CEA Comité Español de Automática

XLV JORNADAS AUTOMÁTICA MÁLAGA 2024

UNIVERSIDADE DA CORUÑA

Jornadas de Automática

Transformer autorregresivo de grafos esqueléticos

Villa, J.[a,*], de la Escalera, A.[a], Armingol, J. M.[a]

[a]*Intelligent Systems Lab (LSI) Research Group, Universidad Carlos III de Madrid, Avda. de la Universidad, nº 30 , 28911, Leganés, España.*

To cite this article: Villa, J., de la Escalera, A., Armingol, J. M. 2024. Skeleton Graph Transformer. Jornadas de Automática, 45. https://doi.org/10.17979/ja-cea.2024.45.10773

Resumen

Analizar, comprender y predecir el comportamiento peatonal es un punto clave en el desarrollo de sistemas de conducción autónoma. En los últimos años, con el desarrollo exponencial en el campo de la visión por computador y el aprendizaje profundo, se han realizado grandes avances en la estimación de la pose humana y la clasificación de secuencias de movimiento en diferentes tipos de actividades. Este artículo propone un nuevo método autorregresivo, enfocado en tareas de predicción de movimiento de peatones. El sistema consta de un *Transformer*, que le permite analizar la información temporal disponible hasta el momento y generar una predicción del futuro inmediato. Además, incluye *Redes Convolucionales de Grafos* que facilitan la comprensión de la estructura espacial del esqueleto. Se han realizado experimentos sobre el conjunto de datos *Kinetics-Skeleton* y al final de este artículo se discute sobre los resultados y las futuras vías de estudio.

Palabras clave: Redes neuronales, Aprendizaje automático, Modelado de series temporales, Sistemas de control de tráfico, Vehículos autónomos.

Autoregressive skeleton graph transformer

Abstract

Analyzing, understanding, and predicting pedestrian behavior is a key aspect in the development of autonomous driving systems. In recent years, with the exponential growth in the fields of computer vision and deep learning, significant advances have been made in human pose estimation and the classification of movement sequences in various types of activities. This article proposes a new autoregressive method focused on pedestrian movement prediction tasks. The system consists of a *Transformer*, which allows it to analyze the temporal information available so far and generate a prediction of the immediate future. Additionally, it includes *Graph Convolutional Networks* that facilitate the understanding of the spatial structure of the skeleton. Experiments have been conducted on the *Kinetics-Skeleton* dataset, and at the end of this article, the results and future research directions are discussed.

Keywords: Neural networks, Machine learning, Time series modelling, Traffic control systems, Autonomous vehicles.

1. Introducción

La comprensión del movimiento humano es un requisito indispensable para poder predecir las acciones que realizará un peatón en el futuro cercano. Este entendimiento detallado es crucial para la creación de sistemas inteligentes capaces de interactuar de manera efectiva y segura con los seres humanos en una variedad de contextos, como la conducción autónoma o la supervisión de entornos urbanos.

1.1. Estado del arte

A lo largo del tiempo se han propuesto diferentes soluciones que conjuntamente contribuyen al diseño de sistemas más sofisticados para el análisis del comportamiento humano. Por un lado, los modelos de estimación de pose mediante visión artificial como *OpenPose* (Cao et al., 2017) o *YOLOv8-pose* (Jocher et al., 2023), abordan la tarea de la predicción de un esqueleto, que sintetiza la localización de ciertas par-

*Autor para correspondencia: javillap@pa.uc3m.es

tes del cuerpo en un instante determinado. En nuestro caso específico, utilizamos los esqueletos preprocesados por Yan et al. (2018), mediante *OpenPose* (Cao et al., 2017). Este modelo obtiene como entrada la imagen que se desea analizar y devuelve las coordenadas de los *keypoints* de cada persona. Por otro lado, los modelos de clasificación de actividad, predicen la acción que ha realizado un sujeto, analizando la secuencia de movimientos que este ha realizado a lo largo de un periodo de tiempo. Estas arquitecturas están basadas en *Redes Convolucionales de Grafos* (Yan et al., 2018; Heidari and Iosifidis, 2021, 2020), las cuales pueden comprender la estructura del esqueleto y permiten dar un contexto al movimiento de la persona. Finalmente, los modelos de predicción de movimiento (Aksan et al., 2019, 2021) funcionan con arquitecturas secuencia-a-secuencia (Sherstinsky, 2020; Vaswani et al., 2017); es decir, reciben un registro de esqueletos y realizan la predicción para los siguientes t instantes. Estos modelos están generalmente enfocados en la predicción de los ángulos de las articulaciones, sin ser capaces de predecir la posición real del esqueleto.

