ISSN No:-2456-2165

License Plate Image Upscaling using GANs

Mayank Kinger
Department of Computer Science and Engineering
SRM Institute of Science and Technology
Kattankulathur, India

Mayur Prajapati
Department of Computer Science and Engineering
SRM Institute of Science and Technology
Kattankulathur, India

Ms. G. Malarselvi
Assistant Professor of Computer Science and Engineering
SRM Institute of Science and Technology
Kattankulathur, India

Abstract:- License plate recognition systems are necessary for a wide range of tasks, including law enforcement, surveillance, and toll booth operations, due to the explosive rise in the number of vehicles in use. In current traffic management systems, people violate rules by jumping traffic light signals and over speeding. Our aim is to create a license plate recognition solution to detect the license number and notify the service advisor with the vehicle's information. By integrating these functionalities, we can report defaulters and speed up the check-in process for enhance and smoother transit. This proposed solution is achieved by using GANs. The Generator accept arbitrary long sequence of geometrically registered license plate images and converts them into a high-resolution counterpart which would become the input for our Discriminator. The discriminator would try to distinguish ground truth images from the generated images which eventually helps better train the Generator model.

I. INTRODUCTION

The application of the License Plate Recognition model in intelligent transportation systems that have been installed in countries for tasks such as traffic law enforcement and traffic monitoring has piqued interest. In addition, License Plate Recognition systems are used to monitor vehicle parking entry and exit, collect toll fees, and control security measures in restricted areas such as military campgrounds and sanctuaries. These systems are often used to deter fraud and strengthen protection in specific areas.

In general, a License Plate Recognition device takes an image or video stream as input and outputs the content of the license plate, typically as text, if the given frame contains a car. A camera is used to take photographs of the vehicles in these systems. Depending on the system's specifications, those images may be colour, black and white, or infrared. The license plate is detected and read using techniques such as object detection, image processing, and pattern recognition. The size, colour, font, and standards of license plates vary widely across regions, making it difficult to provide a single global solution for detecting and recognizing a license plate.

In most electronic imaging applications, high-resolution images are favored. Since a high-resolution image has a high pixel density, it is more accurate. For instance, when cameras placed in a traffic system captures the picture of defaulters, it is often found to be blurry or in low resolution. Now in this case we cannot penalize the person at fault because we do not have the owner's vehicle details.

Similarly, in a scenario where we want to speed up the check-in process at toll booths, rather than manually typing in the license details if we have the clear high-resolution image of the license plate, we won't need to manually type in and again in some cases cross check the characters. To provide a solution to aforementioned issues and problems similar to these, we have come up with a model which would take in low resolution (LR) images of the license plate and output high resolution (HR) images which could be further used as per the requirements. GANs, or Generative Adversarial Networks, are a form of generative modelling that employs deep learning techniques such as CNNs.

II. PREVIOUS WORK

The classical approach to License Plate Recognition involves three stages: license plate detection, character identification, character recognition. A survey of works implementing the classical approach can be found in our literature review report. Here we briefly review only the recent works solving License Plate Recognition problems in an end-to-end method by using deep neural networks.

In the paper by Shh-Chieh Lin and Chiih-Ting Chen, they have used Non uniform Interpolation Method. From a deterministic point of view, using bicubic interpolation seems to be a reliable method but using an area based cross correlation means thinner digits may be completely nullified in the process. x

In the paper by Sung Cheol Park, Min Kyu Park and Moon Gi Kang, the method used is LPRnet and GANs. The LPRnet is specifically designed to train on license plate recognition and thus the CNN outperforms baseline methods. The drawback lies in GANs training instability resulting in Loss function stagnation.

III. PROPOSED WORK

Generally, a License Plate Recognition receives an image or a video stream as input and outputs the content of the license plate if the frame includes a car. The camera would capture and send the picture to the system. The image may not be clear as the quality of images taken from those cameras is not so good.

The low-resolution image would be passed onto our model. We use GANs to convert the LR image to HR image. GANs are a smart way of teaching a generative model by posing it as a supervised learning problem with two sub-models: the generator model, which we train to produce new data, and the discriminator model, which attempts to distinguish examples as real or false. The two models are trained in an adversarial zero-sum game until the discriminator model is fooled about half of the time, indicating that the generator model is producing plausible data.

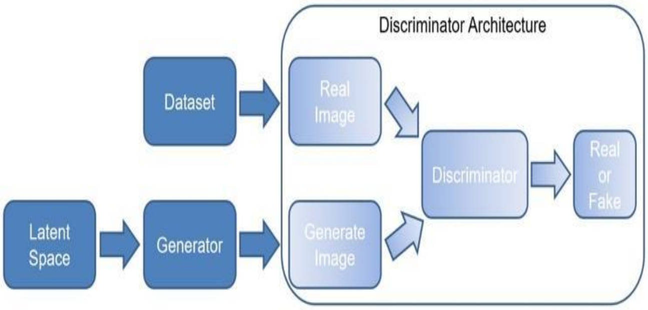

Fig 3.1 – GAN Architecture

Our generator is built on top of two models – the RRDB net and the Residual Dense Network. The generator up-samples a low-resolution image and sends it off to the discriminator. The discriminator is a network that outputs either 0 (fake) or 1 (real).

