from 2008 to 2016 according to the department of motor traffic in Sri Lanka. Traffic congestion and difficulty in finding free parking areas are some of the main concerns in metropolitan areas. As a result, parking areas are expanding and the need for monitoring large vehicle parking areas is becoming a necessity.

https://doi.org/10.54389/TMIM3086

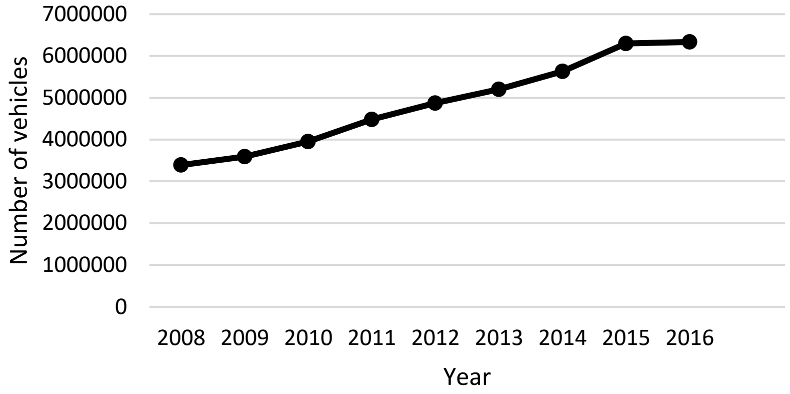


Figure 1: Total vehicle population in Sri Lanka

Nowadays most vehicle parking areas are not running efficiently. Existing parking areas are not efficient and are not meeting the driver's expectations. Because of that drivers may take extra time driving around the parking area on busy days to find a free parking space. Even though the number of parking areas is high, drivers cannot find an available parking space due to the unavailability of the proper status of parking area. Failure to find a free parking slot can cause congestion on the roads, Carbon Dioxide emission, waste of energy, road accidents and an increase in the stress level of drivers. The main problem that always occurs at the car park is wasting time when searching for available parking spaces. These problems are seriously affecting public health and the effective use of resources. In most urban cities, the average search time for an available parking place is around 10 minutes. (Edirisinghe, 2014).

To tackle these problems, smart parking systems have been implemented. The occupancy of the vehicles is detected using sensors. Geomagnetic, ultrasonic, Radio-Frequency Identification (RFID), Internet of Things (IoT) (Aekarat Saeliw, 2019) and magnetic sensors (J. Wolff, 2006) were some of the object detection methods used by past studies. The Canny Edge Detection algorithm is one of the widely used algorithms in the study area (Benjamin Kommey, 2018). High cost, low performance in poor weather conditions, less accuracy and limitation of small parking areas were some of the issues in existing system solutions. There is a dearth of research on solutions which are fast, cost-effective, and suitable for monitoring large area. The study addressed the research question of What is the best solution for overcome in effectivity existing issues of the vehicle parking system using image processing? The aim of the study is to identify problems with existing smart parking systems and introduce an improved and effective method to overcome the issues of the vehicle parking system.

2 LITERATURE REVIEW

Various studies have been carried out related to smart parking areas. Different categories of smart parking areas could be identified based on the technological perspective (Barriga,2019). Smart parking solutions involve several technological components such as sensors, software solutions, and networking infrastructure. Cameras (Lamborinos L. & Dosis A., 2013), ultrasonic sensors (Greg Y. & Cassandras C.G, 2012), cellular sensors (John et.al, 2017), infrared (Vora et al., 2014), radar (Mathews et al, 2014), RFID (Aekarat Saeliw, 2019), laser-based, smartphone sensors and magnetometers (Jones et al, 2017) have been used in past studies as sensors. Sensors were divided into two categories based on the established location. Different studies did the selection of sensors based on criteria invasive, ease of installation, one sensor per slot, several sensors per slot and detection anatomy.

