Multimodal Spatio-Temporal Vehicle Speed Prediction Using Hexagonal Grids in Santiago, Chile

Diego Silva[1,†], Billy Peralta[1,*,†], Orietta Nicolis[1], Andres Bronfman[1], Luis Caro[2] and Hans Lobel[3]

[1]Universidad Andres Bello, Facultad de Ingeniería, Santiago, 7500971, Chile
[2] Universidad Católica de Temuco, Departamento de Ingenieria Informatica, Temuco, Chile
[2] Pontificia Universidad Catolica, Departamento de Ciencias de Computación, Santiago, Chile

Abstract
The rapid growth of e-commerce and the increasing need for logistical optimization in highly congested urban environments require advanced models for vehicle speed prediction. Traditional models often overlook the influence of the geographic environment and rely solely on historical speed data, limiting their accuracy in dynamic scenarios. In addition, most approaches use square grid structures, which introduce spatial distortions and fail to capture the connectivity of road networks effectively. In this work, we propose a multimodal model that integrates spatio-temporal information from GPS sensors with satellite imagery, leveraging HexConvLSTM and MLP neural networks to enhance predictive robustness. Unlike conventional methods, our approach utilizes a hexagonal grid representation, which provides a more uniform spatial structure and improved neighborhood representation that aligns better with road topology than conventional square grids for modeling multidirectional traffic dynamics. This paper presents the implementation and evaluation of the model, highlighting its effectiveness in improving the accuracy of route planning for freight transportation in Santiago Centro. The results show that the multimodal approach significantly reduces the mean absolute error (MAE) to 2.296 in test dataset, outperforming a baseline model based solely on spatiotemporal data by 8.3%. This research validates the benefits of incorporating visual data and hexagonal grid-based spatial modeling into traffic prediction and suggests exploring its applicability in other urban settings.

Keywords
Spatio-temporal prediction, Multimodal, ConvLSTM, Traffic velocities

1. Introduction

The rapid growth of e-commerce has transformed logistics into a critical factor for business competitiveness. Fast and efficient deliveries are now an essential requirement for consumers, who increasingly demand shorter delivery times [1]. In this context, optimizing the planning of transport routes has become a key challenge, particularly in highly congested urban areas such as downtown Santiago, Chile. From a modeling point of view, traffic prediction has evolved from traditional statistical techniques, such as ARIMA and SARIMA, to more advanced deep learning techniques based on recurrent and convolutional neural networks [2]. However, many of these models remain limited by their exclusive reliance on historical speed data and GPS coordinates, failing to incorporate visual environmental information, such as road layout, green space, and building density, which affects traffic flow and is otherwise not encoded in GPS data.

A fundamental limitation of conventional approaches lies in their inability to effectively capture the interaction between urban infrastructure and traffic dynamics. Factors such as building density, the presence of school zones, critical intersections, and recurrent congestion patterns are often ignored in traditional prediction models [3]. As a result, these models struggle to anticipate fluctuations in vehicle speed with sufficient accuracy, which affects decision-making in freight transportation logistics. To address this gap, we explore a multimodal approach that integrates spatiotemporal data from GPS sensors with satellite imagery, providing a more comprehensive representation of the urban traffic environment.

This work introduces a multimodal prediction model based on HexConvLSTM and MLP neural networks. The proposed architecture leverages LSTM networks to capture temporal dependencies, while satellite imagery is processed through a Multilayer Perceptron (MLP) to extract relevant urban features. By integrating these two modalities, our approach improves vehicle speed estimation for freight transportation in Santiago Centro, opti-

STRL'25: Fourth International Workshop on Spatio-Temporal Reasoning and Learning, 16 August 2025, Toronto, Canada
*** Corresponding author.
† These authors contributed equally.
d.silvapea@uandresbello.edu (D. Silva); billy.peralta@unab.cl (B. Peralta); orietta.nicolis@unab.cl (O. Nicolis); abronfman@unab.cl (A. Bronfman); lcaro@uct.cl (L. Caro); lcaro@uct.cl (H. Lobel)
0000-0002-5457-2157 (B. Peralta); 0000-0001-8046-6983 (O. Nicolis); 0000-0002-3122-3237 (A. Bronfman); 0000-0001-9378-397X (L. Caro); 0000-0003-3514-9414 (H. Lobel)

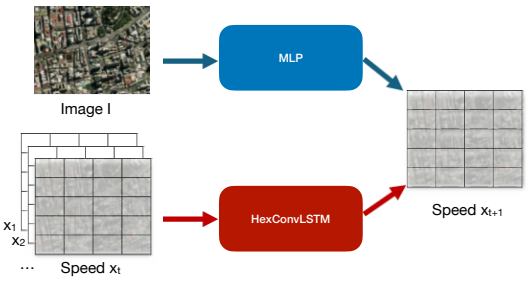

Figure 1: Overview of the proposed multimodal approach, integrating GPS-based spatio-temporal data (X_t) with satellite imagery (I) for enhanced traffic speed prediction.

mizing route planning and contributing to more efficient urban logistics management. Here, vehicle speed denotes the cell-level average traffic velocity.