1.2. Aportación

En este artículo se propone un nuevo método autorregresivo de predicción de movimiento basado en *Redes Convolucionales de Grafos* (Heidari and Iosifidis, 2021) y *Transformers* (Vaswani et al., 2017), capaz de predecir el siguiente movimiento de una persona. Este modelo puede aplicarse en situaciones como la conducción autónoma, en las que se necesita prever el comportamiento de uno o varios sujetos en el corto plazo, para realizar una toma de decisiones. También detallamos el proceso de entrenamiento y conjuntos de datos utilizados, funciones de coste y resultados, así como las futuras vías de investigación en las que se trabajará.

2. Formulación del problema

Mientras que otras aproximaciones limitan su enfoque a predecir los ángulos de cada articulación, sin obtener posiciones reales, esta investigación se ha centrado en la predicción directa de coordenadas posicionales. En el caso específico de Aksan et al. (2021), se realiza una estimación angular para cada articulación, utilizando como características una matriz de rotación que define cómo está orientada cada una de ellas.

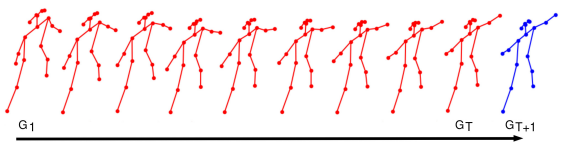

Figura 1: Secuencia de esqueletos espacio-temporal. En rojo se muestran los esqueletos con los que el modelo puede trabajar para predecir el futuro movimiento, el cual está representado en azul.

En este trabajo, se representan los movimientos peatonales a través de un *grafo espacio-temporal*, el cual se puede expresar de forma matricial como un tensor $G \in \mathbb{R}^{N\times C_{\text{in}}\times T\times V}$. En esta representación, N es el tamaño de la muestra (*batch size*), importante para el procesamiento de varias secuencias en paralelo durante el entrenamiento. Durante la inferencia se utiliza un $N = 1$ para hacer predicciones individuales, pero también se pueden concatenar secuencias de diferentes individuos sobre la dimensión del *batch*, para así aprovechar la paralelización de la *GPU*. C_{in} representa las características de cada articulación, es decir, las coordenadas $[x, y]$. T es la duración temporal de la secuencia, es decir, el número de esqueletos. Por último, V corresponde al número de articulaciones en el esqueleto.

Dada una secuencia $G = \{g_1, g_2, \ldots, g_T\}$, el modelo debe predecir el esqueleto g_{T+1} (Figura 1). Este proceso se puede repetir de forma iterativa para realizar una predicción más extendida en el tiempo, utilizando las predicciones previas como entradas para las predicciones futuras.

3. Modelo

En esta sección, se detalla la arquitectura propuesta para abordar el problema. El modelo consta de tres partes diferenciadas: el *Bloque de Embedding*, el *Transformer* y el *Bloque de Proyección*.

3.1. Embeddings y proyección de esqueletos

Los *Transformers* (Vaswani et al., 2017) son arquitecturas que operan sobre un tensor específico $E \in \mathbb{R}^{N\times T\times d_{\text{model}}}$, donde E es un *embedding* de longitud temporal T, en el cual, cada esqueleto se representa como un vector de dimensión d_{model}. Los *embeddings* son vectores que representan un conjunto de entidades en un espacio dimensional continuo ampliado. Como ya se ha mencionado con anterioridad, se trabaja con secuencias espacio-temporales de esqueletos representadas por un tensor $G \in \mathbb{R}^{N\times C_{\text{in}}\times T\times V}$. Por ello, se ha diseñado una arquitectura que nos permite transformar nuestra secuencia G al formato E.