There are various methods to detect vehicles in a parking area. One method is using the Radio-Frequency Identification (RFID) and the Internet of Things (IOT) which can detect the available parking lot. (Aekarat Saeliw, 2019) This IOT based parking system is created by using controllers, sensors, servers, and cloud. Sensors are placed in each parking slot to detect the presence of a car. The sensor will read the number of vacant parking lots and send the data to the microcontroller. Then the microcontroller will display the number of vacant parking spots and display it in the LED display. Therefore, drivers can know the availability of free parking slots before entering the parking area. Figure 2 shows the process of the above-mentioned system.

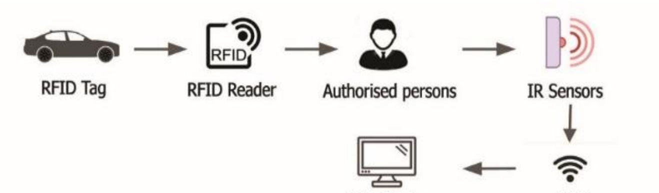


Figure 2. The architecture of Detecting Empty Parking Slot

The problem of this system is the inability of sensors to detect the presence of a vehicle in the bad weather condition. Each slot must have a sensor and having sensors in every parking slot is not cost-effective. Also, it is difficult to maintain a sensor-based parking system. There is another method to determine the vacancy status of parking space, through magnetic sensors by using the effect of Earth's magnetic field. (J. Wolff, 2006)

Another way is to identify the parking slot using image processing technology. (Benjamin Kommey, 2018) This system consists of some operations such as system initialization, image acquisition, and image processing, data interpretation, and display updates. Cameras take snapshots and pass them to the microcontroller unit (MCU) for communication at regular time intervals. The controller forwards the collected data to the base station over the setup local area network. At the base station, the received images undergo a preprocessing stage. The system can display the result using the method of Luminosity, Canny edge detection, and other techniques. The accuracy of this system depends on information captured from images of the parking space. Most of the research used canny edge detection algorithm for identifying the free parking slots. (Benjamin Kommey, 2018)

3 RESEARCH METHOD

The overall conceptualized solution is shown as follows. When a vehicle is entering to the parking area, it shows the status of the parking area using image processing and object detection algorithms (YOLO v3) by evaluating the footage of the surveillance camera established in the parking area. If there are no free parking slots, the system displays the message "parking area is full". Otherwise, the system shows several free available parking slots and directs the driver to the nearest parking area. After the driver parks the vehicle in the free parking slot the system updates the status of each parking slot and displays the result at the vehicle park entrance.

Checking the availability of parking slots could be further divided into the following steps as system initialization, input Livestream video, get coordinates of the parking area, assigning unique numeric labels to each parking slot, get image frame, identifying the vehicles using object detection, check availability of parking lot and display output.

The video camera is fixed in a position above the vehicles, to acquire the video used to get the vacancy of the car park. An inexpensive compact wireless video camera can be used for this purpose. This camera should be in a position where can see all parking lots and the camera should be fixed more than ten feet above the ground level because the mean pixel value of each vehicle in the parking area must be always captured.

3.1 YOLO Algorithm

YOLO (You Only Look Once) is a state-of-the-art, real-time object detection system. It uses Convolutional Neural Networks (CNN) for object detection. CNN was introduced in the 1990s (Lecun et al,1998) and it could be categorized under deep neural network especially invented for image processing and object detection.

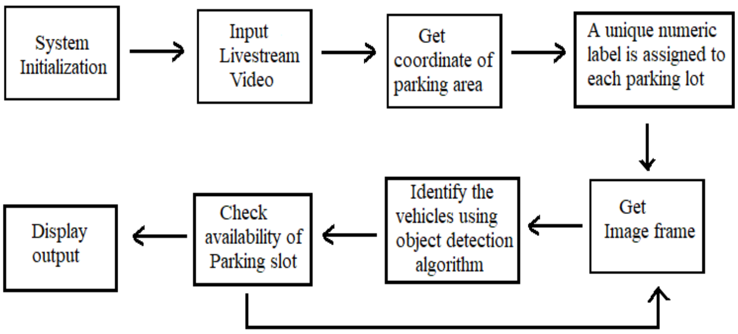


Figure 3. Flowchart of the free parking slot detecting system

It processes data in a grid-like manner and consists of an input layer, a series of hidden layers and an output layer. YOLO name is self-explanatory that it passes the input image through the CNN algorithm only once to get output.

