



Enhancement of license plate recognition performance using Xception with Mish activation function

Anmol Pattanaik¹ · Rakesh Chandra Balabantaray¹

Received: 15 December 2021 / Revised: 12 April 2022 / Accepted: 12 September 2022 /

Published online: 14 October 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

The current breakthroughs in the highway research sector have resulted in a greater awareness and focus on the construction of an effective Intelligent Transportation System (ITS). One of the most actively researched areas is Vehicle Licence Plate Recognition (VLPR), concerned with determining the characters contained in a vehicle's Licence Plate (LP). Many existing methods have been used to deal with different environmental complexity factors but are limited to motion deblurring. The aim of our research is to provide an effective and robust solution for recognizing characters present in license plates in complex environmental conditions. Our proposed approach is capable of handling not only the motion-blurred LPs but also recognizing the characters present in different types of low resolution and blurred license plates, illegible vehicle plates, license plates present in different weather and light conditions, and various traffic circumstances, as well as high-speed vehicles. Our research provides a series of different approaches to execute different steps in the character recognition process. The proposed approach presents the concept of Generative Adversarial Networks (GAN) with Discrete Cosine Transform (DCT) Discriminator (DCTGAN), a joint image super resolution and deblurring approach that uses a discrete cosine transform with low computational complexity to remove various types of blur and complexities from licence plates. License Plates (LPs) are detected using the Improved Bernsen Algorithm (IBA) with Connected Component Analysis (CCA). Finally, with the aid of the proposed Xception model with transfer learning, the characters in LPs are recognised. Here we have not used any segmentation technique to split the characters. Four benchmark datasets such as Stanford Cars, FZU Cars, HumAIn 2019 Challenge datasets, and Application-Oriented License Plate (AOLP) dataset, as well as our own collected dataset, were used for the validation of our proposed algorithm. This dataset includes the images of vehicles captured in different lighting and weather conditions such as sunny, rainy, cloudy, blurred, low illumination, foggy, and night. The suggested strategy does better than the current best practices in both numbers and quality.

✉ Anmol Pattanaik
c119001@iiit-bh.ac.in

Rakesh Chandra Balabantaray
rakesh@iiit-bh.ac.in

¹ International Institute of Information Technology Bhubaneswar, Odisha, India

Keywords Generative Adversarial Networks · Xception · Connected Component Analysis · Improved Bernsen Algorithm · Transfer Learning

1 Introduction

The use of VLPR in applications like parking, road traffic monitoring, tracking of vehicle, and intelligent transportation systems has grown significantly. A LP is a metal plate with letters as well as text that is permanently attached to the exterior of the vehicle's body and used to identify vehicle. This work is particularly difficult because of the variety of plate formats and the variable outside lighting conditions encountered during image capture, such as backdrop [67, 76], illumination [8, 21, 72], speed limit [1], and interspace between camera and vehicle [60]. As a result, the majority of approaches only function in specific situations.

For their great visual quality, images with high definition and finely detailed textures are preferred. Images of LP taken in natural settings, on the other hand, frequently contain intricate, fuzzy artifacts. These hazy structures not only degrade the quality of the LP, but also obscure intricate textures. Unwanted artifacts in the camera lens can decrease the perceived quality of the captured vehicle. This can happen if the vehicle is not properly aligned or moves during the image capture. As a result, one of the key issues in the LP acquisition procedure is obtaining blurred LPs under various challenging circumstances. Much work has previously been done in this area, but it was mostly limited to motion blur [25, 71]. However, additional types of blur can degrade the quality of licence plates. For example, natural fog interference can cause haze blur, optical lens distortion can cause defocus blur, and atmospheric turbulence can cause Gaussian blur, all of which can have a substantial impact on the LP detection and recognition process. As a result, it is important to look into removing blurry artifacts and recreating sharp images.

Numerous research investigations have been conducted over the past decade in order to improve the process of recognising LPs. The majority of this research relies on established image preprocessing algorithms, such as localising LPs, extraction of characters, and pattern recognition [19].

Image super-resolution and deblurring are normally treated individually because they are both recognised to be severely ill-posed problems. On the other hand, images of LPs captured in the natural environment are typically of low-resolution and have complicated blurring patterns. While it may seem logical to perform image super resolution [50], followed by image deblurring [48], and vice versa, a hybrid mechanism of the two methods will lead to numerous complications. As a result, we must consider combining these two processes in order to conserve memory and time, thus increasing computing efficiency. We use the architecture of GAN with DCT loss to address the aforementioned joint problem because remarkable progress has recently been made in related fields of image enhancement [51] by employing GANs [27], which are well-known for their capacity to maintain textural features in images, generate solutions that are close to the real image manifold, and appear perceptually convincing.

After thorough investigation, we have incorporated some of the research gaps which motivates us to do research in this particular area. The main research gaps are given below.

1. A lot of work has been done for blurry licence plate text recognition but only limited to motion blur.

2. Researchers of different have been wondering for years how to get sharp images from low-resolution and blurred images, but the challenge has yet to be resolved because of the intricacy of the blurring process and, more importantly, the rich information in high-resolution images. Most existing approaches may not generate adequate results when the blur kernel is intricate and the intended sharp images are detailed.
3. Existing algorithms exhibit a clear deficiency in terms of LP recognition efficiency, particularly when images are degraded in various ways, such as by flashing, blurring, darkening, or any other external noise.
4. Existing image enhancement approaches only enhance the quality of an image but very few research has been done on joint approach of image deblurring and super-resolution which not only enhances the quality of image but also removes the blurry artifacts.

The main advantage of our proposed approach is that it can provide a solution to the above mentioned problems. The primary objective of our research is to lay out a joint image super resolution and deblurring model to perform on the license plate images present in various challenging environmental condition which results in providing images with great visual quality in order to improve the recognition efficiency and to conserve memory and time. The secondary objective to use transfer learning in character recognition process to get more generalise result.

Our contribution to this research focuses on finding a solution for image disruption caused by different illumination and a diversity of outside situations such as shadows, blurriness, and exposure, which are notoriously tough to control using conventional methodologies. Additionally, we have considered the system's overall performance as well as the performance of each particular model.

The remainder of this paper is laid out as follows: Section 2 provides a concise review of related work. Section 3 summarises the integrated model and provides an overview of each component. Section 4 gives description about various datasets used for our proposed approach. Section 5 continues with experimental verification and discusses the limitations of our proposed work, and Section 6 offers conclusions as well as recommendations for future work.

2 Related work

Machine learning is one of the top emerging technology which has an extremely broad range of applications. Deep learning is a relatively new approach that has received much scrutiny and discussions in recent years. These two techniques have been widely employed in a variety of sectors, including the healthcare industry [2, 61] (in areas, such as cancer detection [36–38], diabetes prediction [10], heart disease prediction [11], liver disorder [30], COVID-19 diagnosis [4, 9, 39, 41, 59, 63, 64, 66]), the Internet of Things [31], and artificial intelligence, for example, semantic parsing, emotion recognition [40, 42], transfer learning, natural language processing, and computer vision [36] and many more [20, 43]. As license plate character recognition is a multi-step procedure, considerable research has been conducted in order to identify characters by employing traditional as well as machine learning and deep learning approaches.

The primary and most important phase of VLPR is image acquisition process. Two major circumstances from which vehicle image can be collected are parking lot surveillance and open road tolling. The requirements may vary significantly based on the circumstances. For example, image captured in parking lot applications are having slow speed where the risk of motion blur is very low whereas, image captured from highway applications are much

trickier and can be motion blur. The image captured from the scene may be complicated by the camera's type, resolution, lighting aids, mounting location, complex situations, shutter speed, and other environmental and system constraints. along with this, the authors [13] discusses an advanced technique which uses a smart camera that is intended for use in security and law enforcement applications in intelligent transportation systems.