En primer lugar, se utiliza una *convolución espacial* para extraer más información de cada una de las articulaciones de cada esqueleto. Se ha utilizado la convolución propuesta en ST-BLN (Heidari and Iosifidis, 2021) como referencia, prescindiendo de la convolución temporal para evitar diluir la información entre los elementos de la secuencia. Se ha implementado la convolución espacial (Heidari and Iosifidis, 2021), la cual no necesita de una matriz de adyacencia predefinida, utilizando una totalmente aprendible por el modelo. Esto permite trabajar con numerosas topologías de esqueletos sin necesidad de configurar los valores de dicha matriz. También se ha modificado la función de activación sustituyendo la ReLU original por una GELU (Hendrycks and Gimpel, 2016) para mejorar la convergencia.

$$G^{(l)} = \text{GELU}\left(\sum_p (M_p^{(l)}) G^{(l-1)} W_p^{(l)}\right) \tag{1}$$

La capa l de convolución (1), opera sobre el subconjunto p de nodos vecinos, donde $M_p^{(l)}$ es la matriz de adyacencia aprendible, $G^{(l-1)}$ es la secuencia de entrada, $W_p^{(l)}$ son los pesos de la convolución y $G^{(l)}$ es la secuencia de salida. En el caso de ST-GCN, las articulaciones de los esqueletos se subdividen en 3 subconjuntos(Yan et al., 2018), siendo el primer subconjunto los nodos raíz , el segundo los vecinos más cercanos al centro de gravedad y el tercero los más lejanos. En la arquitectura propuesta también se ha utilizado $p = 3$, pero los subconjuntos son aprendidos durante el entrenamiento.

El proceso de *embedding* implica la aplicación sucesiva de convoluciones espaciales (1), aumentando el número de canales de $C_{\text{in}} = 2$ hasta $C = 120$. Cuando se ha alcanzado la última convolución, $C \times V$ debe igualar a d_{model}, de modo que con una reconfiguración de ejes, el tensor $G' \in \mathbb{R}^{N \times C \times T \times V}$ pueda convertirse a $E \in \mathbb{R}^{N \times T \times d_{\text{model}}}$.

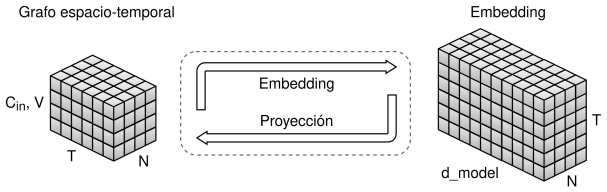

Figura 2: Conversión entre *embedding* y grafo.

La operación de *embedding* es opuesta a la de *proyección*. Así, una secuencia de esqueletos G_1 puede comprimirse en un *embedding* E_1, al igual que un *embedding* E_2 puede proyectarse en un grafo G_2, realizando la operación inversa (Figura 2). El único matiz a tener en cuenta es que, mientras que en el *embedding* el número de canales de las articulaciones aumenta, en la *proyección*, el número se reduce hasta ser bidimensional.

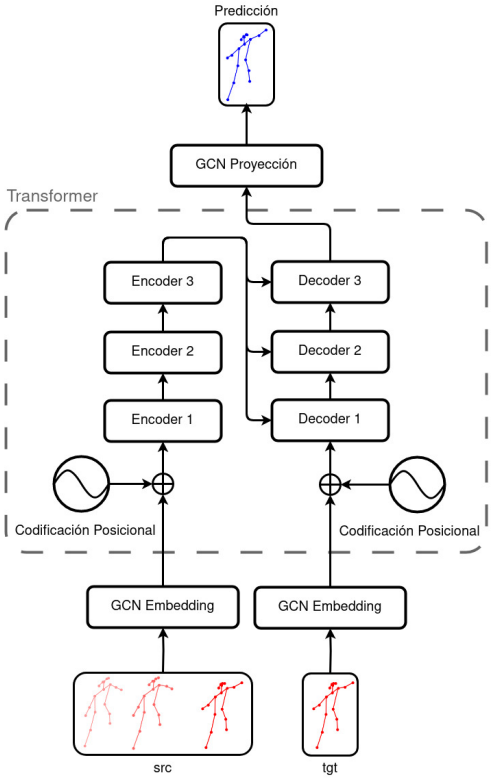

Figura 3: Arquitectura completa.