YOLO has versions as YOLO v1, YOLO v2, and YOLO v3. YOLO v3 can detect multiple objects present in an image in real-time by drawing bounding boxes around them, which is the location and predict the class of the object. YOLO v3 was selected for the study because it was more accurate and faster compared to other versions.

3.1.1 Process of object detection in YOLO v3

YOLO v3 applies a single neuro network for the entire image. The image was divided into grid cells and produced probabilities for every cell. Then predict several bounding boxes that cover some area of the image and choose a bounding box according to the probability.

3.1.2 Network Inputs

Network input consists of a batch of images. The shape of the image is represented using four numbers as (n, 416, 416, 3). The first, second, third and fourth numbers represent the number of images, width, height, and number of channels respectively. Height and width can be interchanged and divisible by 32. Increasing input resolution might improve the accuracy of the output after the training. Middle 2 numbers are considered as the input network size. YOLO v3 accept images of any size to the network as all the images will be resized to network size.

3.1.3 Architecture of YOLO v3

YOLO uses convolutional layers. YOLO v3 consists of 53 CNN layers (Darknet -53) and staked additional 53 layers for detection tasks. Detections are made in 82,94 and 106 layers. Each CNN layer is followed by a batch normalization layer and Leaky ReLU. In addition to that YOLO v3 consists of essential elements residual blocks, skip connections and up-sampling. There are no pooling layers in YOLO v3 but convolutional strid 2 layers are present. It prevents losing small features as it considers all numbers in the feature map.

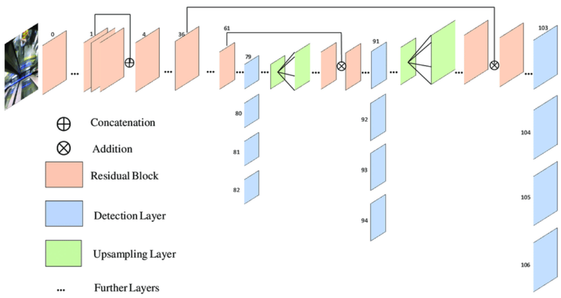


Figure 4. YOLO v3 NNC Architecture

The network makes detections at layer 82,94 and 106 by down-sampling input by following 32,16 and 8 factors respectively. These numbers are known as Network Strides and show the output in 3 separate places in the network. If the network input size is 416 x 416, layer 82, 94 and 106 output sizes are respectively 13 x 13,26 x 26 and 52 x 52. 13 x 13 detects large objects while 52x 52 detects smaller objects.

YOLO v3 applies 1 x1 detecting kernels (filters) to produce the feature maps. 1 x 1 kernel's shape includes the depth which can be calculated by the following equation.

$$\text{Depth of kernel(D)} = (b*5+c) \qquad (1)$$

Where b is the number of bounding boxes. YOLO v3 predicts 3 bounding boxes for each cell of the feature map. Each bounding box consists of 5 attributes. Centre X and Y coordinated of the binding box (t_x and t_y), width and height of the bounding box (t_w and t_h) and objectness score of the bounding box (P_0) are five attributes. C is the list of confidence for every class (Lorry, car, van,…) this particular bounding box might belong to (P_1, P_2,P_3,…,P_C). The shapes of the feature map produced in layer 89, 94, 106 are (13,13, D), (26,26,D), (52,52,D). (Kathuria, 2018) Objectness score is the probability that the bounding box contains an object inside.

The network predicted 3 bounding boxes for each cell in the feature map. Each cell predicted an object through its one bounding box if the center of the object belongs to the center cell of the bounding box. This was done during the model training.

During the training, YOLO v3 has one ground truth binding box for one object. The Centre cell of the bounding box was assigned to predict the object. The objectless score for this was 1.