Since the mid-2000s, deep learning technology has made significant advancements, particularly in the field of object detection [53]. Numerous algorithms have been proposed for detecting plates. Some of these algorithms work by looking for horizontal and vertical edges [1, 8, 60]. Such algorithms use the Canny edge detector to localise plates. Other algorithms detect plates by locating their boundaries using the hough transform [25, 71], which is a tedious technique that requires a large amount of memory. [50]. Plates with no visible boundaries are impossible to detect using this method. Additionally, plate detection has been accomplished through the use of wavelet analysis. [19, 48]. Wavelet-based approaches use high-frequency coefficients to identify plate candidates. Because these coefficients are related to edges, they have many of the same drawbacks as edge detection algorithms. Certain detection algorithms combine mathematical morphology with CCA [27, 51]. These algorithms require images with a moderate to high contrast range for plate detection. Furthermore, colour is not a reliable feature for detecting dirty plates. Hojin et al. [74] suggested a homography transformation model based on deep neural networks for rectifying a tilted licence plate.

There are two types of existing image super-resolution approaches: conventional techniques and deep learning-based techniques [5, 6]. Shallow learning methods [45, 55, 69] are presented to address challenges based on reconstruction methods. These techniques rebuild images by establishing a mapping link between the images of low and high intensities. However, issue with this technique is that different steps of this method are unrelated, and the model's ability to extract features is insufficient. Deep learning has been frequently employed in recent years to solve challenges requiring super-resolution. Approaches based on convolutional neural networks (CNNs) [24, 46, 49] are used to address various shortcomings associated with shallow learning methods. Additionally, Ledig et al. [52] incorporated a GAN for improving the aesthetic effect of super-resolution significantly. Similarly, Liu et al. [57] presented a multi-scale skip connection network (MSN) in order to enhance the visual quality of the super resolution image.

There are numerous character segmentation techniques available, out of which, some are based on morphological operations [28] and others are based on CCA (CCA) [32, 54]. Prior to applying any further processing, a suitable thresholding approach must be employed to obtain a binary image of the plate. Plate binarization can be achieved using thresholding methods such as Niblack [14], SAUVOLA [62], Wolf and Jolion [65], and OTSU [70].

Character recognition has been accomplished through the use of a variety of classification methodologies in the past, including Artificial Neural Networks (ANN), Support Vector Machines (SVM), Bayes classifier, K-nearest neighbor, etc. Classifiers are provided with the features retrieved from image segmentation. Numerous techniques have been proposed for feature extraction, including character skeleton [17], active areas [34], HOG [47], horizontal and vertical projection [22], and the multiclass AdaBoost approach [73]. The character recognition system in some methods, such as SIFT [7] and SURF [26], is focused on key point localization. Menotti et al. [58] advocated the use of random CNNs for extracting features to recognise characters, demonstrating much improved results than image pixels or back-propagation based learning. Li and Chen [56] suggested treating character recognition as a challenge of sequence labelling. To label the sequential data, a Recurrent Neural

Network (RNN) is used in conjunction with Connectionist Temporal Classification (CTC), which recognises the entire LP without character-level segmentation. Sometimes, digitised texts are noisy and require post-correction. Numerous articles demonstrated the critical nature of improving OCR results by examining their impact on information retrieval and natural language processing applications. A research published recently [35] suggests an OCR post correction system based on a parametrized string distance measure.

3 Proposed work

This section explains how our proposed model works. Our paper begins with the process of LP localization using the Haar Cascade technique, followed by the DCTGAN-based joint image super resolution and deblurring method to get sharp and clear images. The LP is then detected using the CCA method. Finally, the proposed Xception model with transfer learning is used to identify the characters in LP. Figure 1 shows a visual representation of the total workflow of our proposed model.

The algorithm of our proposed work is given below in step wise manner:

1. Haar Cascade classifier is implemented for localizing LPs.
2. Then, joint image super resolution and deblurring approach is used to restore the high resolution clear image from low resolution blurred image.
3. LP images are detected using the Improved Bernsen Algorithm with CCA.
4. Finally, LP characters are recognised by employing the modified Xception model with transfer learning.
5. Proposed model's performance is validated on four distinct test datasets as well as our own dataset.

3.1 License plate localisation using Haar Cascade classifier

Face detection was the first application to make use of a Haar Cascade classifier. A set of haar-like characteristics is extracted in the default window. Haar Cascade is a machine learning-based technique for identifying and emphasising the region of interest in an image. Here region of interest is the location of license plate. It is a classifier that can tell the difference between the training object and the remainder of the image. So, it can distinguish between the LP location and the rest of the picture. In its simplest form, a Haar Cascade is an XML file containing the object's feature set.

For the training process of the Haar Cascade classifier, a large number of both positive and negative samples are required. Here positive sample implies the images of the object to be trained and negative sample refers to the random images. It is necessary to ensure that the negative samples must not contain any component of the object. In the framework suggested in this paper, a primitively trained Haar Cascade is employed. It has been shown to be more effective for LP localization than other algorithms.

3.2 A combined approach for image super-resolution and deblurring based on DCTGAN

This part begins with a review of the core formula of GAN. After that, we have discussed the suggested network architecture as well as the loss function that will be used in training.

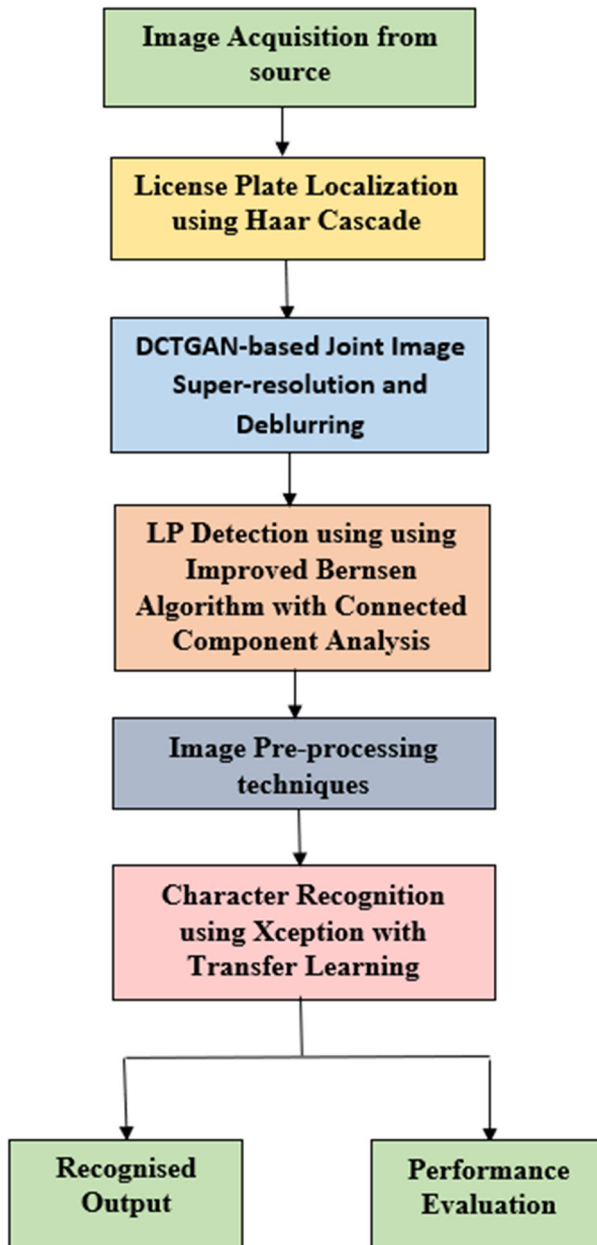


Fig. 1 Flowchart of our Proposed Work

Generative adversarial network Goodfellow et al. [27] first proposed the concept of GAN. They were looking to define a competition between two contending networks: the discriminator and the generator. It is composed of two models [28]: a generative model (G) and a discriminative model (D). A sample is generated by the generator throughout the training

phase, which accepts noise as an input and then generates a sample. The objective of the generator G is to produce as much genuine data as possible in order to trick the discriminator D , while the target of the discriminator D is to separate the generated data from the actual data that is available. The goal of GAN is to make the discriminator unable to differentiate between produced and actual data, as well as to equalise the distribution of these data. Our generator accepts a blurry and low-resolution image as input and outputs a high-quality deblurred image. The discriminator distinguishes between the generated image and the true, clear image.