3.2. *Transformer*

El *Transformer* fue propuesto originalmente por Vaswani et al. (2017) para solucionar problemas de *Procesamiento de Lenguaje Natural*. En este tipo de problemas existe un vocabulario limitado, compuesto por letras o pseudo-sílabas que permiten conformar palabras y potencialmente textos. En el problema abordado en esta investigación, se cuenta con la desventaja de que los esqueletos tienen articulaciones posicionadas en un sistema bidimensional de coordenadas, lo que supone una infinita cantidad de soluciones para cada uno de ellos. Como ya se ha mencionado antes, el *Bloque de Embedding* soluciona este problema, funcionando a la vez como *tokenizador* continuo y capa de *embedding*.

El modelo procesa la información tal y como se detalla en la Figura 3. Dada una secuencia de entrada $G_{\text{in}} = \{g_{\text{in}}^1, g_{\text{in}}^2, \ldots, g_{\text{in}}^T\}$, se extraen dos secuencias necesarias para el *Transformer*: $G_{\text{src}} = \{g_{\text{in}}^1, g_{\text{in}}^2, \ldots, g_{\text{in}}^T\}$ y $G_{\text{tgt}} = \{g_{\text{in}}^T\}$. Ambas se introducen en el *Bloque de Embedding* obteniendo E_{src} y E_{tgt}, posteriormente se les aplica una *codificación posicional* (Vaswani et al., 2017). En este punto, la secuencia E_{src} se procesa a través de tres *encoders* utilizando *mecanismos de atención*. La secuencia E_{tgt} es introducida junto con la salida del tercer *encoder* en cada uno de los tres *decoders* implementados. A pesar de utilizar un G_{tgt} con longitud temporal unitaria, se ha implementado una *máscara de atención* en los *decoders*, que asegura un funcionamiento causal. La salida del último *decoder* E_{pred} es una representación de alta dimensionalidad, que debe ser procesada por el *Bloque de Proyección*, obteniendo como resultado la predicción del siguiente esqueleto.

3.3. *Entrenamiento del modelo*

El modelo propuesto ha sido entrenado sobre el conjunto de datos *Kinetics-Skeleton* (Kay et al., 2017), utilizando las herramientas de pos-procesado de Passalis et al. (2022) sobre los datos provistos por Yan et al. (2018). Estos datos contienen anotaciones de esqueletos de 18 articulaciones obtenidos a través de *OpenPose Toolbox* (Cao et al., 2017), anotados sobre 300.000 vídeos en los que los sujetos efectúan diferentes acciones. Por lo tanto, únicamente se tratan de anotaciones de esqueletos, no de imágenes originales del dataset *Kinetics-Skeleton*. Entre las actividades realizadas por los sujetos se encuentran algunas como "pasear al perro", "correr en una cinta", "jugar al baloncesto" o "beber". Las coordenadas de cada esqueleto se encuentran normalizadas, por lo tanto, no son posiciones reales en las imágenes originales.

Se utiliza la persona con mayor confianza de detección, eliminando además las secuencias que tengan un gran porcentaje de *frames* sin anotar. Después de este filtrado inicial, se dividen las secuencias de 300 instantes de duración en secuencias de 10. De esta manera se obtienen tensores de la forma $G_{\text{in}} = \{g_{\text{in}}^1, g_{\text{in}}^2, \ldots, g_{\text{in}}^{10}\}$, que por un lado deben dividirse en $G_{\text{src}} = \{g_{\text{in}}^1, g_{\text{in}}^2, \ldots, g_{\text{in}}^9\}$ y $G_{\text{tgt}} = \{g_{\text{in}}^9, g_{\text{in}}^{10}\}$. Los datos pos-procesados se dividen en proporciones 70-30 % para conjuntos de entrenamiento y validación respectivamente. Durante el entrenamiento, se desplazan hacia la derecha las secuencias G_{tgt}, de modo que pasan a ser $\{g_{\text{in}}^9\}$. De esta manera, se introducen al modelo G_{src} y G_{tgt} y se compara el resultado utilizando g_{in}^{10} como referencia.