To predict bounding boxes, YOLO v3 used predefined /default bounded boxes called Anchor boxes or priors. Anchors are used to calculating the predicted bounding box's real width and height. In YOLO v3 three anchor boxes were used for each scale (82, 94, 106) to calculate 3 bounding boxes for each scale. K-means clustering was applied to calculate anchors in YOLO v3. Anchors do not predict the definite height and width to eliminate the unstable gradient. Therefore c_x,c_y,p_w and p_h made normalize c_x,p_w by dividing the width of the image and c_y,p_h by the height of the image.

To predict the correct height and width of predicted binding boxes YOLO v3 calculated the offset of the anchor. It is also known as Log space transform. To calculate the center of the binding boxes, YOLO3 version passed parameters through the sigmoid function. Below mentioned equations are used to obtain the height and width of the center coordinates. b_x, b_y, b_w, b_h are the center, width, height of coordinates and t_x, t_y, t_w, t_h are outputs of CNN. c_x and c_y are the top left corner coordinates of anchor box p_w and p_h are the anchor's width and height.

$$b_x = \sigma(t_x)+c_x \qquad (2)$$

$$b_y = \sigma(t_y)+c_y \qquad (3)$$

$$b_w = p_w e^{tw} \qquad (4)$$

$$b_x = P_h e^{th} \tag{5}$$

YOLO v3 center coordinates were passed using the sigmoid function by giving values 0 and 1. Using the below-mentioned equations absolute values of the binding box were calculated.

$$BB_X = b_x * \text{width of image} \tag{6}$$

$$BB_y = b_y * \text{ height of image} \tag{7}$$

$$BB_w = b_w * \text{width of image} \tag{8}$$

$$BB_h = b_h * \text{height of image} \tag{9}$$

To select one bounding box, extract probabilities of bounding boxes to find out whether the bounding box contains a certain class. For that computed the product of P_0 and list of confidences which was $\{P_1, P_2, P_3, \ldots, P_C\}$. The binding box with maximum probability was the relevant class of the object.

The equation for objectness score can be represented as follows where is the predicted probability binding box and between the predicted and ground truth binding box.

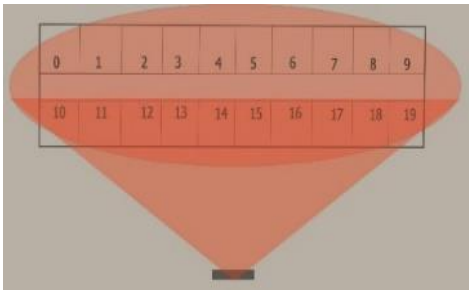
$$P_{object} * IoU = \sigma(to) = Po \tag{10}$$

To find the mean pixel value of each vehicle takes the fixed numbers of pixel values inside the vehicle bounding box. Here n is the fixed number of points.

$$\text{Mean of x values } \bar{x} = \frac{1}{n}\sum x \tag{11}$$

$$\text{Mean of x values } \bar{y} = \frac{1}{n}\sum y \tag{12}$$

After segmenting the video into frames, the system use frame of the empty parking area to record the location of all parking slots. According to that frame, the system inputs the coordinates of each parking slot to a text file and labels each parking slot using a unique numeric value.



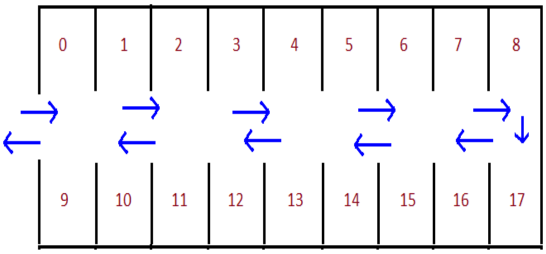


Figure 5. The view of camera position and the architecture of parking area

The maximum number of slots that could be labelled depends on the capture area of the camera and the training model. For the vehicle detection and occupancy of the parking lots, the YOLO v3 object detection algorithm was used. Using the non-maximum suppression technique determines the class of each object. After getting the coordinate of different types of vehicles marked the bounding boxes. The status of each parking slot can be determined by comparing the mean pixel value of each vehicles and the coordinates of bounding boxes. If the mean pixel value of the vehicle is inside the given bounding box then, that parking slot is not empty. Otherwise, it is empty. Sequentially update the status of each parking slot and display it. The vacant parking slot is displayed according to the distance from the entrance to the parking slot.