3.2.1 Architecture of generator

Figure 2 illustrates the generator network. We employ an encoder-decoder structure that has been proven to be effective in addressing the low resolution and deblurring problems. Unlike SRN, DeblurGAN, and Deblur-GANv2, our encoder-decoder lacks a multi-scale technique for speeding up computation. The proposed generator has a single InBlock, two Encoder Blocks (EBlocks), eleven Residual Blocks (ResBlocks), two Decoder Blocks (DBlocks), and a single OutBlock as its components. We use Parametric ReLU (PReLU) [32] to update the ideal parameters during training in order to avoid overfitting when the number of hidden layers is large. The InBlock is made up of a single convolutional layer and a single PReLU activation function. Each EBlock has one layer of convolution, one layer of batch normalisation, and one layer of PReLU. Likewise, each DBlock consists of a deconvolution layer, a batch normalisation layer, and a PReLU layer. EBlock and DBlock both incorporate batch normalisation to facilitate training, while DBlock is symmetrical to EBlock. Each ResBlock is a replication of the original ResNet [54]. (However, the activation function is PReLU). The OutBlock begins with a convolution layer and ends with a tanh activation function to normalise the final output. The first and last convolution layers in this network employ 7x7 kernels, whereas the remaining convolution layers use 3x3 kernels.

3.2.2 Architecture of discriminator

The architecture of the DCT discriminator is depicted in Fig. 3. This method is similar to Lee's suggested GAN-based image super-resolution approach [54]. Rather than using ReLU, our proposed discriminator makes use of the LeakyReLU activation function. In our approach, each convolutional block is composed of a convolutional layer, a batch

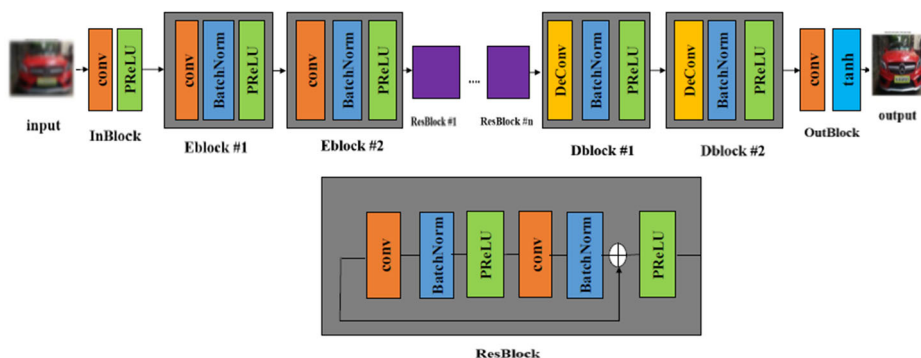


Fig. 2 Generator Architecture

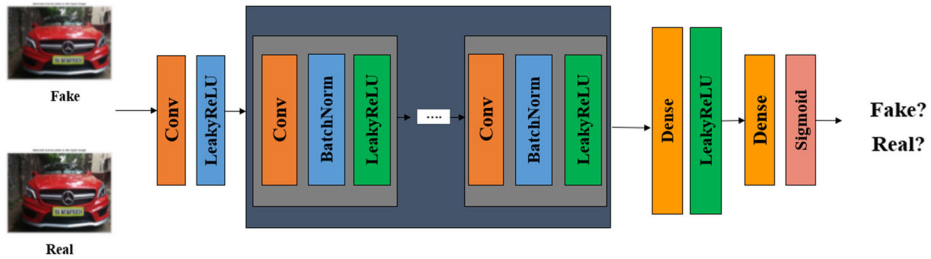


Fig. 3 Discriminator Architecture

normalisation layer, and a LeakyReLU layer. There are five ConvBlocks. In Fig. 4, the final result is normalised using a sigmoid function.

3.2.3 Step-wise working principle

The step-wise working principle of DCTGAN model is as follow:

1. The generator accepts images of degraded quality as an input and generates high resolution and deblurred images passing through different blocks of the generative model.
2. Then the generated image $G(y)$ and the real sharp/clear image x are converted to grayscale images.
3. Now the images are transformed into the frequency domain applying DCT in order to efficiently train the network.
4. These images are fed as an input to the discriminative model. So, the model is called as DCT Discriminator model.
5. Both the generated image from the generator and the real image are fed to the discriminator

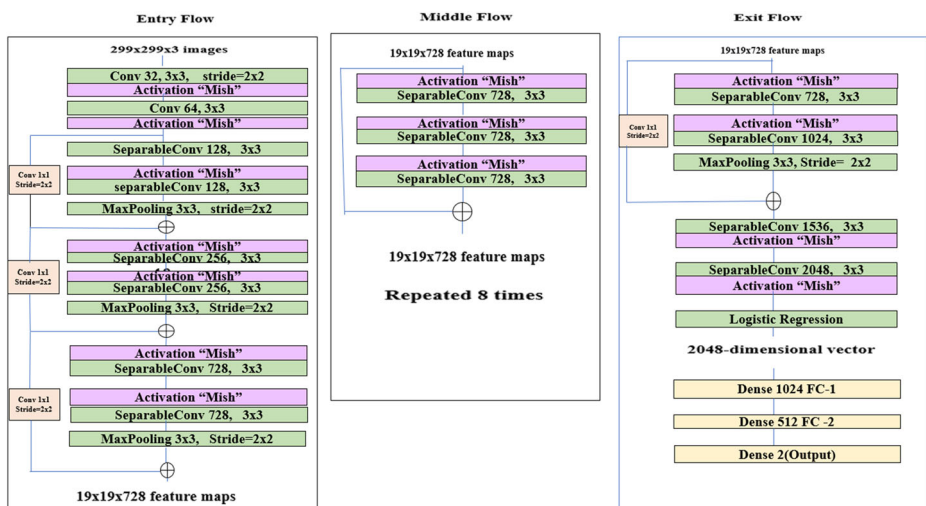


Fig. 4 Xception Model Architecture

6. Like the generative model, these images also pass-through different blocks of discriminative model to distinguish between the actual image and the generated image till loss function is minimised

3.2.4 Losses

Generally, the GAN generator is pre-trained to avoid adversarial training failure. In Fig. 5, Mean Squared Error (MSE) is described as the pre-processing loss in our paper

$$L_{MSE} = ||x - G(y)||^2 \quad (1)$$

The learning generator's loss function of DCTGAN is specified as follows:

$$L_{overall} = \alpha L_{abs} + \beta L_{cont} + \gamma L_{adv} + \delta L_{DCT} \quad (2)$$

Where $\alpha, \beta, \gamma, \delta$ are the parameters, L_{abs} is absolute deviations between the output of generator and ground truth image in our method defined as

$$L_{abs} = |x - G(y)| \quad (3)$$

L_{cont} denotes the content loss between the $G(y)$ and x feature maps in the VGG-19 network pre-trained on ImageNet [17], L_{adv} is the adversarial loss function defined as

$$L_{adv} = \min_G \max_D \mathbb{E} [\log(D(x))] + \mathbb{E} [\log(1 - D(G(y)))] \quad (4)$$

L_{DCT} is the DCT loss function which is least absolute deviation between the absolute value of the sharp image and the generated image is defined as

$$L_{DCT} = ||DCT(G(y))| - |DCT(x)|| \quad (5)$$

Parameters of DCTGAN algorithm: Table 1 summarises the parameters utilised in DCTGAN joint image super-resolution and deblurring model. For the two models of the generative adversarial network, we used two distinct activation functions. The generator model implements the model using the PReLU activation function, whereas the discriminative model implements the model using the Leaky ReLU activation function. Total computing time to execute DCTGAN model is 139s 263ms.