Las métricas utilizadas durante el entrenamiento consisten en una combinación del *Error Cuadrático Medio* (MSE) y el *Error Angular Medio* (MAE). Para cada una de las muestras n, el MSE (2) recorre cada articulación v, comparando cada canal c, obteniendo un error promedio del posicionamiento del esqueleto.

$$\text{MSE} = \frac{\sum_n^N \sum_v^V \sum_c^C (g^{(n,v,c)} - g_{\text{real}}^{(n,v,c)})^2}{N \cdot V \cdot C_{\text{in}}} \tag{2}$$

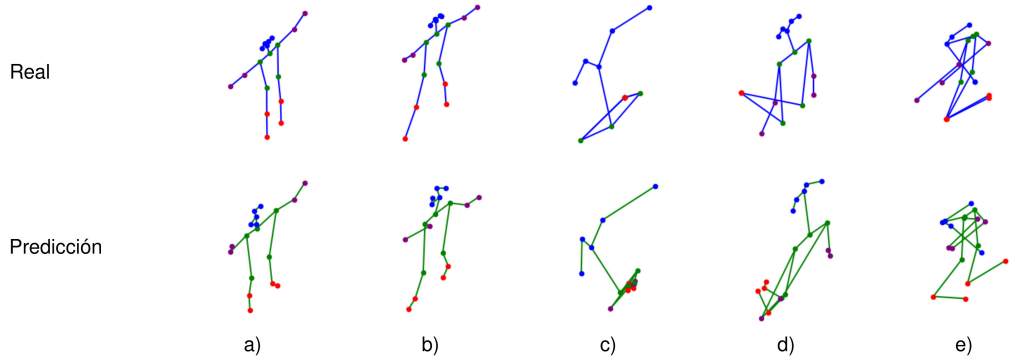

Figura 4: Resultados de predicción de movimiento. Se muestran diferentes predicciones realizadas sobre varias secuencias. Se ha utilizado un código de colores común para los nodos (azul para la cabeza, verde para el torso, morado para los brazos y rojo para las piernas), de manera que se pueda identificar fácilmente la parte del cuerpo representada. Además, las aristas se han representado en azul para el esqueleto real y en verde para la predicción realizada por el modelo.

El *Error Angular Medio* (3) también opera para todas las muestras del lote n, seleccionando tríos de articulaciones $\{j,k,l\}$ conectados entre sí. Las conexiones entre nodos vecinos vienen definidas por la matriz de adyacencia A. Para cada trío, se calculan los vectores que unen la articulación j con sus vecinos k y l, tanto para el esqueleto predicho como para el real. De esta forma, se obtiene el ángulo que forma cada pareja de vectores y el error correspondiente.

$$\text{MAE} = \frac{\sum_{(j,k,l)\in A}\sum_{n}^{N}\left|\cos^{-1}\left(\frac{\mathbf{v}_{jk}^{(n)}\cdot\mathbf{v}_{jl}^{(n)}}{|\mathbf{v}_{jk}^{(n)}||\mathbf{v}_{jl}^{(n)}|}\right)-\cos^{-1}\left(\frac{\mathbf{v}_{jk,\text{real}}^{(n)}\cdot\mathbf{v}_{jl,\text{real}}^{(n)}}{|\mathbf{v}_{jk,\text{real}}^{(n)}||\mathbf{v}_{jl,\text{real}}^{(n)}|}\right)\right|}{|A|\cdot N} \tag{3}$$

En el entrenamiento, se atribuye un 60 % del peso al MAE y el 40 % restante al MSE, asegurando que el modelo no solo se enfoca en minimizar el error posicional del esqueleto, sino que también aprende a minimizar el error angular. La decisión de no utilizar únicamente MSE es puramente experimental, ya que se ha observado que es muy sencillo que el modelo quede atrapado en un mínimo local, produciendo una misma salida para cualquier secuencia de entrada.

Se ha utilizado como optimizador un *Adam*, con técnicas de regularización como un *weight_decay* $= 10^{-3}$ y un *dropout* $= 10^{-1}$. Además se ha utilizado un *learning_rate* $= 10^{-4}$, un *batch_size* $= 128$ y se ha entrenado el modelo durante 300 *epoch*.

4. Resultados

4.1. Predicción de esqueletos

Todas las pruebas han sido realizadas sobre el *dataset Kinetics-Skeleton* (Kay et al., 2017), utilizando específicamente los datos proporcionados por Yan et al. (2018).

Tabla 1: Resultados de MSE y MAE para las secuencias de la Figura 4.

