

---

# Predicción de movimiento de trayectoria de peatones utilizando Deep Learning

---

Silvy Peñaloza  
Guatemala, 2022  
silvyagalileo.edu

## Abstract

1 La predicción de la trayectoria futura del movimiento de peatones es una tarea  
2 relevante y desafiante en el área de inteligencia artificial que cuenta con diversas  
3 aplicaciones, que van desde sistemas de vigilancia de multitudes hasta sistemas de  
4 toma de decisiones en la conducción autónoma. El objetivo de este estudio es la  
5 evaluación de tres modelos basados en redes neuronales aplicados a la predicción  
6 de trayectorias futuras de peatones sobre el dataset Stanford Drone Dataset, SDD.  
7 Nuestros resultados al evaluar los tres modelos, red neuronal multi-capas (MLP), red  
8 neuronal convolucional (CNN) y red Long Short-term memory (LSTM)) muestran  
9 que la arquitectura basada en redes recurrentes LSTM tiene mejor rendimiento,  
10 esto utilizando Average Displacement Error (ADE) y Final Displacement Error  
11 (FDE) como métrica de rendimiento.

## 12 1 Introducción

13 Cuando las personas se movilizan en espacios concurridos estas siguen ciertas reglas de sentido  
14 común y etiquetas sociales. Diversos esfuerzos se han realizado para entender cómo otros humanos  
15 interactúan en diferentes espacios físicos, ya sea con otros humanos, ciclistas, patinadores o vehículos.  
16 Dichos esfuerzos se basan en la observación histórica del movimiento con el objetivo de encontrar  
17 patrones que puedan ser utilizados en la investigación del comportamiento humano y en la creación  
18 de modelos que formen parte de sistemas inteligentes que interactúen directamente con personas.

19 La complejidad de las interacciones humanas en espacios concurridos es compleja, existiendo diversos  
20 factores que afectan las decisiones que un peatón puede tomar, por ejemplo la cantidad de objetos en  
21 movimiento, las dimensiones del espacio o el destino final.

22 El objetivo de este paper es evaluar diferentes técnicas para la predicción de trayectorias de peatones,  
23 utilizando un conjunto de datos reales de trayectorias de peatones en un campus universitario,  
24 delimitando el problema al análisis de una escena sin tomar en cuenta la interacción con otros objetos.  
25

## 26 2 Definición del problema

27 En este documento, se propone un modelo para predecir la trayectoria futura que puede tomar un  
28 peatón, conociéndose el histórico de movimiento de dicho peatón. La trayectoria histórica del peatón  
29 comprende 8 instantes de tiempo (time step) iniciales con las coordenadas de la ubicación del peatón.



Figure 1: Escena de estudio.

30 Esta trayectoria es utilizada como entrada para el modelo que es capaz de predecir los siguientes 12  
 31 instantes de tiempo que corresponden a las 12 ubicaciones que tomará el peatón.

32 Por simplicidad en la evaluación de los modelos propuestos no se toma en cuenta la interacción de  
 33 cada peatón con otros peatones o con otros objetos para predecir el movimiento futuro.

### 34 3 Método

#### 35 3.1 Datos

36 El dataset utilizado para este estudio es Stanford Drone Dataset, SDD, [4] es un conjunto de datos a  
 37 gran escala de imágenes y videos de vistas aéreas, con diferentes clases de objetivos como peatones,  
 38 ciclistas, carros y buses, del campus universitario. Este conjunto de datos cuenta con diferentes  
 39 escenas, es decir, tomas de imágenes de diferentes escenarios, calles horizontales, calles con redondel,  
 40 calles con intersecciones, entre otras.[2]

41 Del dataset original únicamente se tomaron en cuenta los datos que corresponden a los peatones y la  
 42 escena bookstore. La unidad de medida de los instantes de tiempo se encuentra en metros, y la escena  
 43 de estudio cuenta con un área aproximada de 2481.89 metros cuadrados. Se utilizó la información de  
 44 1772 trayectorias de peatones para el entrenamiento de los modelos y 432 trayectorias de peatones  
 45 para pruebas.

#### 46 3.2 Estructura de la solución

47 Debido a que el problema presentado, es un problema de predicción de series de tiempo, las redes  
 48 neuronales recurrentes se adaptan bien para encontrar una solución, por lo que se propone un modelo  
 49 utilizando una red recurrente Long Short Term Memory (LSTM).