3.3 License plate detection

It is possible that the light intensity of distinct locations on a LP image will change depending on the severity of the plates and the intensity of the illumination. As LP is obtained through a variety of illumination conditions and challenging backgrounds, shadows and uneven illumination can not be deleted from the LP. As a result, eliminating shadows or

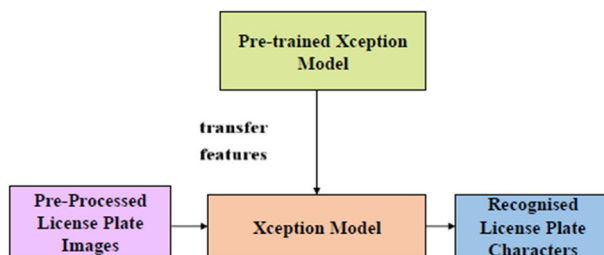


Fig. 5 Transfer Learning

Table 1 Parameters of the DCTGAN Algorithm

Parameter used	Description
ReLU Activation function	It is used in generator for the purpose of updation of the ideal parameters during the training and to avoid overfitting when the number of hidden layers are more.
Leaky ReLU Activation function	Discriminator uses Leaky ReLU activation function to make the layer much more optimised and to get better performance result.

uneven illumination is critical. Due to the fact that a binary approach with a global threshold may not produce adequate results in certain situations all the time, some adaptive local binary methods, such as the Bernsen Algorithm [14] and the Niblack algorithm [62], are frequently used. As evidenced by the literature, these algorithms are susceptible to noise, which interferes with the extraction of characters. To resolve this issue, we have implemented an improved version of the Bernsen Algorithm in this paper called the Improved Bernsen Algorithm (IBA).

3.3.1 Improved Bernsen Algorithm

Assume that $f(m, n)$ represents the grey value of the point (m, n) . Consider a block with a centre defined by a point (m, n) and a size defined by $(2z + 1) \times (2z + 1)$. The threshold $T(m, n)$ of $f(m, n)$ is evaluated by

$$T_1(m, n) = \frac{\max_{-z \leq s, b \leq z} f(m + b, n + s) + \min_{-z \leq s, b \leq z} f(m + b, n + s)}{2} \quad (6)$$

Then, the binary image is as follow

$$B(m, n) = \begin{cases} 0 & \text{if } f(m, n) < T_1(m, n) \\ 255, & \text{else} \end{cases} \quad (7)$$

Assume that x represents the grey value acquired by the Gaussian filter, y implies the scale of the Gaussian filter, and s and b are the dimensions of the window. The steps for the IBA are as follows.

1. Using (6) as a guide, calculate the threshold $T_1(m, n)$ of $f(m, n)$.
2. Compute the Gaussian filter for the window $w = (2z + 1) \times (2z + 1)$ of $f(m, n)$ i.e.,

$$\hat{f}(m, n) = \frac{1}{(2z + 1)^2} \sum_{m, n \in w} f(m, n) \times \exp\left(\frac{-1}{2} \left[\left(\frac{m}{\sigma}\right)^2 + \left(\frac{n}{\sigma}\right)^2\right]\right) \quad (8)$$

3. Calculate the threshold $T_2(m, n)$ of $\hat{f}(m, n)$ as

$$T_2(m, n) = \frac{\max_{-z \leq s, b \leq z} \hat{f}(m + b, n + s) + \min_{-z \leq s, b \leq z} \hat{f}(m + b, n + s)}{2} \quad (9)$$

4. We can derive a binary image by

$$B(m, n) = \begin{cases} 0, & \text{if } f(m, n) < \beta(1 - \alpha)T_1(m, n) + \alpha T_2(m, n); \beta \in (0, 1) \\ 255, & \text{else} \end{cases} \quad (10)$$

α denotes the variable that will be used to adjust the trade-off between BA and Gaussian filter ($\alpha \in [0, 1]$). When α approaches to 0, the predicted model is called a BA. When α

is close to one, the proposed method is BA algorithm in conjunction with a Gaussian filter.

5. Utilize a median filter to eliminate noise.

3.3.2 Connected Component Analysis (CCA)

CCA [65] is a popular image processing approach that analyses an image and classifies its pixels into components on the basis of their pixel connectivity, which indicates that all pixels present in a connected component have an identical level of pixel intensity. CCA can be used with binary or gray-level images, and several connection measurements are supported. In this present application, we used CCA to apply to the binary images. Two detection algorithms were used: The first one detected a white frame using CCA, while the second detected black characters using CCA. If the variety of licence plate is unknown, the procedure for detecting licence plates is as follows:

Approach 1 Placement of LP with a frame. Candidate frames are identified based on the foreknowledge of LPs. Several candidate areas can be identified using approach 1. A frame is recognised using the connected component algorithm, which is edge-sensitive. When the frame is damaged, the LP can not be identified accurately. Approach 1 is straightforward to calculate, and one must first determine whether the recognised area corresponds to the plate size.

Approach 2 Placement of LP without a frame. When it is impossible to detect the LP frame, approach 2 is used. By utilising huge numeral method, this approach detects the plate. When an LP with white text or a black background is not anticipated, the reverse image is inspected.

In general, approach 1 takes less time than approach 2. The implementation of approach 1 or approach 2 in the candidate frame may result in simultaneous provision of the maximum number of frames. To achieve a uniform size for connected components, certain criteria are used depending on advanced information about the characters, such as the pixel quantity of a connected component, a width of more than 10, a height of more than 20, and a height-to-width ratio of less than 2.5 or more than 1.5. This ensures that all preserved related components are of comparable size. Then, to eliminate some non character-linked components, additional limitations such as the knowledge about how far apart two characters are or the average character tilt angle are given by the character position information on the LP. Thus, discrete connected components are deleted, and the remaining connected components may be the retained characters.

3.4 Character recognition using Xception with transfer learning

3.4.1 Proposed Xception network architecture

An improved Inception [70] variant, designed by Google, is called the Xception [17] model. This Google-developed model implements an advanced version of Inception [70], by utilising a refined depth wise separable convolution algorithm based on the hypothesis that the cross-channel correlations will be mapped with a 1x1 convolution, and then the spatial correlations of each output channel will be mapped separately. It is composed of a sequence of separable convolutions (“referred to as Extreme Inception”), residual connections that are comparable to ResNet [34], and a number of standard operations.

It is 126 layers deep and includes 36 convolutional layers for extracting features. Those 36 convolutional layers are divided into 14 modules, each of which is connected by a linear residual connection, with the exception of the beginning and end modules, which have non-linear skip connections. The original xception model uses average-pooling layer after convolutional layer, but in the proposed xception framework, we used a layer of logistic regression instead of average-pooling layer. We employ two fully connected layers between the convolutional layers and the output layer to improve the accuracy of the results. ReLU activation function of the original model is replaced by Mish activation function. Three distinct components comprise the 36 convolutional layers: input flow, middle flow, and exit flow. The data is routed through the input flow first, then the middle flow eight times, and lastly, the exit flow, before being sent to the final destination. It has eight convolutional layers in the entry flow, three convolutional layers in the middle flow (as a result, a total of twenty-four convolutional layers in the middle flow), and four convolutional layers in the exit flow.

3.4.2 Transfer learning

Transfer learning is a type of machine learning in which a model generated for one task is utilised as the basis for another. As its title indicates, it basically relocates parameters from a previously trained model to a new model in order to aid in the training of the newly learned model. The process of transfer learning begins with the training of a basic model on original data sets, followed by the transfer of learned features to the target system and retraining of the target data sets.

3.4.3 Step-wise working principle

The step-wise working principle of modified Xception model is as follow:

1. Create an Xception model with parameters that have been randomly initialised.
2. Obtain the weights from the pre-trained Xception model using the ImageNet datasets and apply those to the initialised Xception model excluding the final two layer.
3. Include two fully connected convolutional layer (1024 and 512) as well as an output layer in the final two layers.
4. Include Logistic Regression Layer instead of GlobalAveragPooling layer.
5. Use Mish activation function instead of ReLU.
6. We train the model without allowing any model layers to be frozen throughout the training procedure.