Secuencia	MSE (normalizado)	MAE (radianes)
a)	0.0032	0.1530
b)	0.0067	0.1785
c)	0.0027	0.3337
d)	0.0206	0.3205
e)	0.0034	0.2525

En la Figura 4 se presentan diversas predicciones realizadas por el modelo, acompañadas de sus correspondientes *ground truth*. Para cada predicción, el modelo emplea los 9 instantes anteriores. En la Tabla 1 se muestran las métricas MSE (2) y MAE (3) correspondientes a cada una de las predicciones ilustradas en la Figura 4.

4.2. Visualización de la atención

El modelo propuesto emplea *mecanismos de atención* tanto *espacial* (Heidari and Iosifidis, 2021) como *temporal* (Vaswani et al., 2017), los cuales son esenciales para analizar la relevancia de los elementos en el grafo G. En la Figura 5 se muestra la atención espacial en una de las capas del *Bloque de Embedding*. Este gráfico proporciona una representación visual de la matriz $M_p^{(l)}$ utilizada en la *convolución espacial de grafos* (1), ilustrando la atención que el modelo presta a cada articulación del esqueleto durante las operaciones de convolución. Este *mecanismo de atención* funciona como una *matriz de adyacencia*, modelando totalmente las conexiones del grafo formado por el esqueleto, a diferencia de las convoluciones propuestas en Yan et al. (2018) y Heidari and Iosifidis (2020), donde se aplican *máscaras de atención* sobre una *matriz de adyacencia* predefinida. Cabe destacar que en cada capa l, esta atención varía, adaptándose a las características específicas que el modelo necesita aprender en cada nivel.

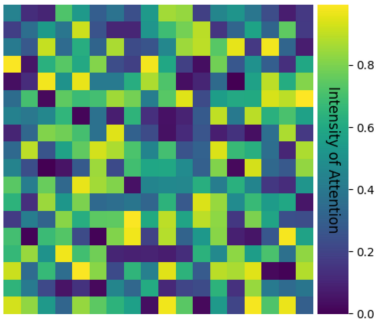

Figura 5: Visualización de la *atención espacial* en el *Bloque de Embedding*. Se muestra la atención que presta el modelo a cada articulación respecto de las otras articulaciones en una de las capas convolucionales.

El *Transformer* implementado, al igual que el original Vaswani et al. (2017), utiliza *mecanismos de atención* para puntuar la importancia de cada elemento de la secuencia. En

este caso, la atención se puede entender como *temporal*, ya que cada elemento de la secuencia corresponde a un *embedding* de un esqueleto en un instante de tiempo concreto.

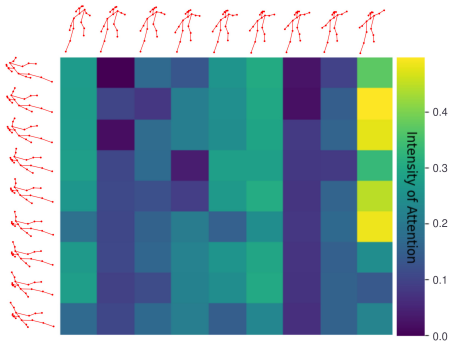

Figura 6: Visualización de la *atención temporal* en un *multi-head attention* del *encoder*.

En el caso de los *encoder*, la atención se puede visualizar como la matriz representada en la Figura 6. Dado que se trabaja son secuencias G_{src} de 9 esqueletos, esta matriz A es de 9×9 elementos, donde el elemento a_{ij} representa la atención que presta el esqueleto de posición i al que se encuentra en la posición j.

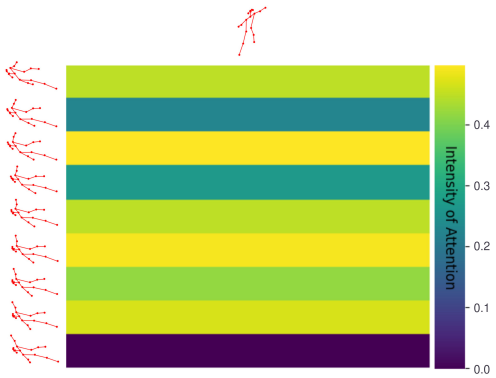

Figura 7: Visualización de la *atención temporal* en un *multi-head attention* del *decoder*.

