

## Informe de progreso I

### Problema a resolver.

La idea inicial del proyecto es realizar un coche que pueda realizar una conducción autónoma. Obviamente esta propuesta escapa a las dimensiones de un trabajo de final de grado por recursos, tiempo y complejidad. Por este motivo he buscado una alternativa que me permita obtener conocimientos sobre ese tipo de proyecto y que se adapte a un proyecto de final de carrera. Para ello mi idea es realizar un clasificador de señales.

### Objetivos

Se han añadido tres objetivos más para definir mejor el abasto del trabajo.

- 1.- Generar un algoritmo para poder clasificar señales y ver el rendimiento del sistema entrenado para diferentes casos.
- 2.- Entrenar diferentes redes neuronales y estudiar cómo obtener mejores resultados.
- 3.- Implementar una red neuronal que detecte señales de tráfico en un entorno controlado.
- 4.- Estudiar el funcionamiento de la red neuronal en entornos sin controlar (salvajes).

### Planificación.

Algunas tareas se han visto modificadas debido a el diferente enfoque obtenido del trabajo en el primer mes de desarrollo. La explicación de dichas modificaciones y sus motivos viene en el siguiente punto metodología.

Las tareas planificadas actualmente son:

- 1.- Obtener documentación más profundamente.
- 2.- Escoger base de datos de señales de tráfico.
- 3.- Entrenar red neuronal 1.
- 4.- Entrenar red neuronal 2.
- 5.- Entrenar red neuronal 3.
- 6.- Comparar rendimiento de las redes neuronales utilizando porcentajes de precisión.
- 7.- Comprobar el funcionamiento de las redes detectando fotografías de señales de tráfico con fondo blanco.
- 8.- Comprobar el funcionamiento de las redes detectando fotografías de señales de tráfico con fondos complejos.

9.- Desarrollar aplicación para comprobar el funcionamiento de las redes mediante la grabación con la cámara de un dispositivo. Esta comprobación se hará tanto en entornos amigables como en entornos salvajes.

10.- Escoger la mejor red neuronal.

11.- Conclusiones

Calendario:

Todas las tareas tienen dependencia entre ellas, por lo tanto, hasta que no se finaliza la que se está realizando no se puede pasar a la siguiente.

Marzo		Objetivos cumplidos
Semana 1	1	✓
Semana 2	1, 2	✓
Semana 3	3	✓
Semana 4	3	✓
Abril		
Semana 1	4	X
Semana 2	4	✓
Semana 3	4, 5	✓
Semana 4	5	✓
Mayo		
Semana 1	5	✓
Semana 2	6, 7	✓
Semana 3	8	✓
Semana 4	9	✓
Junio		
Semana 1	9	
Semana 2	10, 11	
Semana 3		
Semana 4		

Al ver el calendario podemos ver dos semanas vacías, para poder realizar informes y presentaciones. También estas semanas se pueden utilizar por si no se ha logrado completar alguno de los objetivos semanales tener la holgura para poder completarlos todos.

### Metodología i desenvolupament.

Inicialmente se buscará una base de datos para poder entrenar la red neuronal, utilizando 70% de los objetos como train i 30% como test, pudiendo variar estos porcentajes según se obtengan mejores o peores resultados.

La base de datos no reconocerá todas las señales ya que será imposible atacar todas ellas debido a la gran variedad de señales que existen, esta base de datos estará clasificada por diferente tipo de señales dependiendo de su forma, i dentro de cada tipo de señal tendrá una

etiqueta con el nombre de la señal. La base de datos dispone de 62 señales que son las siguientes:



Finalmente se ha desestimado la generación del robot debido a falta de tiempo y a el enfoque del trabajo en obtener la mejor clasificación de señales posible, el comportamiento del clasificador se realizará mediante otros métodos.

El entrenamiento de las redes neuronales se está realizando, utilizando Python y las funciones proporcionadas por Tensorflow. Inicialmente se había estimado el uso de tres redes neuronales diferentes. Se han generado estas tres redes neuronales y finalmente se ha estimado realizar una cuarta para obtener un nuevo enfoque.

La primera red neuronal, bautizada como Dense, utiliza una capa de entrada Flatten, seguida por dos capas ocultas Dense con 150 neuronas cada una y con una activación relu. Finalmente, una capa de salida la cual tiene el mismo número de neuronas que los resultados posibles de la red neuronal, es decir 62. Esta ultima capa utiliza una activación softmax la cual genera un numero del 0 al 61 y el valor que tenga más próximo es el resultado de la clasificación. El funcionamiento de esta red neuronal es simple, utiliza los pixeles de una fotografía para compararlo con los pixeles de otra y así poder clasificar las señales que observa.

