Towards lane detection using a generative adversarial network

Tomás Emmanuel Juárez Vallejo 1 / Sebastián Salazar Colores 2 / Juan Manuel Ramos Arreguín 1

1 Autonomous University of Queretaro, Faculty of Engineering. Queretaro, Mexico. Queretaro, Mexico
2 Optical Research Center. Leon, Mexico

Correspondence author: juan.ramos@uaq.mx

Reception: 02-12-2021 / Acceptation: 17-03-2022 / Publication: 21-11-2023
© Nova Scientia, under Creative Commons license / ISSN 2007-0705

Abstract: traffic accidents are one of the main causes of death in Mexico, the collisions are caused mostly due to human error, therefore attempts have been made to reduce these shortcomings with driver assistance systems. This paper presents a study conducted to explore the capabilities of a Generative Adversarial Network in terms of application in lane detection on a highway, it is proposed to use a metric known as Dice index which measures the similarity between images and a pre-processing method based on color spaces, as well as a technique called Superpixels which is based on clustering. Finally, the results are compared with a neural network called LaneNet developed for the TuSimple database. The results obtained from this methodology needs to be optimized with future work, however, it opens the door to possible research with this type of network.

Keywords: segmentation; machine learning; neural networks; color spaces; TuSimple; GAN

Resumen: los accidentes de tráfico constituyen una de las principales causas de muerte en México, los siniestros causados son en su mayoría por errores humanos por lo cual se ha tratado de disminuir éstos percances con sistemas de ayuda al conductor. En este artículo se presentan un estudio realizado para explorar las capacidades de una Red Generativa Antagónica en cuanto a la aplicación en la detección de carriles en una carretera, se propone utilizar una métrica conocida como índice Dice la cual mide la similitud entre las imágenes y un preprocesamiento basado en espacios de color, así como una técnica llamada Superpíxeles la cual está basada en agrupamiento. Finalmente, los resultados son comparados con una red neuronal llamada LaneNet desarrollada para la base datos TuSimple. Los resultados obtenidos a partir de esta metodología requieren ser optimizados con próximos trabajos, sin embargo, permite abrir la puerta a posibles investigaciones con este tipo de red.

Palabras clave: segmentación; aprendizaje automático; redes neuronales; espacios de color; TuSimple; GAN

1. Introduction

Traffic accidents are one of the main causes of loss of life in Mexico, being the first cause of death for people between 5 and 29 years old, and the fifth cause of mortality in the general population (INSP, 2019). In 2019 64.85% of road accidents were caused by human error, due to fatigue, distractions, speeding, driver’s physical health (INEGI, 2020). Driver assistance systems can help reduce road accidents, an example of this is a lane departure warning system, generating an alarm when it detects that the car is not driving within its designated limits.

There is a wide variety of techniques for lane detection, so it is necessary that these methods be studied in different situations to evaluate their performance in adverse conditions. Since road status are highly variable, the objective is to make sure that they are adaptable to any type of situation. With the development of new research, detection methods have been improved and the algorithms used are becoming more sophisticated. Some algorithms begin to incorporate artificial intelligence models to obtain a more reliable measurement in different conditions.

Deep Learning is a set of artificial intelligence methods that has been widely used in lane detection, and one of its most widely used methods has been Convolutional Neural Networks (CNNs). CNNs have the ability to ignore noise in the images, with the disadvantage of requiring a large database and a considerable amount of time for training. In 2017, a CNN was implemented with an Extreme Machine Learning algorithm achieving a reduction in learning time and improved performance (Kim, Kim, Jang, & Lee, 2017). In 2017 Hou et al. presented a Self-Attention Distillation algorithm which allows the CNN to learn from itself improving its performance (Hou, Ma, Liu, & Loy, 2019). Despite this advances CNNs still face the challenges of different weather conditions, shadows, variant lane sizes, etc.
Towards lane detection using a generative adversarial network

Generative Adversarial Networks (GAN) had been proposed to tackle these issues (Zhang, Lu, Ma, Xue, & Liao, 2021) (Ghafoorian, Nugteren, Baka, Booj, & Hofmann, 2019). GANs use two networks, a generator whose job is to create fake images that mimic the ground truth of the input image and a discriminator that evaluates the output image to determine if it’s the ground truth or a generated image.

Before these approaches were developed a variety of techniques had been tried, some traditional technics used color spaces to mitigate noise, Chiu et al. proposed a method based on color information, The authors used a color-based segmentation to detect the lane boundary, the algorithm required low computational power and memory requirements (Chiu & Lin, 2005).

Assidiq et al. (2008) proposed a lane detection algorithm to be on both painted and unpainted road as well as curved and straight roads in different weather conditions. The experimental results prove that the proposed algorithm is robust and fast enough for real time requirements (Assidiq, Khalifa, Islam, & Khan, 2008).