50 También se presentan 2 modelos para solucionar este problema, el primer modelo se construye con  
 51 una red neuronal multi-capa (MLP), y el segundo con una red neuronal convolucional (CNN). Estos  
 52 modelos se utilizan como línea base para poder comparar el comportamiento del modelo propuesto  
 53 que utiliza una red LSTM. [1]

#### 54 3.3 Red Neuronal Multi-capa, MLP

55 Se propone utilizar una red neuronal multi-capa, MLP, para crear un modelo que sirva de base para  
 56 evaluar el comportamiento del tercer modelo propuesto para la predicción de las trayectorias de los  
 57 peatones.

58 La estructura del modelo está compuesta por una capa de entrada de 16 dimensiones, con 100  
 59 unidades y activación relu. Como salida utiliza una capa de 24 unidades correspondiente a las 12  
 60 posiciones a predecir, correspondiente a las coordenadas x,y en la escena de estudio.

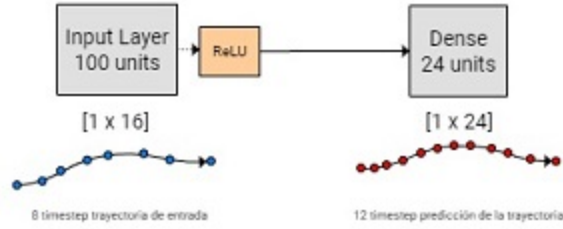


Figure 2: Arquitectura red neuronal multi-capa, MLP.

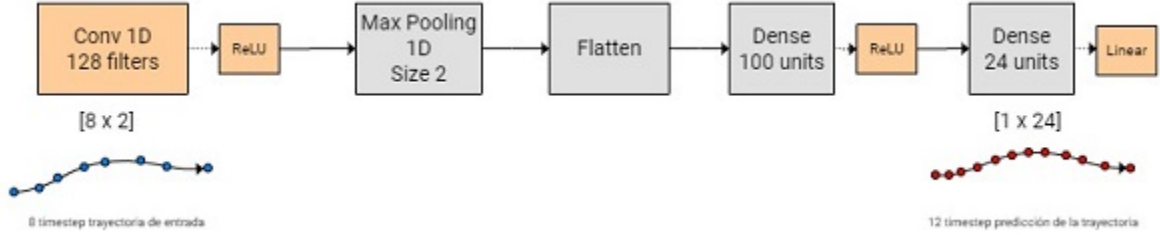


Figure 3: Arquitectura red neuronal convolucional, CNN.

### 3.4 Red Neuronal Convolucional, CNN

Se propone un modelo con una red neuronal convolucional, CNN, para la predicción del movimiento de la trayectoria de los peatones. Este modelo se utiliza como línea base para evaluar el comportamiento del tercer modelo propuesto para la predicción de las trayectorias de los peatones.

La arquitectura de este modelo utiliza una capa convolucional 1D [3] con 128 filtros, kernel de tamaño 2 y activación relu, posterior a esta capa, cuenta con una capa de max pooling 1D de tamaño 2 y una capa flatten. Una capa densa de 100 unidades y una capa de salida de 24 unidades que corresponden a las 12 posiciones a predecir, es decir 12 posiciones, con coordenadas x, y en la escena de estudio.

### 3.5 Red Neuronal Recurrente Long Short Term Memory, LSTM

Se propone un tercer modelo utilizando una red neuronal recurrente, Long Short Term Memory, LSTM para predecir la trayectoria futura de peatones.

Se infiere a que la predicción de movimiento de trayectorias es un problema secuencial, sea este modelo con el que se obtenga mejores resultados al momento de realizar nuevas predicciones.

La arquitectura del modelo propuesto cuenta con una estructura encoder-decoder multivariable. Utiliza una capa de entrada con dimensiones de 8x2, para el encoder una capa LSTM, de 512 unidades y un vector latente de 12 unidades como salida. Para el decoder utiliza una capa LSTM de 512 unidades y una capa Time Distributed con dos unidades de salida para poder obtener las 12 salidas con las coordenadas x,y de las trayectorias correspondientes a las coordenadas en la escena.