4 The PKLot dataset

The PKLot database was used as the dataset for this study. The PKLot dataset includes 12,417 images of parking lots and 695,899 images of segmented parking spaces, checked and numbered manually. Both photographs were bought at the Federal University of Parana and the Pontifical Catholic University of Parana parking lots in Curitiba, Brazil. The dataset for parking slots is acquired by following the image acquisition process. This process was executed with a 5-min time-lapse interval for more than 30 days by means of a low cost full high definition camera (Microsoft LifeCam, HD-5000) positioned at the top of a building to minimize the possible occlusion between adjacent vehicles. Here is a short summary of this dataset.

- Photos are taken under unregulated lighting that depicted various climatic conditions (sunny, rainy, and overcast periods).
- Images were taken from different parking lots presenting distinct features.
- Images show a variety of issues, such as shadow effect, over-exposure to sunlight, low light in rainy days, disparity in perspective, etc.

Extensible Markup Language (XML) file containing the position and location (vacant or occupied) of each parking space was created for each parking lot image. (Jiwoong Choi, 2019)

Figure 6: Image of PKLot dataset without boundary boxes

Each image of the database has an XML file associated with the coordinates of all the parking spaces. Example for XML file as follows.

```xml
<space id="1" occupied="0">
  <rotatedRect>
    <center x="300" y="207" />
    <size w="55" h="32" />
    <angle d="-74" />
  </rotatedRect>
  <contour>
    <point x="278" y="230" />
    <point x="290" y="186" />
    <point x="324" y="185" />
    <point x="308" y="230" />
  </contour>
</space>
```

Findings and Discussion

The following figure shows the marked boundary boxes and the unique id of each parking slot.

Figure 7. Image of PKLot dataset with unique ID

The online system is getting images from the camera while the offline system is getting images from a video file. The performance of the proposed system is measured by using the following equation. (Almeida P., n.d.)

TPS = Total Parking slots, ANC = Actual Number of vehicles, PNC = Predicted Number of vehicles (detected vehicles)

$$\text{Performance}= 1- ((|\ ANC - PNC\ |)/TPS) *100 \tag{11}$$

The percentage of error in the proposed system can be found by using this equation.

$$\text{Percentage Error} = ((|\ ANC - PNC\ |)/TPS) *100 \tag{12}$$

For getting the performance of the algorithm, used different images in different weather conditions. It is observed that the average performance is 88.01 %. The accuracy of the proposed system also depends on the type of camera used for monitoring the parking lot. The following table shows the performance of the algorithm in different weather conditions.

Table 1: Accuracy of algorithm

	Total Number of slots	**Actual Number Of vehicles**	**Predicted Number of vehicles**	**Performance**
Sunny Day	80	78	71	91.25%
Cloudy Day	70	68	60	88.5 %
Rainy Day	70	33	22	84.28%

Conclusion

In the current project, the use of a camera as a sensor for video image detection is proposed. This approach offers the advantage of detecting a large parking area at once and detecting the park lots in any weather condition, while being efficient and cost-effective. The number of cameras required depends on the area to be covered. Unlike existing automated parking systems that rely heavily on various sensors, the proposed system uses image processing algorithms to automate parking with footage obtained from surveillance cameras in the parking lot. These algorithms detect empty parking spaces and provide information to drivers.

This method is capable of managing large areas with just several cameras. Camera positions can be adjusted to improve performance, ensuring that each camera captures the entire area of every parking slot for maximum accuracy. Drivers can receive useful real-time parking lot information through a guidance information display. An integrated image processing approach is utilized to reduce the cost of sensors and wiring complexity.