Kim presented a robust lane-detection-and-tracking algorithm for complex scenarios, including a lane curvature, worn lane markings, lane changes, and others. The algorithm was based on random-sample consensus and particle filtering (Kim Z., 2008).

Li et al. proposes an algorithm with Gabor filters. Which uses the texture feature to estimate the vanishing-point locations. Also, edge detection is used to detect the road lane ahead of the vehicle. The algorithm proposed is insensitive to variations of road condition and illuminations (Li, Ma, & Liu, 2016). In 2018, Neven et al. proposed a fast algorithm for lane detection on a road, at a speed of 50 fps, and considers a variable number of lanes and lane change. He uses the TuSimple database to test the algorithm (Neven, Brabandere, Georgoulis, Proesmans, & Gool, 2018). In other works, Zhang et al. established a multiple task learning framework to segment lane areas and detect the lane boundaries in a simultaneous way. A novel loss function with two geometric constraints is proposed. The framework is evaluated on the KITTI dataset, CULane dataset and RVD dataset (Zhang, Xu, Ni, & Duan, 2018).

Yang et al. proposed a lane detection network inspired by the LaneNet model, using a semantic segmentation concept with multiple level features on the encoder. By the reduction of computation in decoders, the algorithm uses the multilevel features to precisely predict the high-quality lane maps. The authors are using TuSimple and CuLane datasets to evaluate the model (Yang, Cheng, & Chung, 2019).

Sometimes, the lanes on the road are curved. Dorj et al. (2020) presents a cutting-edge curve lane detection algorithm based on the Kalman filter and is tested with a self-driving car and the simulation software: Gazebo. The experimental results show an average of 10 frames per second (Dorj, Hossain, & Lee, 2020). In other works, a lane boundary marker network is used to detect key points in the lane, road geometry is used to detect markers of lines, curves and to predict missing lines on the road. The authors are using TuSimple datasets to test the algorithm, with 95.5% of accuracy in TuSimple (Khan, et al, 2020).

The Muthalagu et al. (2020) work is focused on demonstrating a powerful end-to-end lane detection algorithm, applying computer vision contemporary techniques for self-driving cars. They propose an improved lane detection technique based on perspective transformations and histogram analysis. The algorithm can detect both straight and curved lane lines. The results are presented with several simulation environments. The processing time is 63.65 ms (Muthalagu, Bolimera, & Kalaichelvi, 2020).

2. Methods, techniques, and instruments

Our proposal is based on one of the most powerful Conditional Generative Adversarial Networks (CGAN’s) model: the Pix2Pix model. In this section, we develop the theoretical background necessary to understand our proposal, by briefly outlining terms such as Generative Adversarial Networks (GAN’s), CGAN’s, and the Pix2Pix model. Finally, the architecture of our proposal is described.

2.1. Generative Adversarial Networks (GAN’s)

Generative models use two deep neural networks: a Generator (\( G \)) and a Discriminator (\( D \)). These two networks are adversarial, i.e., they “play” a zero-sum game in which where one network wins, the other loses.

\[
\min_G \max_D V(D,G) = \mathbb{E}_{data(x)}[\log D(x)] + \mathbb{E}_{z \sim p(z)}[\log (1 - D(G(z)))]
\]  

(1)
Where $p_{z(z)}$ are input noise variables distributions and $p_{data(x)}$ are examples of a probabilistic distribution to learn by $G$. The Discriminator ($D$) is part of the training process by classifying in real or fake $x$ and $z$. Its Loss function is given by:

$$L_{GAN}(G, D) = log D(x) + log \left(1 - D(G(z))\right)$$

(2)

2.2. Conditional Generative Adversarial Networks (CGAN's)

A variant architecture of GANs where a prior condition $y$ is set is called Conditional Generative Adversarial Networks (CGANs). Its loss function is composed by two losses: a GAN Loss $L_{CGAN}$ and a L1 loss $L_{L1}$ given by:

$$L_{GAN}(G, D) = log D(x, y) + log \left(1 - D(G(x, y), y)\right)$$

(3)

$$L_{L1}(G) = |y - G(x, y)|$$

(4)

Finally,

$$L_{CGAN} = L_{CGAN}(G, D) + \lambda L_{L1}(G)$$

(5)

Where $\lambda > 0$ is the importance $L_{L1}$.

2.3. The pix2pix model

The Pix2Pix model it’s conformed by a Generator based in an U-net network and a convolutional PatchGAN proposed as discriminator, this network has been used in a range of scenarios for example: Labels to Street Scene, Labels to Facade, Aerial to Map, etc. (Isola, Zhu, Zhou, & Efros, 2016). A representation of the Pix2Pix GAN model can be seen in figure 1.