## 4 Experimentos

### 4.1 Métricas

Para evaluar el comportamiento de los modelos propuestos se utilizó como métrica de rendimiento Average Displacement Error (ADE), y Final Displacement Error (FDE).[5]

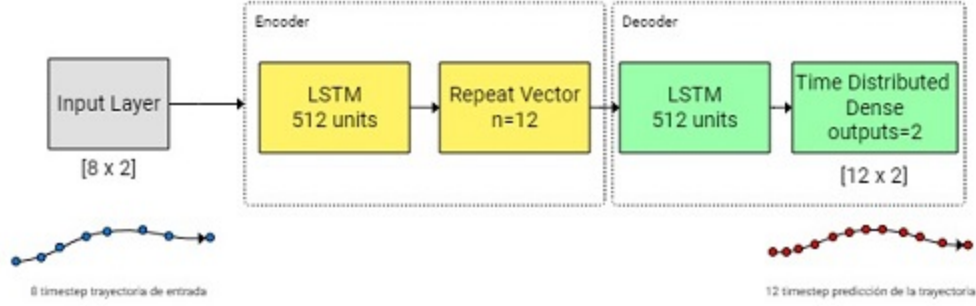


Figure 4: Arquitectura red neuronal recurrente, LSTM.

$$ADE = \frac{\sum_{i=1}^n \sum_{t=T_{obs}}^{T_{pred}-1} \|\hat{Y}_t^i - Y_t^i\|}{n(T_{pred} - T_{obs})}$$

$n$  numero de peatones

$\hat{Y}_t^i$  posición de la predicción del peaton  $i$  en el tiempo  $t$

$Y_t^i$  posición futura real del peaton  $i$

$\|\cdot\|$  norma euclideana

Figure 5: Average Displacement Error, ADE.

#### 4.1.1 Average Displacement Error, ADE

Es la distancia euclideana promedio sobre todos los puntos obtenidos de la predicción y todos los puntos reales de las trayectorias, de todos los peatones. La ecuación para calcular esta metrica se muestra en Figure 5.

#### 4.1.2 Final Displacement Error, FDE

Es la distancia euclideana promedio entre la ultima posición de la predicción y la ultima posicion real, de todos los peatones. La ecuación para calcular esta metrica se muestra en Figure 6.

#### 4.2 Red Neuronal Multi-Capa, MLP

Se realizaron algunos experimentos con las diferentes métricas utilizando una red neuronal multi-capa, MLP. En la Table 1 y Table 2, se muestran los resultados obtenidos de los expementos. Como se puede observar, se obtuvo mejor resultado, tanto para el conjunto de datos de entrenamiento como de prueba, utilizando una capa de 100 unidades, 200 épocas, learning rate de 0.001 y un batch size de 8.

$$FDE = \frac{\sum_{i=1}^n \|\hat{Y}_{T_{pred}-1}^i - Y_{T_{pred}-1}^i\|}{n}$$

$n$  numero de peatones

$\hat{Y}_{T_{pred}-1}^i$  ultima posición de la predicción del peaton  $i$

$Y_{T_{pred}-1}^i$  ultima posición futura real del peaton  $i$

$\|\cdot\|$  norma euclideana

Figure 6: Final Displacement Error, ADE.

Table 1: Experimentos con red MLP, Datos de entrenamiento.

Learning Rate = 0.001						
Capa	Unidades	Epoca	Batch Size	Loss	ADE	FDE
1	100	200	8	0.00024397	0.733	1.332
1	128	200	32	0.00035040	0.955	1.531
2	128	200	32	0.00032271	0.895	1.630
2	128	400	32	0.00022113	0.760	1.201

Table 2: Experimentos con red MLP, Datos de prueba.

Learning Rate = 0.001						
Capa	Unidades	Epoca	Batch Size	Loss	ADE	FDE
1	100	200	8	0.00029056	0.724	1.342
1	128	200	32	0.00037995	0.922	1.490
2	128	200	32	0.00037258	0.917	1.674
2	128	400	32	0.00028953	0.767	1.245

Table 3: Experimentos con red CNN, Datos de entrenamiento.

Learning Rate = 0.001						
Unidades	Epoca	Batch Size	Loss	ADE	FDE	
64	20	32	0.00182925	2.230	3.279	
128	400	32	32	0.00044860	1.120	1.776

Table 4: Experimentos con red CNN, Datos de prueba.