This paper proposes a vehicle and parking space detection method based on an improved YOLO v3 algorithm, which achieves good results. Different scale feature maps are used for object detection, allowing deeper networks to extract more detailed features. The experiments demonstrate that the algorithm improves the accuracy of vehicle and parking space detection in the parking lot. However, factors such as illumination and weather can affect the algorithm's detection performance, necessitating further algorithmic improvements. Future research could focus on developing the same model using multiple cameras to enhance its effectiveness.

Acknowledgement

The Robotics and Imaging Parking Laboratory of Vision Lot database and yolov3 object detection algorithm have been assisted in leading these findings of this study.

References

Aekarat Saeliw, W. H. (2019). Smart Car Parking Mobile Application based on RFID and IoT. *International Journal of Interactive Mobile Technologies (iJIM)*, 1-10.

Almeida, P. O. (n.d.). Parking Lot Database. (Laboratório Visão Robótica e Imagem) Retrieved from http://web.inf.ufpr.br/vri/databases/parking-lot-database/

Benjamin Kommey, A. S. (2018). A Smart Image Processing-based System for Parking Space Vacancy Management-J. International Journal of Computer Applications.

Department of motor traffic Srilanka. (2015). Retrieved from http://www.motortraffic.gov.lk/

Edirisinghe, D. (2014). Managing traffic congestion in Colombo and its suburbs. (Sri Lanka Institute Of Development Administration Colombo Sri Lanka) Retrieved from http://www.slida.lk/slidatest/application_resources/images/mpm_reso/policy paper/pp5.pdf

Geng, Y.; Cassandras, C.G. (2010) A new "Smart Parking" System Infrastructure and Implementation. Procedia Soc.Behav. Sci. 2012.

Idris, Mohd & Y.Y, Leng & E.M, Tamil & N.M, Noor & Razak, Zaidi. (2009). Car Park System: A Review of Smart Parking System and its Technology. Information Technology Journal. 8. 10.3923/itj.2009.101.113.

Jones, M.; Khan, A.; Kulkarni, P.; Carnelli, P.; Sooriyabandara, M. (2017) ParkUs 2.0: Automated Cruise Detectionfor Parking Availability Inference. In Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services; ACM: New York, NY, USA, 2017; pp. 242–251.

J. Wolff, T. H. (2006). Parking monitor system based on magnetic eld sensors. IEEE Conference on Intelligent Transportation Systems, (pp. 1275–1279).

Jiwoong Choi, D. C.-J. (2019). Gaussian YOLOv3. Retrieved from http://openaccess.thecvf.com/content_ICCV_2019/papers/Choi_Gaussian_YOLOv3_An_Accurate_and_Fast_Object_Detector_Using_Localization_ICCV_2019_paper.pdf

K., A. M. (2013). opencv-python-tutroal. Retrieved from https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_canny/py_canny.html

Kathuria, A. (2018, 04 16). How to implement a YOLO (v3) object detector from scratch in PyTorch: Part 1. Retrieved from https://blog.paperspace.com/how-to-implement-a-yolo-object-detector-in-pytorch/

Lambrinos, L.; Dosis, A(2013). Applying mobile and internet of things technologies in managing parking spaces for people with disabilities. Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing New York, New York, USA, 2013; pp. 219–222

Mathew, S.S.; Atif, Y.; Sheng, Q.Z.; Maamar, Z. (2014) Building Sustainable Parking Lots with the Web of Things.Pers. Ubiquitous Comput. 2014,18, 895–907

Nazia Bibi, H. D. (2017). Automatic Parking Space Detection System. 2017 2nd International Conference on Multimedia and Image Processing.

Pantha, N. (2019, 04 19). Dzone. Retrieved from https://dzone.com/articles/understanding-object-detection-using-yolo

Vora, A.; Kumar, M.A.; Srinivasa, K.G (2014). Low Cost Internet of Things based Vehicle Parking Information System. In Proceedings of the 6th IBM Collaborative Academia Research Exchange Conference (I-CARE) on I-CARE2014—I-CARE 2014; ACM Press: New York, NY, USA, 2014; pp. 1–4

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P., "Gradient-Based Lerning Applied to Document Recognition," proc. IEEE, 1998, [Online]. Available: http://ieeexplore.ieee.org/document/726791/#full-text-section