Parameters of proposed Xception model: Different parameters are used for the implementation of the proposed Xception model which is elaborated in Table 2. It elaborates about the activation function, classifier, loss function, optimisation algorithm, learning rate, dropout rate and number of epochs used for the execution of the model.

4 Datasets

We have used five different datasets for the implementation of our proposed approach, out of which four datasets are publicly available and one dataset is self-collected. Four publicly

Table 2 Prameters of the Proposed Xception Algorithm

Parameter used	Name/Value	Description
Activation function	Mish	Due to the smooth and non monotonic nature of Mish activation function, we replaced the ReLU activation of the original model with Mish activation function so that its unbounded above and bounded below property can improve the performance of the proposed model.
Classifier	Softmax Classifier	We have used softmax classifier in the output layer of the proposed Xception framework which is otherwise called as multinomial logistic regression as it performs better in multi-class classification problem by giving probability for each class label.
Loss function	categorical cross entropy	Due to the multi-category nature of the challenge, categorical cross entropy is employed as the model’s loss function.
Optimization algorithm	Adam optimization algorithm	Model’s loss function is minimised by implementing the Adam [47] optimization algorithm. It is utilised in particular to optimise the Xception network.
Learning rate	0.001	It has been choosen heuristically
Dropout rate	0.5	It has been choosen heuristically
Total number of epochs	50	It has been choosen heuristically

available datasets are:Stanford Cars Dataset, FZU Cars Dataset, HumaIn 2019 challenge Dataset and Application Oriented License Plate dataset.

4.1 Existing dataset used

A combination of four publicly available datasets was used for the experimental investigation. In all the below datasets, 80% of the data is used for training purpose and 20% of the data is used for validation purpose. Figures 6, 7, and 8 represent the sample datasets used in the proposed method.

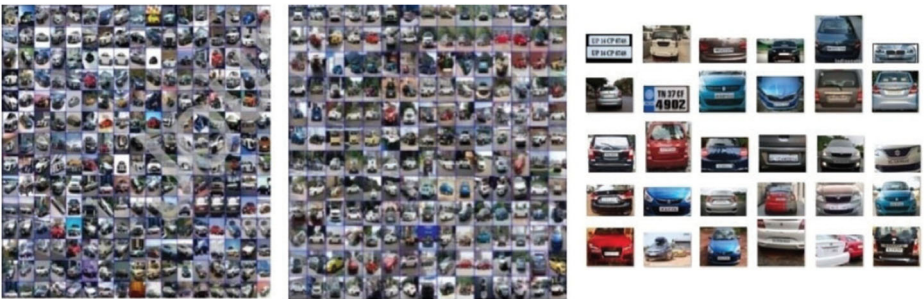


Fig. 6 Stanford cars dataset,FZU cars dataset, HumaIn 2019 challenge dataset



Fig. 7 Sample of AOLP dataset with three subsets: AC, LE and RP

4.1.1 Stanford cars dataset

The Stanford Cars collection features 297 model cars and 43,615 images in its original version.

4.1.2 FZU cars dataset

In the FZU car dataset, 196 model cars were used, comprising 16,185 images.

4.1.3 HumAIn 2019 challenge dataset

Images from the HumAIn 2019 Challenge dataset were used (<https://campuscommune.tcs.com/enin/intro/contests/tcshumain-2019>).

4.1.4 AOLP dataset

2049 Taiwan licence plate images are available in the application oriented licence plate (AOLP) benchmark database with three distinct subsets: access control (AC), which contains 681 samples, traffic law enforcement (LE), which comprises 757 samples, and road patrol (RP), having 611 samples. AC refers to instances when a vehicle crosses a fixed route at a reduced speed or comes to a complete stop. The images were taken in a variety of lighting and weather circumstances. LE refers to instances in which a vehicle breaks traffic laws and is filmed on video at a roadside. The background is quite busy, with a road sign and many plates all crammed into a single image. RP refers to situations in which the camera is mounted on a patrol vehicle and images are taken from arbitrary views and ranges. Any



Fig. 8 Our own dataset

two subsets of data are used to train our model, and the third is used for validation. Here, AC and RP is used for training purpose and LE is used to validate model's efficacy.

4.2 Data acquisition

We integrated two distinct data sets in this research and used them for training, testing, and evaluation purposes. A total of 32,00 vehicle images were collected for the data acquisition procedure. Major automobile information websites such as Autocar, Xcar, Auto Sina, and Sohu Auto present images of the general appearance of different vehicles. As a result, vehicle images can be gathered using a customer-friendly image capture approach without interfering with normal internet access. Utilizing this customer-friendly technique, we have acquired over 22,00 images of vehicles in a variety of environmental conditions. In addition, we manually collected 10,00 vehicle images, out of which 20% were photographed using three distinct cameras, i.e., a Nikon digital camera, an iPhone 7 Plus, and an Android mobile, and 80% were collected from the internet. Images captured with multiple cameras often do not have the same clarity, even if their resolution and frame rate are same. This is owing to the fact that different cameras have different focusing, bit rate, lens length, and optical image stabilisation parameters. These images were taken on roads and city streets under various lighting conditions (sunny, night, rainy, cloudy). The images were taken at distinct periods of the day and on several days throughout the year, allowing for significant variations in shadows, illuminations, reflections, and weather conditions. The dataset is freely available and will be provided on request.

5 Result and discussions

We have undertaken comprehensive analysis in this area to evaluate our LP identification system against state-of-the-art recognition methods. Numerous experiments were conducted to illustrate the efficacy of the proposed model on five distinct LP datasets.

Using a PC setup with an Intel Core i5, 8th generation processor, and 16 GB of RAM, the described approach was evaluated. The suggested simulation is performed using the Python 3.6.5 tool in conjunction with several packages, including tensorflow, keras, numpy, pickle, matplotlib, sklearn, pillow, and opencv-python to achieve the desired results.

5.1 Performance metrics

Four assessment metrics i.e., accuracy, precision, recall, and F1 Score are used to conduct performance evaluation. Four matrices have the following formulae:

$$Precision = \frac{TP}{TP + FP} \quad (11)$$

$$Recall = \frac{TP}{TP + FN} \quad (12)$$

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (13)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (14)$$

where TP = True Positive, TN = True Negative, FP = False Positive and FN = False Negative. In our evaluations, we have calculated the overall accuracy of our proposed model and compared with the existing models. It is calculated based on the formula:

$$\text{Overall Accuracy} = (D \times S \times R)\% \quad (15)$$

where D = Plate Detection Rate, S = Plate Segmentation Rate and R = Character recognition Rate.

Table 3, analyses the performance of the State-of-the-Art LP recognition system on the reported datasets. We have calculated LP detection rate and how effectively the characters present in our LP are recognised for our proposed approach. As we have not performed any segmentation technique, we calculated the overall accuracy of our proposed model based on these above two parameters. The table gives a comparative analysis of the detection rate, segmentation rate, recognition rate, overall accuracy, and the type of character used in those models and our proposed approach. We can observe that our system is superior to the previous models with the highest plate detection rate of 0.993, the highest recognition rate of 0.990, and an overall accuracy rate of 0.983. Although [7] and [18] use same dataset for their model as us, there is a significant improvement in the result of our proposed model.

We have also compared the results of different existing models, i.e., VGG16, ResNet50, InceptionV3, MobileNetV2, Faster RCNN and YOLO 9000, with our proposed model on four different benchmark datasets and one self-collected dataset, which are depicted in Tables 4, 5, 6, 7 and 8. In addition to that, the execution time of all the models with respect to our self-collected dataset is also mentioned in Table 8.

In Table 4, comparison is performed for the FZU Cars dataset and our model produced a precision of 0.978, a recall of 0.966, and an F1-Score of 0.972 which is highest as compared to the existing models.