Learning Rate = 0.001						
Unidades	Epoca	Batch Size	Loss	ADE	FDE	
64	20	32	0.00271799	2.784	4.142	
128	400	32	32	0.00050332	1.070	1.694

Table 5: Experimentos con red LSTM, Datos de entrenamiento.

Learning Rate = 0.001						
Capa	Unidades	Epoca	Batch Size	Loss	ADE	FDE
1	128	20	32	0.00114510	1.885	2.909
1	128	100	32	0.00031806	0.972	1.725
1	128	400	32	0.00020208	0.715	1.217
2	128,128	400	32	0.00017601	0.669	1.189
1	512	400	32	0.00018763	0.631	1.191
1	1024	400	32	0.00028904	0.674	1.210

Table 6: Experimentos con red LSTM, Datos de prueba.

Learning Rate = 0.001						
Capa	Unidades	Epoca	Batch Size	Loss	ADE	FDE
1	128	20	32	0.00161728	2.136	3.424
1	128	100	32	0.00038440	0.990	1.779
1	128	400	32	0.00028795	0.800	1.443
2	128,128	400	32	0.00025212	0.716	1.312
1	512	400	32	0.00025774	0.690	1.318
1	1024	400	32	0.00026150	0.717	1.395

Table 7: Comparación de resultados de los modelos propuestos.

Modelo	ADE(m)	FDE(m)	Loss
MLP	0.733	1.332	0.00024397
CNN	1.120	1.776	0.00044860
LSTM	0.631	1.191	0.00018763

95 Por lo que se tomaron estos parametros para la creación del modelo final propuesto utilizando una  
96 red MLP.

### 97 4.3 Red Neuronal Convolucional, CNN

98 Para el modelo creado con una red neuronal convolucional, CNN, se obtuvo mejor resultado con una  
99 capa de 128 unidades, 400 epocas, un learning rate de 0.001 y batch de 32 como se muestra en la  
100 Table 3 y Table 4, tanto para conjunto de datos de entrenamiento como para el conjunto de datos de  
101 pruebas.

### 102 4.4 Red Neuronal Recurrente, Long Short Term Memory, LSTM

103 Se realizó distintos experimentos variando la estructura del modelo utilizando una red neuronal  
104 recurrente, long short term memory, LSTM, obteniendo mejor resultado, tanto en el conjunto de  
105 datos de entrenamiento como en el conjunto de datos de prueba utilizando una capa de 512 unidades,  
106 400 epocas, learning rate de 0.001 y batch size de 32 como se muestra en la Tabla 5 y Tabla 6.

### 107 4.5 Resultados

108 Como se muestra en los resultados de la Table 7, el modelo propuesto utilizando una red neuronal  
109 recurrente Long Short Term Memory, LSTM, es el modelo con el que se obtienen los mejores  
110 resultados de predicción de trayectoria de peatones, con metricas menores para Average Displacement  
111 Error, ADE y Final Displacement Error, así como para el costo, en comparación con el modelo que  
112 utiliza una red neuronal multi-capas, MLP, que es el que mostró ser el segundo con buenos resultados,  
113 quedando en tercer lugar el que utiliza una red neuronal convolucional, CNN.

114 En la Figure 7, se muestran algunos ejemplos de resultados de la predicción de las trayectorias  
115 utilizando los tres modelos propuestos. En las imagenes la trayectoria en color gris representa la  
116 trayectoria inicial, la trayectoria de color verde corresponde a la trayectoria real del peaton.

117 La trayectoria con puntos azules corresponde a los resultados obtenidos de la predicción utilizando el  
118 modelo MLP, la trayectoria de color púrpura corresponde a la predicción con el modelo CNN y la  
119 trayectoria roja corresponde al modelo que utiliza la red neuronal LSTM.

120 Como se puede observar en la Figure 7, algunas trayectorias son difíciles de predecir para los modelos,  
121 un ejemplo de esto, es la imagen (c) de la Figure 7, donde un cambio de trayectoria fue difícil predecir  
122 para los tres modelos, así como para la imagen (d), donde se puede observar que el peaton cambio  
123 la velocidad de la trayectoria. Sin embargo, en trayectorias rectas, los modelos obtuvieron mejores  
124 resultados en la predicción de la trayectorias. Estos resultados se muestran en la Table 8.