![Simple GAN model](image)

Figure 1. Simple GAN model.

These networks are defined as:

Generator: The architecture of the generator it’s a modified U-net which consists in an encoder-decoder network that shares information between layers. The encoder of the network it’s given by a down sample block and the decoder by an up sample block. The down sample is conformed by the layers: a 2D Convolution, Batch Normalization and Leaky ReLU as the activation function, the decoder layers are a Transposed Convolution, Batch Normalization and a ReLU as activation, a dropout is applied in the first 3 layers. A representation of the Generator can be seen in figure 2.

Discriminator: The discriminator is designed for a high frequency structure using the L1 loss to correct the lower frequencies. The discriminator needs to restrict the attention of the model in local image patches, it tries to classify
each patch to determine if the image is real or fake, to obtain the final result it averages all patches. In figure 3 the discriminator architecture is illustrated. The discriminator receives two inputs, the ground truth, and the predicted image, it uses a block composed of a Convolution, Batch Normalization and Leaky ReLU as the activation function, the batch size is 30 x 30 pixels.

Figure 2. Pix2Pix Generator, In which the yellow block represents the Input Layer, the red blocks represent the down sample, the green the up sample and the blue is a convolutional transpose layer.

Figure 3. Discriminator model. Were the first two blocks representing the Input layers, the next one represents a concatenate layer, then the colors represent different layers, green represents a Sequential layer, blue it’s Zero Padding 2d, black represents a Convolution layer and yellow denotes Batch Normalization.
2.4. Dataset Description

The dataset used was *TuSimple* (TuSimple, 2017) which consists of video clips taken on roads with a resolution of 1280 x 720, some of its characteristics are: varying weather states, different traffic conditions and different number of lanes. It has 3626 training images and 2782 test images. The labels for the lane lines are in a JSON file. In order to train the Pix2pix model it requires an image consisting in two parts: the ground truth and the input image, to fulfill this requirement an image with the lanes was created from the dataset labels with a resolution of 1280 x 720 pixels and a line size of 12 pixels, which was then concatenated with the original image, after that the images were resized to a resolution of 1024 x 256, an example of this can be seen in figure 4.

![Image](image.png)

*Figure 4. Training image, where: A is the Ground truth, and B the Input image.*

2.5. Comparative Model

*LaneNet* is a model proposed in (Wang, Ren, & Qiu, 2018) with the purpose of detecting lanes in roads, it divides the task in two parts: lane edge proposal and lane line localization. The neural networks used are a Convolutional Neural Network (CNN) to create the segmentation and an iterative procedure for clustering the lanes. It was trained in the *TuSimple* dataset.

The approach in this paper was to propose the problem of detecting lanes as a segmentation instance divided into lanes and backgrounds instead of each lane having its own class, the authors proposed two networks: *H-net* and *LaneNet*. *H-net* it is used to estimate the ideal perspective transformation for the input image after that *LaneNet* creates the binary and instance segmentation, finally a clustering algorithm it’s applied to the binary and instance segmentation to create the lane lines and give them a unique class, a representation of this network is indicated in figure 5.

The results obtained in this paper were a 96.4% accuracy in the *TuSimple* official metrics, the accuracy in the dataset is calculated with the following formula:

$$ acc = \sum \frac{TP}{GT} $$

(6)

Where TP are the number of correct points and GT the number of ground truth points.
Towards lane detection using a generative adversarial network

2.6. Image Pre-processing
For the image pre-processing we proposed 3 different methods, two of them are changes in the color space such as HSV (Hue, Saturation, Value) and CIELAB, the third method is a clustering algorithm known as Superpixel. The color spaces have been used in traditional lane detection models since the lanes are generally painted in yellow or white and it helps to highlight the desired colors and mitigate lighting changes in lanes (Muthalagu, Bolimera, & Kalaichelvi, 2020).

The HSV Color space is an alternative representation of the RGB model and it's based on the human perception of color, it divides RGB into hue, saturation and value, it’s been used in computer vision because the hue component its effective and insensitive to shadows (Huang, Kong, Li, & Zheng, 2007). CIELAB is a color space that describes all the colors visible to the human eye, it conveys colors in L which is the perceptual lightness and a and b which are the combination of the colors: red and green for a and blue and yellow for b, this can help mitigate the changes in illumination (Karavaev & Al-Naim, 2020).

The Superpixel algorithms are a clustering method to address the computational power of processing an image, it clusters the pixels that have similar values then it averages the value between them and creates a simpler image, despite its loss of information it has been used in computer vision in problems like: tracking, 3d reconstruction, object detection, etc. (Wang, Liu, Gao, Ma, & Soomro, 2017). An example of the Superpixel algorithm is illustrated in figure 6.