Similarly, Table 5, provides the analysis for the Stanford Cars dataset. Proposed model gives a 0.979 precision score, a 0.969 recall score, and a 0.974 F1-score. We can see a

Table 3 Comparison of State-Of-The-Art ANPR systems with the proposed ANPR System

system	Plate detection	Character segmentation	Character recognition	Overall accuracy	Character type
Ref [22]	NR	0.99	NR	NR	Korean
Ref [73]	0.971	0.983	0.978	0.935	English Japanese
Ref [7]	0.965	NR	0.891	0.86	English
Ref [26]	0.973	NR	0.945	0.919	Persian
Ref [23]	0.969	0.987	0.945	0.904	Persian
Ref [44]	0.959	NR	0.923	0.90	Chinese English
Ref [12]	0.993	NR	0.966	0.96	Persian
Ref [16]	0.973	NR	0.957	0.931	English
Ref [29]	0.971	NR	0.964	0.936	English
Ref [18]	NR	NR	NR	0.91	English
Ref [15]	0.979	NR	0.956	0.937	English
Proposed system	0.993	-	0.990	0.983	English

^a NR: Not Reported

Table 4 Result analysis of existing models with proposed model on FZU Car dataset

Model	Precision	Recall	F1-Score
VGG16	0.942	0.920	0.940
ResNet50	0.953	0.935	0.971
InceptionV3	0.954	0.907	0.963
MobileNetV2	0.973	0.929	0.970
Faster RCNN	0.970	0.902	0.965
YOLO 9000	0.974	0.873	0.969
Proposed Model	0.978	0.966	0.972

Table 5 Result analysis of existing models with proposed model on stanford cars dataset

Model	Precision	Recall	F1-Score
VGG16	0.907	0.850	0.876
ResNet50	0.911	0.872	0.918
InceptionV3	0.932	0.908	0.964
MobileNetV2	0.941	0.929	0.951
Faster RCNN	0.945	0.931	0.959
YOLO 9000	0.951	0.905	0.962
Proposed Model	0.979	0.969	0.974

Table 6 Result analysis of existing models with proposed model on HumAIn 2019 dataset

Model	Precision	Recall	F1-Score
VGG16	0.898	0.870	0.884
ResNet50	0.891	0.879	0.907
InceptionV3	0.850	0.830	0.920
MobileNetV2	0.921	0.913	0.893
Faster RCNN	0.934	0.898	0.912
YOLO 9000	0.946	0.876	0.922
Proposed Model	0.972	0.961	0.967

Table 7 Result analysis of existing models with proposed model on AOLP dataset

Model	Precision	Recall	F1-Score
VGG16	0.856	0.815	0.910
ResNet50	0.907	0.887	0.909
InceptionV3	0.935	0.916	0.943
MobileNetV2	0.927	0.876	0.928
Faster RCNN	0.957	0.941	0.966
YOLO 9000	0.968	0.929	0.963
Proposed Model	0.982	0.958	0.971

Table 8 Result analysis of existing models with proposed models on our own dataset

Model	Precision	Recall	F1-Score	Execution time
VGG16	0.954	0.912	0.925	2 min 15 sec
ResNet50	0.943	0.916	0.907	62 min
InceptionV3	0.949	0.945	0.946	37 min
MobileNetV2	0.961	0.955	0.972	29 min 15 sec
Faster RCNN	0.952	0.922	0.907	9 min 78 sec
YOLO 9000	0.965	0.911	0.975	3 min 22 sec
Proposed Model	0.995	0.962	0.991	116 min 667 sec

significant improvement in recall score and small increment in precision and F1-score in our proposed model as compared to the existing models.

Likewise, the proposed model achieved precision, recall, and an F1-Score of 0.972, 0.961, and 0.967, respectively, on the HumAIn 2019 dataset which is reflected in Table 6. Experimental evidence shows that our proposed model performs better than the existing models for all performance parameters.

Table 7, focuses on the result analysis of the AOLP dataset. Our proposed approach produces a 0.982 precision score, a 0.958 recall and a 0.971 F1-Score on this dataset. As we have used AC and RP for training purpose and LE for validation purpose, it is seen that our proposed model produces good improvements in all the performance matrices.

As shown in Table 8, our model performs well on our self-collected dataset as well, with a precision of 0.995, a recall of 0.961, and an F1-Score of 0.991. Although the dataset was obtained independently utilising different cameras in challenging environmental conditions, it significantly improves outcomes for all parameters. Additionally, we determined the execution time for each model using our own dataset. Although our proposed model is a complicated one with the most layers in comparison to the other models, it executes in 116 minutes and 667 seconds, or about two hours. However, when we implemented the traditional Xception model on our self-collected dataset, the execution time was 137 minutes and 153 seconds, which is longer than our proposed Xception model.

From the experimental findings, we can observe that, in every case, our proposed model outperforms existing approaches in all performance matrices. We demonstrate that, in every scenario, our proposed model's performance metric is better than that of existing methods as well as for all the five datasets. We have also observed that our self-collected dataset performs better than the other four benchmark datasets in terms of all performance parameters. Additionally, we have implemented the standard Xception model on our self-collected dataset and discovered that our proposed Xception model executed faster than the standard Xception model.

Figure 9 demonstrates the step wise recognition of the vehicle image. The first row contains the original vehicle image presented in the dataset. Second row gives the result after localizing the LP after applying Haar cascade classifier. Third row provides the resultant images after implementing DCTGAN based joint image deblurring and super resolution technique. Fourth row reflects the result of the LP after detection technique. Finally the characters present in the LP images are provided in last row after applying proposed xception model. We can see that all the characters have been successfully recognised in the last step.



Fig. 9 Step wise Recognition Process

We have also given precision, recall and F1-score of different models in the graph for interpretation of our result for five different datasets which is visualised with the help of Fig. 10. Figure 11 depicts the accuracy and loss curves for the last ten epochs of our proposed Xception model. For accuracy curve, the X-axis represents the iteration step, while the Y-axis represents the attained accuracy value. Similarly, for loss curve, the X-axis represents the iteration step, while the Y-axis represents the obtained loss value. The accuracy curve indicates that training and validation accuracy are growing with each iteration, reaching 0.962 and 0.963 for the last iteration. This indicates that our proposed model accurately recognises the characters on licence plates. Similarly, we can see from the loss curve that loss decreases with each iteration, and the training and validation losses for our proposed model are 0.1211 and 0.1224, respectively, for the last epoch. This demonstrates that our proposed model is behaving appropriately.

Although our proposed approach performs better than the state-of-the-art approaches on different kinds of the datasets, there are some limitations. The first limitation is that

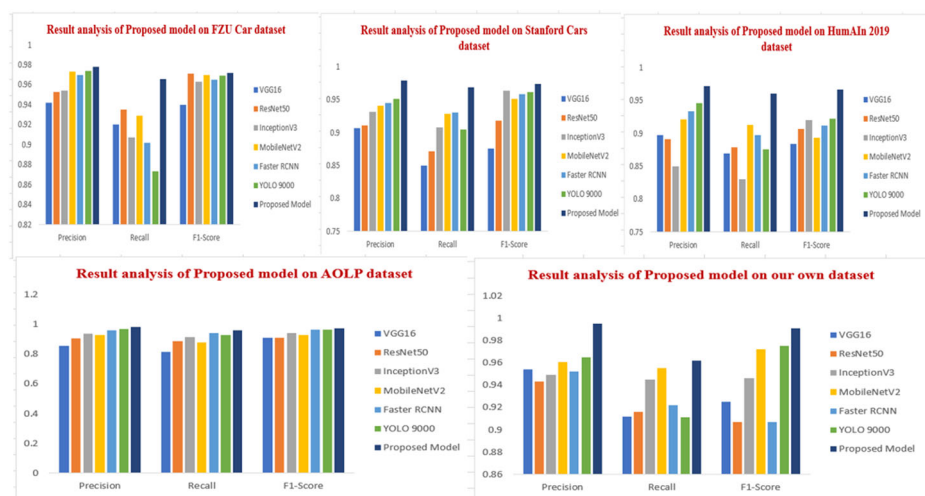


Fig. 10 Comparative Result Analysis of Existing and Proposed methods on different Datasets