Por otro lado, en los *decoder* se trabaja con secuencias G_{tgt} de un único elemento. Por lo tanto, la *matriz de atención temporal* en los *multi-head attention* de cada *decoder* tiene forma 9×1 (Figura 7). En este caso, cada elemento a_i representa la atención que presta el esqueleto del G_{tgt} al elemento i del G_{src}.

5. Conclusiones

En este artículo se ha propuesto un nuevo método de predicción de esqueletos autorregresivo, basado en *Redes Convolucionales de Grafos* y *Transformers*, capaz de predecir la posición de las articulaciones de un esqueleto en el siguiente instante, dada una secuencia de movimiento. Los resultados obtenidos en esta investigación son prometedores y permitirán perfeccionar los métodos propuestos en el futuro.

5.1. Trabajos futuros

Es importante destacar que los datos utilizados en esta investigación no cuentan con una perspectiva vehicular, sino infrastructural. Por lo tanto, el trabajo futuro deberá centrarse en entrenar el modelo propuesto con *datasets* más sofisticados, que incluyan un punto de vista vehicular. Por otro lado, se buscará mejorar la arquitectura para procesar secuencias temporales más largas y aumentar la precisión. Además, para aumentar la información para la toma de decisiones del vehículo, el trabajo se deberá centrar en obtener la proximidad de los peatones respecto del vehículo, incluso la velocidad con la que se acercan o alejan de él. Por último, se entrenará el modelo sobre datos que permitan la evaluación en conjuntos de prueba estandarizados.

Agradecimientos

Agradecer a los proyectos PID2021-124335OB-C21, PID2022-140554OB-C32 y PDC2022-133684-C31 financiado por MCIN/AEI/10.13039/5011000 11033.

Referencias

Aksan, E., Kaufmann, M., Cao, P., Hilliges, O., 2021. A spatio-temporal transformer for 3d human motion prediction, in: 2021 International Conference on 3D Vision (3DV), IEEE Computer Society, Los Alamitos, CA, USA. pp. 565–574. doi:10.1109/3DV53792.2021.00066.

Aksan, E., Kaufmann, M., Hilliges, O., 2019. Structured prediction helps 3d human motion modelling. doi:10.1109/ICCV.2019.00724.

Cao, Z., Simon, T., Wei, S.E., Sheikh, Y., 2017. Realtime multi-person 2d pose estimation using part affinity fields, in: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR).

Heidari, N., Iosifidis, A., 2020. Temporal attention-augmented graph convolutional network for efficient skeleton-based human action recognition. CoRR abs/2010.12221. arXiv:2010.12221.

Heidari, N., Iosifidis, A., 2021. On the spatial attention in spatio-temporal graph convolutional networks for skeleton-based human action recognition, in: 2021 International Joint Conference on Neural Networks (IJCNN), pp. 1–7. doi:10.1109/IJCNN52387.2021.9534440.

Hendrycks, D., Gimpel, K., 2016. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. CoRR abs/1606.08415. arXiv:1606.08415.

Jocher, G., Chaurasia, A., Qiu, J., 2023. Ultralytics yolov8. URL: https://github.com/ultralytics/ultralytics.

Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., Suleyman, M., Zisserman, A., 2017. The kinetics human action video dataset. CoRR abs/1705.06950. arXiv:1705.06950.

Passalis, N., Pedrazzi, S., Babuska, R., Burgard, W., Dias, D., Ferro, F., Gabbouj, M., Green, O., Iosifidis, A., Kayacan, E., Kober, J., Michel, O., Nikolaidis, N., Nousi, P., Pieters, R., Tzelepi, M., Valada, A., Tefas, A., 2022. Opendr: an open toolkit for enabling high performance, low footprint deep learning for robotics, in: Proceedings of the 2022 IEEE/RSJ international conference on intelligent robots and systems.

Sherstinsky, A., 2020. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. Physica D: Nonlinear Phenomena 404, 132306. doi:https://doi.org/10.1016/j.physd.2019.132306.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I., 2017. Attention is all you need, in: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Advances in neural information processing Systems, Curran Associates, Inc.

Yan, S., Xiong, Y., Lin, D., 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. Proceedings of the AAAI Conference on Artificial Intelligence 32. doi:10.1609/aaai.v32i1.12328.