This continuous adversary between generator and discriminator helps train the model. But to keep track of the GAN model we use a content loss which calculates Mean Squared Error (MSE) in the feature space of VGG network. We also use Peak Signal to Noise ratio and Structural similarity index measure to constraint the output space to letter-like reconstructions only and avoid over-smoothening of characters that differ from actual characters.

IV. IMPLEMENTATION

A. Pre-process

We load in the dataset. Convert the jpeg file format to float so that we can work on it. Introduce random brightness and contrast to the image for image augmentation. And save the files in batches of 24. Then we extract the license plate from the image, and save the resultant image. Now from these images, we filter out those with very low or high brightness and bad contrast. This process is carried out by calculating mean and standard deviation of the image. The mean provides an overall brightness characteristic of the image while the standard deviation gives contrast value.

B. Training and Validation

Training & Validation – To keep track of all the errors, losses, accuracy and other metrics we introduce logging method which is a part of TensorFlow - Keras package.

We train the generator to generate an image close to the ground truth image set. For that we calculate the MSE, entropy and the cumulative loss for the Generator system. Our aim is to minimize this loss. Similarly, for training the discriminator we use discriminator loss and discriminator accuracy function. It helps in training the discriminator better for distinguishing between the generated image and ground truth image. The validation step is the same but on a different dataset of images. And we update our log metric for all the values calculated here.

C. Loss Metric

Several metrics are used to ensure better image reconstruction. We have used Peak Signal to Noise Ratio (PSNR), Structural similarity index measure (SSIM), The PSNR is calculated over MSE and it is the ratio between maximum possible value of a signal and the power of distorting noise that affect the quality. SSIM is used for measuring the similarity between two images unlike MSE which focuses more on similarity between the pixels of 2 images. VGG19 is applied over generated images and real images. The intermediate layers of a pre-trained VGG19 network work as feature extractors and can be used to extract features of generated and real images.

$$Loss = 0.01 * \frac{1}{n}\sum_{i=1}^{n}\left\|VGG(\hat{Y}) - VGG(Y)\right\|_2 + \frac{1}{n}\sum_{i=1}^{n}\left\|\hat{Y} - Y\right\|_2$$

We calculate discriminator loss and generator loss with the help of Binary Cross-Entropy Loss function.

D. Models Used:

Generator

RRDBNet is called in a sequential fashion – a 3x3 Convolutional layer passed onto the RRDB.

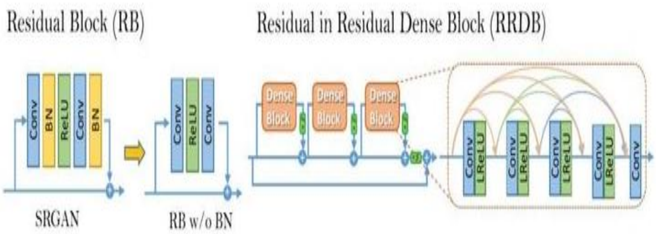

Fig 4.1 – RRDBNet Block Diagram

The RRDB block which takes in original input in addition with alpha times the output after 10 residual blocks. Then this is passed to a 3x3 Convolutional twice. Using LeakyReLU as the activation function so that the function allows small negative values when the input is less than zero. And finally, again passing it through a 3x3 Convolutional layer. Storing this result in a tensor.

Residual-to-residual dense block – LeakyReLU with a 3x3 Convolutional layer be tensor x1. Concatenating this with tensor x gives tensor x2. Repeating this gives tensor x3 with concatenated tensors x, x1, x2. Again, doing this gives the final tensor x4 with concatenated value of x, x1, x2, x3.

The output is the convolution of all five tensors concatenated times the alpha (hyperparameter) plus the tensor from RRDB block x.

Discriminator

It has a 3x3 Convolutional layer with 64 channels. There are multiple convolution block pairs where each block is a 2x2 convolution block. Batch Normalization is performed on this layer to standardize the inputs for each batch. This helps in stabilizing the process. Again, using Leaky ReLU as the activation function with hypermeter alpha. Thus, we have 2 fully connected layers.

