

# Procesamiento de imágenes utilizando Deep learning y Carla Simulator

## Resumen—

### I. INTRODUCCIÓN

### II. MARCO TEORÍCO

#### II-A. Deep Learning

Las redes neuronal son muy utilizadas en el campo de la minería de datos, debido a su gran versatilidad. Dentro de este mundo existen 2 grandes tipos de entrenamientos para los supervisados que dada una entrada  $X$  se conoce la salida deseada  $Y$  y otros casos los algoritmos no supervisados como los modelos de Q-learning o los mapas autoorganizados (SOM). En este proyecto se opto por un entrenamiento supervisado.

Para el modelado de la red neuronal se utilizó un modelo llamado Pix2Pix [4]. El cual pertine dada una imagen de entrada  $X$  obtener una imagen de salida  $Y$  haciendo uso de una arquitectua U-Net [3].

Este modelo cuenta con 2 redes neuronales una denominada el generador en cual recibe la imagen  $X$  y debe ser capaz de aprender a transformala en la imagen  $\hat{Y}$ , la cual es una aproximación de la imagen  $Y$ . Y por otro lado el discriminador el cual debe tomar la imagen  $\hat{Y}$  para lograr reconocer las partes de esta que sean erroneas. La salida del discriminador debe ser una imagen resucida en tamaño que cuente con un unico canal de color, siendo esta imagen una cuantificacion del error de la red generativa.

Este problema desemboca en una pelea entre en generador ya que debe aprender a engañar al discriminador y el discriminador que debe lograr siempre reconocer que la imagen  $\hat{Y}$  es totalmente falsa, ya que es una creación de la red generativa.

#### II-B. Preporcesado de los datos

Un paso fundamental para trabajar con algoritmos de imágenes es pasar del dominio  $[0; 255]$  a un dominio mas acotado, para este trabajo se utilizo el domino de  $[-1; 1]$ . De esta forma se logra una mayor eficiencia computacional.

Debido a la gran variedad de dimensiones de las imágenes es recomendable hacer un resize a dimensiones mas pequeñas para disminuir el uso de RAM cuando son cargadas en memoria y el uso de GPU al realizar las convoluciones. En este caso las imágenes de entrada y de salida son redimensionadas a  $256 \times 256$ .

### III. DESARROLLO

#### III-A. Obtención del dataset

Para obtener el dataset se colocaron una camara RGB y una camara de segmentación semántica [2], ubicadas en la misma posición. Se recolectaron los datos cada 5 segundos.

Las condiciones climaticas utilizadas fueron aleatorias, lo que permite obtener un dataset mucho mas variado.

Para lograr un manejo autónomo del vehiculo se utilizó el autopilot [1]. Con la finalidad de aumentar la variedad de imagenes se utilizaron todos los mapas posibles y se adicionaron otros vehiculos autónomos.

#### III-B. Entrenamiento de la red

Como fue mencionando en el marco teoríco el modelo implementado se denomina Pix2Pix [4]. Se utilizó como entrada la imagen de la camara RGB y la salida será una imagen segmentada, para ello se utilizó el dataset que contava con casi 3000 imágenes.

Todas las imágenes fueron llevadas al dominio  $[-1; 1]$  como es nombrado en la

El dataset fue dividido en 2 partes una utilizada para el entrenamiento, aproximadamente el 80 % de las imágenes y otra parte para el testing de la red utilizando el 20 % restante.

Luego de  $N$  iteraciones se presentan los resultados en la fig.1

Figura 1. Imágenes obtenidas luego del entrenamiento

Las perdidas de la red generativa y del discriminador fueron recolectados para lograr mostrar con mayor facilidad los avances del modelo (ver fig.2-3)

Figura 2. Error del discriminador en función de la iteración con el dataset de testing.

Figura 3. Error del generador en función de la iteración con el dataset de testing.

### IV. CONCLUSIONES

#### REFERENCIAS

- [1] "Carla api documentation," [https://carla.readthedocs.io/en/latest/python\\_api/](https://carla.readthedocs.io/en/latest/python_api/).
- [2] "Carla sensors reference," [https://carla.readthedocs.io/en/latest/ref\\_sensors/](https://carla.readthedocs.io/en/latest/ref_sensors/).
- [3] "U-net architecture," <https://en.wikipedia.org/wiki/U-Net>.
- [4] Phillip.I, J.Zhu, T.Zhou, and A.A.Efros, "Image-to-image with conditional adversarial networks," 2018.