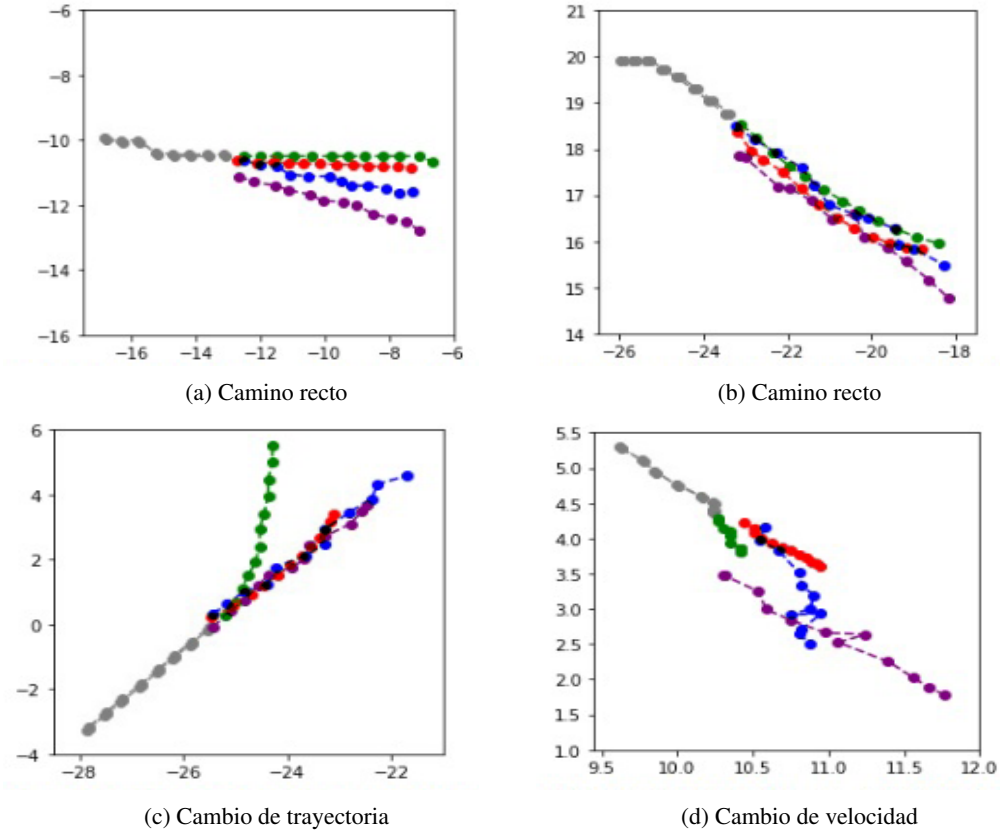


Figure 7: Ejemplos de algunas predicciones de los modelos

## 5 Conclusiones

En este trabajo, se propone tres modelos que se pueden adaptar a la predicción de movimiento de trayectorias de peatones. Este trabajo describe una arquitectura que utiliza una red neuronal recurrente LSTM para ser comparada con una arquitectura que utiliza una red neuronal multi-capa y una que utiliza una red neuronal convolucional, para mostrar el beneficio que se obtiene de utilizar un modelo con este tipo de arquitectura en tareas de predicción de movimiento de trayectoria de peatones. Por otro lado, acorde a los resultados de los experimentos realizados se pudo observar que, a pesar, que los modelos en general logran predecir las trayectorias futuras con poca distancia de diferencia, los modelos tienen dificultad para predecir algunas trayectorias como por ejemplo cuando hay un cambio de dirección.

## References

- [1] Jason Brownlee. *Deep Learning for Time Series Forecasting, Predict the Future with MLPs, CNNs and LSTMs in Python*. 2018.
- [2] Stanford Drone Dataset. [https://cvgl.stanford.edu/projects/uav\\_data/](https://cvgl.stanford.edu/projects/uav_data/), 2016.
- [3] Ali Haidar and Brijesh Verma. Monthly rainfall forecasting using one-dimensional deep convolutional neural network. *IEEE Access*, 6:69053–69063, 2018.
- [4] Alexandre Robicquet, Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 549–565, Cham, 2016. Springer International Publishing.

- 144 [5] Simone Zamboni, Zekarias Tilahun Kefato, Sarunas Girdzijauskas, Christoffer Norén, and Laura  
145 Dal Col. Pedestrian trajectory prediction with convolutional neural networks. *Pattern Recognition*,  
146 121:108252, 2022.