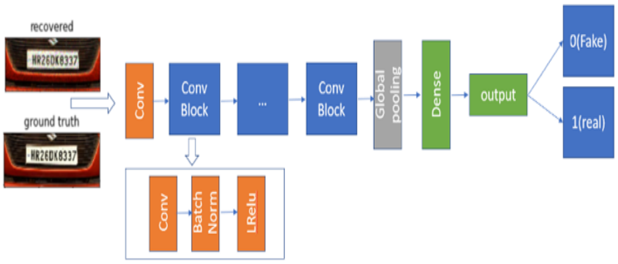

Fig 4.2 – Discriminator Architecture

Adam Optimizer is used wherever necessary with parameters, learning rate = 1e-3, beta1 = 0.9, beta2 = 0.999, epsilon = 1e-7. This helps to avoid divide by zero error.

For Generator, learning rate = 1e-4, decay rate = 0.2(exponential decay)

For Discriminator, learning rate = 1e-4, decay rate = 0.2(exponential decay)

Leaky ReLU slope = 0.2 wherever used.

V. RESULTS

The results that we have achieved so far seem to be carrying somewhat noise in them. This may or may not be due to the arbitrary values of hyperparameters wherever used. Unless for License Plate Recognition systems, this job necessitates a significant amount of time, effort, and energy. In addition, manual intervention in these tasks can result in inaccurate interpretations.

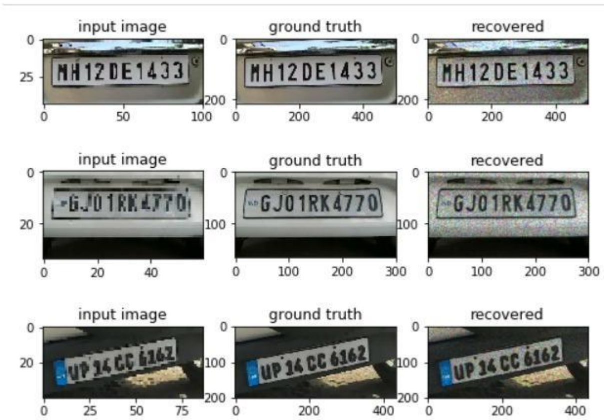

Fig 5.1

VI. CONCLUSION

Our License Plate Image Upscaling model belongs to the multi-stage license plate recognition group. Plate recognition is broken down into three levels of multi-stage license plate recognition systems: license plate detection, character identification, and character recognition.

We've demonstrated images can be upscaled inexpensively. The corresponding representation can typically be stored in a fraction of the space required by a raster image. The exceptions are pictures that are extremely complex or "busy." While doing the project, we studied multiple survey papers to understand the work done by others in this field in order to comprehend better with the project. After studying those papers, we conducted a literature survey out of those papers. After the course of designing the model and fetching the model with training and testing data, we finally conclude that the predictions made by our model were fast and efficient.

However, other considerations such as license plate rotations and occlusions restrict the applicability of much of the scheme to real-world applications. As a result, under current constraints, the technique for upscaling low-resolution images is computationally challenging.

VII. CHALLENGES AND FUTURE PROSPECTS

Various applications involving vehicle identification, such as automating parking systems and payments, tracking toll fee payments, regulating state border pass and protection measures in countries, traffic management, and law enforcement, use License Plate Recognition. These applications benefit the community by reducing crime and fraud, reducing manpower requirements, and ensuring security. Despite the fact that these systems have improved significantly, there is still a need for a more robust and reliable system to cope with various types of license plates and changing environmental conditions. Due to the scarcity of night-time data, one of the major limitations of current systems is their poor or non-existent performance at night. Thermal Infrared images are preferred for creating a dataset for night vision because they provide more detail than a typical image taken by cameras at night. There are a lot of labelled RGB datasets for License Plate Recognition, but there aren't many datasets with Thermal infrared photos. Furthermore, the location of the camera and its hardware requirements have an impact on the Recognition system's performance. Furthermore, when the gap between the vehicle and the camera is large, the license plate number loses too much data, causing even the high-resolution model to fail.

REFERENCES

[1]. Shih-Chieh Lin, Chih-Ting Chen, "Reconstructing Vehicle License Plate Image from Low Resolution Images using nonuniform Interpolation Method"
[2]. Sung Cheol Park, Min Kyu Park, Moon Gi Kang, "Super-Resolution Image Reconstruction"
[3]. T. Sharmila, L. Megalan Leo, "Image Upscaling based Convolutional Neural Network for Better Reconstruction Quality"
[4]. Michael Elad, Arie Feuer, "Restoration of a Single Super resolution Image from Several Blurred, Noisy, and under sampled Measured Images"
[5]. Chao Dong, Chen Change Loy, Member, Kaiming He, Member, Xiaoou Tang, "Image Super-Resolution Using Deep Convolutional Networks"
[6]. Sven Olsen, Bruce Gooch, "Image Simplification and Vectorization"
[7]. Jithmi Shashirangana, Heshan Padmasiri, Dulani Meedeniya, Charith Perera , "Automated License Plate Recognition: A Survey on Methods and Techniques"