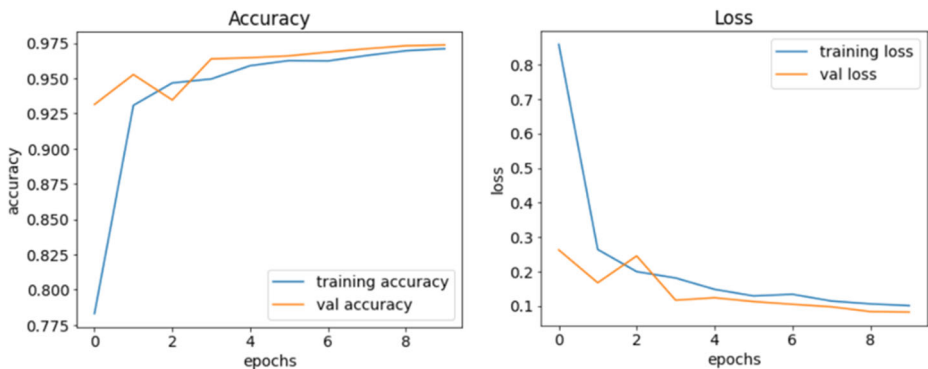


Fig. 11 Accuracy and Loss plot of our proposed model

due to the disparity between ImageNet data and LP data, the transferred model cannot be used in its entirety with the LP recognition system. Secondly, our system fails to recognise the LPs if the image contains more than two vehicles. We have implemented our proposed algorithms in a system with GPU configuration. Although it provides significant time saving as compared to the other existing approaches, the computing time of proposed xception model is almost 2 hours which is one of the implementation challenge we have faced.

6 Conclusions

This paper offers a methodology based on the GAN for reconstructing high-resolution, clear images straight from blurred low-resolution images. The model simultaneously addresses two important problems: image deblurring and super-resolution. By using GAN and employing the DCT loss, it is helpful to limit block noise or ringing artifacts. The DCT loss, in particular, is intended to cancel out the generator's extra frequency component. During the first stage, LPs are localised with the help of the Haar Cascade, and the detection process is accomplished through the use of IBA and CCA models. To enhance the recognition rate and real-time performance of the LP recognition system, we use transfer learning in conjunction with a deep learning algorithm called Xception, with some modifications. Indeed, as compared to conventional methods, this study delivers not only great accuracy and real-time performance, but also significant time savings. In future, other deep learning approaches can be implemented which can recognize the LPs if the images contains more than two vehicles. Also in future, the efficacy of the system can be verified using LP datasets from different countries.

Declarations

Conflict of Interests The authors declare that there are no conflicts of interest.

References

- Adorni G, Bergenti F, Cagnoni S (1998) Vehicle license plate recognition by means of cellular automata. In: Proceedings of IEEE International conference on intelligent vehicles vol 2

2. Akter L, Islam MM (2021) Hepatocellular carcinoma patient's survival prediction using oversampling and machine learning techniques. In: 2021 2nd international conference on robotics, electrical and signal processing techniques (ICREST), IEEE, pp 445–450
3. Akter L, Islam M, Al-Rakhami MS, Haque M (2021) Prediction of cervical cancer from behavior risk using machine learning techniques. *SN Comput Sci* 2(3):1–10
4. Al-Rakhami MS, Islam MM, Islam MZ, Asraf A, Sodhro AH, Ding W (2021) Diagnosis of COVID -19 from X-rays using combined CNN-RNN architecture with transfer learning. *MedRxiv*, pp 2020–08
5. Aly H, Dubois E (2003) Regularized image up-sampling using a new observation model and the level set method. In: *Proceedings 2003 International conference on image processing*, IEEE, Barcelona, Spain
6. Aly HA, Dubois E (2005) Image up-sampling using total-variation regularization with a new observation model. *IEEE Trans Image Process* 14(10):1647–1659
7. Anagnostopoulos CN, Anagnostopoulos IE, Loumos V, Kayafas E (2006) A license plate-recognition algorithm for intelligent transportation system applications. *IEEE Trans Intell Transport Syst* 7(3):377–392
8. Anagnostopoulos CN, Anagnostopoulos IE, Loumos V, Kayafas E (2006) A license plate-recognition algorithm for intelligent transportation system applications. *IEEE Trans Intell Transp Syst* 7(3):377–392
9. Asraf A, Islam M, Haque M (2020) Deep learning applications to combat novel coronavirus (COVID-19) pandemic. *SN Comput Sci* 1(6):1–7
10. Ayon SI, Islam M (2019) Diabetes prediction: a deep learning approach, *international journal of information engineering & electronic business* 11(2)
11. Ayon SI, Islam MM, Hossain MR (2020) Coronary artery heart disease prediction: a comparative study of computational intelligence techniques. *IETE Journal of Research*, pp 1–20
12. Azad R, Davami F, Azad B (2013) A novel and robust method for automatic license plate recognition system based on pattern recognition. *Adv Comput Sci: An Int J* 2(3):64–70
13. Baran R, Rusc T, Fornalski P (2016) A smart camera for the surveillance of vehicles in intelligent transportation systems. *Multimed Tools Appl* 75(17):10471–10493
14. Bernsen J (1986) Dynamic thresholding of gray-level images. In: *Proc Eighth Int conf Pattern Recognition*, pp 1251–1255, Paris
15. Chang SL, Chen LS, Chung YC, Chen SW (2004) Automatic license plate recognition. *IEEE Trans Intell Transport Syst* 5(1):42–53
16. Chen ZX, Liu CY, Chang FL, Wang GY (2009) Automatic license-plate location and recognition based on feature salience. *IEEE Trans Veh Technol* 58(7):3781–3785
17. Chollet F (2017) Xception: Deep learning with depthwise separable convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 1251–1258
18. Comelli P, Ferragina P, Granieri MN, Stabile F (1995) Optical recognition of motor vehicle license plates. *IEEE Trans Veh Technol* 44(4):790–799
19. Conci A, Carvalho J, Rauber T (2009) A complete system for vehicle plate localization, segmentation and recognition in real life scene. *IEEE Lat Am Trans* 7(5):497–506
20. Das S, Sadi MS, Haque MA, Islam MM (2019) A machine learning approach to protect electronic devices from damage using the concept of outlier. In: 2019 1st international conference on advances in science, engineering and robotics technology (ICASERT), IEEE, pp 1–6
21. Davies P, Emmott N, Ayland N (1990) License plate recognition technology for toll violation enforcement. In: *IEE Colloquium on image analysis for transport applications*, pp 7–1, IET
22. Deb K, Gubarev VV, Jo KH (2009) Vehicle license plate detection algorithm based on color space and geometrical properties. In: *International conference on intelligent computing*, pp 555–564, Springer, Berlin, Heidelberg
23. Dehshibi MM, Allahverdi R (2012) Persian vehicle license plate recognition using multiclass adaboost. *Int J Electr Comput Eng* 4(3):355–358
24. Dong C, Loy CC, He K, Tang X (2015) Image super-resolution using deep convolutional networks. *IEEE Trans Pattern Anal Mach Intell* 38(2):295–307
25. Elder JH, Zucker SW (1998) Local scale control for edge detection and blur estimation. *IEEE Trans Pattern Anal Mach Intell* 20(7):699–716
26. Ghofrani S, Rasooli M (2011) Farsi license plate detection and recognition based on characters features. *Majlesi J Electr Eng* 5(2):44–51
27. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2020) Generative adversarial networks. *Commun ACM* 63(11):139–44
28. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. *Adv Neural Inf Process Syst*. p 27
29. Guo JM, Liu YF (2008) License plate localization and character segmentation with feedback self-learning and hybrid binarization techniques. *IEEE Trans Veh Technol* 57(3):1417–1424