En segunda instancia se ha realizado una segunda red neuronal, llamada CNN. Utiliza tres capas ocultas Conv2D que utiliza mascarar para poder tratar las imágenes y así poder generar desenfoques, afilamientos o detectar bordes. A continuación, utiliza MaxPool2D para reducir el tamaño de la imagen y así realizar las entradas de la siguiente capa Conv2D con diferentes posiciones de la imagen. Después de realizar este proceso 3 veces se aplica una capa de entrada Flatten seguido de una Dense con 100 neuronas con activación relu, imitando el funcionamiento de la red neuronal Dense pero habiendo realizado un tratamiento anterior. Finalmente para obtener la salida se ha vuelto a aplicar softmax. El funcionamiento de esta red neuronal, es aprenderse formas para después poder relacionarlas y poder clasificar las imágenes.

La tercera red neuronal, llamada CNN2. Tiene el mismo comportamiento que la CNN però aplicando un Dropout para eliminar periódicamente algunas neuronas y variar su comportamiento a la hora de clasificar. También previamente se han tratado las imágenes para realizar data augment y obtener imágenes con diferentes dimensiones, torcidas o volteadas.

Una vez realizadas las 3 redes neuronales se han comparado las métricas obtenidas mediante gráficos donde se pueden visualizar los resultados de entrenamiento y se ha estimado finalmente realizar una cuarta red neuronal aún por definir para lograr obtener mejores métricas y profundizar un poco más en las diferentes redes neuronales existentes.

Actualmente se está creando una aplicación en la que se pueda comprobar el resultado en tiempo real, para visualizar el funcionamiento de las redes neuronales en tiempo real.

Para acabar de ver el comportamiento de las redes neuronales en las próximas semanas se recopilarán fotografías tanto con fondos blancos como fotografías salvajes que puedan confundir al clasificador de señales y se realizara una comparativa entre los resultados de la clasificación.

Finalmente, con todos los datos extraídos se hará un estudio y se decidirá cual es la red neuronal que nos proporciona un mejor clasificador y se sacaran las conclusiones oportunas.

### Resultados provisionales

En cuanto a los resultados obtenidos de las redes neuronales, a falta de realizar una ultima y algunas pruebas más, aparte de sacar las graficas definitivas. Podemos hacer un breve resumen de el comportamiento de cada una de las redes.

Primero de todo la red Densa, así es como la mencionamos el apartado anterior. Encontramos una red con un accuracy muy alto, del 100%. Però en verdad no podemos fiarnos de este valor ya que la red se aprende los pixeles de las imágenes y después los compara con los pixeles de la imagen a clasificar y así obtiene la clasificación. Por lo tanto, en el momento que la red encuentra una imagen torcida o deformada no sabrá predecir el valor.

La segunda red, la CNN, encontramos que esta si que es capaz de diferenciar formas y así poder clasificar las imágenes según la relación de formas y llegar a tener un accuracy del 100%. A falta de realizar más pruebas parece que es una red neuronal que será bastante fiable y obtendrá bastante rendimiento.

Finalmente la última, la CNN2, tiene el mismo comportamiento que la CNN. Pero al haber utilizado data augment, puede diferenciar más fácilmente las formas al haber obtenido los datos de imágenes modificadas. El problema de esta red es que obtenemos un accuracy del 85% debido a la diferencia entre las formas y la escasa cantidad de datos en la base de datos a entrenar. Si obtuviéramos una base de datos más grande o aumentáramos la que disponemos actualmente esta red sería la mejor hasta el momento sin ningún tipo de dudas. Esta pendiente de realizar pruebas aumentando el numero de neuronas y de capas de entrenamiento para ver si se puede solucionar la escasez de datos.

### Conclusiones

Como conclusiones podemos añadir que la aplicación realizada obtiene buenos resultados y es un primer paso para poder realizar uno de los muchos módulos necesarios para generar una conducción autónoma 100% fiable. Però para poder generar esta conducción autónoma seria necesario ampliar las funcionalidades de la aplicación para poder encontrar las señales de tráfico en una imagen, poder enfocarlas y realizar una clasificación des de ese punto.

Bibliografía de referencia consultada i fuentes complementarias.

- 1.- <https://programmerclick.com/article/9213796567/>
- 2.- [https://es.mathworks.com/help/matlab/import\\_export/convert-between-image-sequences-and-video.html](https://es.mathworks.com/help/matlab/import_export/convert-between-image-sequences-and-video.html)
- 3.- <https://es.mathworks.com/help/images/image-analysis.html>
- 4.- <https://www.tensorflow.org/install/pip?hl=es-419>
- 5.- <https://www.tensorflow.org/guide/keras?hl=es-419>
- 6.- <https://rubialesalberto.medium.com/conducci%C3%B3n-aut%C3%B3noma-reconocimiento-de-se%C3%B1ales-de-tr%C3%A1fico-con-tensorflow-con-ann-y-cnn-desde-cero-5b563a0e5d8a>