2.7. Metrics used
The metrics known as Jaccard Index and Dice Coefficient, measure the similarity between images in a range from 0 to 1, they have been used in medical imaging and diverse computer vision fields, some of its uses are: evaluate the segmentation of an image, measure the inpainting of an algorithm, obtain the similarity of two sets of data, etc. (Amirkhani & Bastanfard, 2021) (Lee, 2017). Taking in consideration that the GANs try to recreate the Ground Truth learning the features from the input images we decided to measure the similarity of the predicted image with these metrics. The Jaccard index is defined as the intersection of the sets divided by the union of the sets. It is defined by the following formula (Real & Vargas, 1996):
The Dice Coefficient or Sørensen–Dice coefficient (SDI) is defined as 2 times the intersection of the sets divided over the sum of the sets; it is defined by the following formula (Carass, et al., 2020):

\[
SDI(A, B) = \frac{2|A \cap B|}{|A| + |B|}
\]  

3. Results and discussion

The baseline results were obtained by evaluating 400 images corresponding to the test dataset, this was applied to LaneNet and the Pix2Pix model, the number of images was decided upon hardware limitations. A simple post-processing was applied to the images before obtaining the metrics which consisted in binarizing the output image. For the baseline in LaneNet the original trained weights were used, for the Pix2Pix baseline model the training was made without pre-processing and without trained weights this was applied also to the change in color spaces, in each test the model was retrained. The baseline results are shown in Table 1.

<table>
<thead>
<tr>
<th>Baseline Model</th>
<th>Jaccard Index</th>
<th>Dice Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pix2Pix</td>
<td>0.348</td>
<td>0.499</td>
</tr>
<tr>
<td>LaneNet</td>
<td>0.607</td>
<td>0.753</td>
</tr>
</tbody>
</table>

The experimental results were obtained the same way as the baseline but the pre-processing for the input images were changed to evaluate which one delivered better results. The experimental results are shown in Table 2 and the predictions of the network are in the figure 7 for the baseline, figure 8 for the CieLab pre-processing and figure 9 for the Superpixel pre-processing.

<table>
<thead>
<tr>
<th>Pre-processing (Pix2Pix)</th>
<th>Jaccard Index</th>
<th>Dice Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>HSV color space</td>
<td>0.267</td>
<td>0.401</td>
</tr>
<tr>
<td>CieLab color space</td>
<td>0.356</td>
<td>0.514</td>
</tr>
<tr>
<td>SuperPixel Algorithm</td>
<td>0.322</td>
<td>0.482</td>
</tr>
</tbody>
</table>
Towards lane detection using a generative adversarial network

Figure 7. Baseline Results, where A is the input image; B, the ground truth, and C, the predicted image.

Figure 8. CieLab Results, where A is the input image; B, the ground truth, and C, the predicted image.
4. Conclusions
In the present work, we used a general-purpose GAN to detect lanes in the dataset TuSimple, by using different color spaces and the superpixels algorithm. Even though the network delivered results with all the color spaces only with the CIELAB color space, the network showed improvements over the baseline in both metrics, neither HSV nor the Superpixel algorithm granted the model an edge over the baseline model, they even caused a decline in the results for both metrics as seen in Table 2. The network still needs further work to be able to compete with state-of-the-art models. Despite this, we were able to prove that the Pix2Pix model can learn lanes features as seen in figures 7, 8 and 9. And that changing the color space in the input images can help the network in the training process, but it can also worsen it. This work can be further extended by using different architectures in the u-net base model as well as different loss functions.

5. Supplementary information
There is no additional information.

Author’s ORCiD
Tomás Emmanuel Juárez Vallejo1 0000-0002-8601-2605
Sebastián Salazar Colores2 0000-0002-6353-0864
Juan Manuel Ramos Arreguín1 0000-0002-2604-9692

Author’s contribution statement
Tomás Emmanuel Juárez Vallejo: Carried out an investigation in the state of the art and developed the experiments carried out in this work. Sebastián Salazar Colores: Gave support in the investigation of the state of the art, as well as contributed in the technical development of the neural network. Juan Manuel Ramos Arreguín: Gave support in structuring the project and the investigation, he carried out an investigation in the models of neural networks and their uses.

Conflict of interest declaration
The authors declare that there is no conflict of interest.
Towards lane detection using a generative adversarial network

References
Wang, Z., Ren, W., & Qiu, Q. (2018, 7 4). LaneNet: Real-Time Lane Detection Networks for Autonomous Driving.

Nova Scientia, 15(31), 1-11
ISSN 2007-0705
10