30. Haque MR, Islam MM, Iqbal H, Reza MS, Hasan MK (2018) Performance evaluation of random forests and artificial neural networks for the classification of liver disorder. In: 2018 international conference on computer, communication, chemical, material and electronic engineering (IC4ME2), IEEE, pp 1–5
31. Hasan M, Islam MM, Zarif MII, Hashem MMA (2019) Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. *Inter Things* 7:100059
32. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
33. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
34. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
35. Hládek D, Staš J, Ondáš S, Juhár J, Kovács L (2017) Learning string distance with smoothing for OCR spelling correction. *Multimed Tools Appl* 76(22):24549–24567
36. Islam M (2020) An efficient human computer interaction through hand gesture using deep convolutional neural network. *SN Comput Sci* 1(4):1–9
37. Islam M, Haque M, Iqbal H, Hasan M, Hasan M, Kabir MN (2020) Breast cancer prediction: a comparative study using machine learning techniques. *SN Comput Sci* 1(5):1–14
38. Islam MM, Iqbal H, Haque MR, Hasan MK (2017) Prediction of breast cancer using support vector machine and K-Nearest neighbors. In: 2017 IEEE region 10 humanitarian technology conference (r10-HTC), IEEE, pp 226–229
39. Islam MZ, Islam MM, Asraf A (2020) A combined deep, CNN-LSTM Network for the detection of novel coronavirus (COVID-19) using X-ray images. *Inform Med unlocked* 20:100412
40. Islam MR, Islam MM, Rahman MM, Mondal C, Singha SK, Ahmad M, Moni MA (2021) EEG Channel correlation based model for emotion recognition. *Comput Biol Med* 136:104757
41. Islam MM, Karray F, Alhajj R, Zeng J (2021) A review on deep learning techniques for the diagnosis of novel coronavirus (COVID-19). *IEEE Access* 9:30551–30572
42. Islam MR, Moni MA, Islam MM, Rashed-Al-Mahfuz M, Islam MS, Hasan MK, Lió P (2021) Emotion recognition from EEG signal focusing on deep learning and shallow learning techniques. *IEEE Access* 9:94601–94624
43. Islam MM, Tayan O, Islam MR, Islam MS, Nooruddin S, Kabir MN, Islam MR (2020) Deep learning based systems developed for fall detection: a review. *IEEE Access* 8:166117–166137
44. Jiao J, Ye Q, Huang Q (2009) A configurable method for multi-style license plate recognition. *Pattern Recogn* 42(3):358–369
45. Jiji CV, Chaudhuri S (2006) Single-frame image super-resolution through contourlet learning. *EURASIP Journal on Advances in Signal Processing*, pp 1–5
46. Kim J, Lee JK, Lee KM (2016) Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1646–1654
47. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization, arXiv:1412.6980
48. Kupyn O, Budzan V, Mykhailych M, Mishkin D, Matas J (2018) Deblurgan: blind motion deblurring using conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8183–8192
49. Lai WS, Huang JB, Ahuja N, Yang MH (2017) Deep laplacian pyramid networks for fast and accurate super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 624–632
50. Ledig C, Theis L, Huszár F, Caballero J, Cunningham A, Acosta A, Aitken A, Tejani A, Totz J, Wang Z, Shi W (2017) Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4681–4690
51. Ledig C, Theis L, Huszár F, Caballero J, Cunningham A, Acosta A, Aitken A, Tejani A, Totz J, Wang Z, Shi W (2017) Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4681–4690
52. Ledig C, Theis L, Huszár F, Caballero J, Cunningham A, Acosta A, Aitken A, Tejani A, Totz J, Wang Z, Shi W (2017) Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4681–4690
53. Lee J, Hwang KI (2021) YOLO With adaptive frame control for real-time object detection applications. *Multimedia Tools and Applications*, pp 1–22
54. Lee OY, Shin YH, Kim JO (2019) Multi-perspective discriminators-based generative adversarial network for image super resolution. *IEEE Access* 7:136496–136510
55. Li X, Lam KM, Qiu G, Shen L, Wang S (2009) Example-based image super-resolution with class-specific predictors. *J Vis Commun Image Represent* 20(5):312–322

56. Li H, Shen C (2016) Reading car license plates using deep convolutional neural networks and LSTMs, arXiv:16010.05610
57. Liu J, Ge J, Xue Y, He W, Sun Q, Li S (2021) Multi-scale skip-connection network for image super-resolution. *Multimedia Systems*, pp 821–836
58. Menotti D, Chiachia G, Falcao AX, Neto VO (2014) Vehicle license plate recognition with random convolutional networks. In: 2014 27th SIBGRAPI conference on graphics, patterns and images, pp 298–303, IEEE
59. Muhammad LJ, Islam M, Usman SS, Ayon SI (2020) Predictive data mining models for novel coronavirus (COVID-19) infected patients' recovery. *SN Comput Sci* 1(4):1–7
60. Naito T, Tsukada T, Yamada K, Kozuka K, Yamamoto S (2000) Robust license-plate recognition method for passing vehicles under outside environment. *IEEE Trans Veh Technol* 49(6):2309–2319
61. Nasr M, Islam MM, Shehata S, Karray F, Quintana Y (2021) Smart healthcare in the age of AI: recent advances, challenges, and future prospects, *IEEE Access*
62. Niblack W (1986) An introduction to digital image processing, pp 115–116, Upper Saddle River NJ: Prentice-Hall
63. Rahman MM, Islam M, Manik M, Hossen M, Al-Rakhami MS (2021) Machine learning approaches for tackling novel coronavirus (COVID-19) pandemic. *Sn Comput Sci* 2(5):1–10
64. Rahman M, Manik MMH, Islam MM, Mahmud S, Kim JH (2020) An automated system to limit COVID-19 using facial mask detection in smart city network. In: 2020 IEEE international IOT, electronics and mechatronics conference (IEMTRONICS), IEEE, pp 1–5
65. Rosenfeld A, Pfaltz JL (1966) Sequential operations in digital picture processing. *Journal of the ACM (JACM)* 13(4):471–494
66. Saha P, Sadi MS, Islam MM (2021) EMCNet: Automated COVID-19 diagnosis from X-ray images using convolutional neural network and ensemble of machine learning classifiers. *Informatics in medicine unlocked* 22:100505
67. Salgado L, Menendez JM, Rendon E, Garcia N (1999) Automatic car plate detection and recognition through intelligent vision engineering. In: *Proceedings IEEE international carnanah conference on security technology*, pp 71–76. IEEE
68. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition, arXiv:1409.1556
69. Su C, Zhuang Y, Huang L, Wu F (2005) Steerable pyramid-based face hallucination. *Pattern Recogn* 38(6):813–824
70. Szegedy C, Ioffe S, Vanhoucke V, Alemi AA (2017) Inception-v4, inception-resnet and the impact of residual connections on learning. In: *Thirty-first AAAI conference on artificial intelligence*, pp 4278–4284
71. Wang R, Li R, Sun H (2016) Haze removal based on multiple scattering model with superpixel algorithm. *Signal Process* 127:24–36
72. Wang F, Man L, Wang B, Xiao Y, Pan W, Lu X (2008) Fuzzy-based algorithm for color recognition of license plates. *Pattern Recogn Lett* 29(7):1007–1020
73. Wen Y, Lu Y, Yan J, Zhou Z, von Deneen KM, Shi P (2011) An algorithm for license plate recognition applied to intelligent transportation system. *IEEE Trans Intell Transport Syst* 12(3):830–845
74. Yoo H, Jun K (2021) Deep corner prediction to rectify tilted license plate images. *Multimed Syst* 27(4):779–786
75. Zeiler MD, Fergus R (2014) Visualizing and understanding convolutional networks. In: *European conference on computer vision*, pp 818–833, Springer, Cham
76. Zheng D, Zhao Y, Wang J (2005) An efficient method of license plate location. *Pattern Recognit Lett* 26(15):2431–2438

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.