Figure 1 illustrates the architecture of the proposed multimodal model for vehicle speed prediction, integrating spatiotemporal data ($x_1, x_2, ..., x_t$) from GPS sensors with visual information from a satellite image (I). The HexConvLSTM network models spatiotemporal relationships, while the MLP extracts features from I, combining both sources to predict future speed (x_{t+1}).

2. Background

2.1. Related work

Prediction of vehicle speed in urban environments has been extensively studied using deep learning techniques.

Stienen et al. [4] proposed a deep neural network model that integrates satellite data, meteorological information, and GPS trajectories to predict vehicle speed in regions with limited data availability. Their approach demonstrated that combining these data sources improves the accuracy of the prediction in areas lacking extensive historical records. The results showed that their model reduced the mean squared error compared to traditional methods, validating the importance of incorporating environmental data into traffic forecasting.

Guo et al. [5] developed NanoSight–YOLO, an optimized model for the detection of micro-vehicles in satellite imagery. Their work implemented an architecture based on Faster R-CNN and attention mechanisms to enhance the detection of small objects in highly congested urban environments. The proposal stood out for its use of advanced precision optimization techniques, which achieved improvements in recall and model accuracy, demonstrating the effectiveness of integrating computer vision into traffic monitoring.

Cheng et al. [6] explored the automatic detection of traffic regulators at intersections using a model based on Conditional Variational Autoencoders (CVAE). Their approach combined GPS data with satellite imagery to classify intersections into different categories based on the presence of traffic lights or priority signs. Using LSTM and CNN networks, they improved the identification of critical points in road infrastructure, facilitating their integration into traffic prediction systems.

Chowdhury and Sarwat [7] introduced GeoTorchAI, a deep learning framework designed to process spatio-temporal data in raster images and neural networks. Their methodology improved efficiency in handling large-scale geospatial data, optimizing segmentation, and classification of satellite images for traffic prediction applications. The use of model pretraining significantly reduced computational costs without compromising prediction accuracy.

Wang et al. [8] proposed a hybrid CNN-LSTM architecture for short- and long-term traffic prediction using remote sensing data. Their model transformed traffic information into a time-space matrix, improving the ability to anticipate congestion patterns. Although their results indicated a reduction in error in scenarios with high temporal dependence, their study did not incorporate visual data, limiting its applicability in densely populated urban environments. Instead, [9] presented a method for detecting vehicles and estimating their speeds using PlanetScope SuperDove satellite imagery. Using a Keypoint R-CNN model to track vehicle trajectories across RGB bands, a band timing difference was used to estimate speed. The validation was carried out using drone footage and GPS data from highways in Germany and Poland.

Sheehan et al. [10] explored the use of deep learning and high-resolution WorldView satellite imagery for large-scale traffic monitoring in Barcelona. Using the YOLOv3 object detection model, the study identifies vehicles in the city, achieving a precision of 0.69 and a recall of 0.79 and faced challenges in detecting vehicles on narrow streets, in shadows and under obstructions.

Kashyap et al. [11] reviewed recent advances in deep learning for traffic flow prediction, covering architectures such as CNN, RNN, LSTM, restricted Boltzmann machines (RBMs), and stacked autoencoders (SAEs). These models leverage multiple layers to extract higher-level features from raw input data. Similarly, Afandizadeh et al. [2] provided a detailed comparative analysis of deep learning (DL) and classical models for traffic forecasting. The study highlights that while DL algorithms (such as RNNs, CNNs, and LSTMs) offer higher accuracy and adaptability, classical models (such as ARIMA and regression-based methods) remain valuable in structured, low-complexity environments. Finally, Mystakidis et al. [12] explore advanced Traffic Congestion Prediction (TCP) methods, focusing on statistical models, ML, Deep Learning (DL), and ensemble approaches. They

evaluated various forecasting techniques, considering both regression and classification metrics. In addition, it outlines a step-by-step methodology commonly used in TCP research.

While prior work has demonstrated the benefits of integrating satellite or spatiotemporal data with deep learning architectures, our method is the first to explicitly combine a HexConvLSTM model operating on hexagonal grids with a visual MLP that processes satellite imagery, producing a unified multimodal model for short-term speed prediction.

2.2. HexConvLSTM

Prediction of vehicle speeds in urban environments is essential for optimizing traffic flow, a task commonly tackled using deep learning approaches such as ConvLSTM and Transformers. However, these approaches often assume a square grid representation, introducing distortions in spatial connectivity. Unlike square grids, the hexagonal structure offers better connectivity, as each cell has six equidistant neighbors instead of four or eight [13]. Recently, Bahamondes et al. [14] proposed HexConvLSTM, a neural network based on ConvLSTM adapted to hexagonal grid sequences, optimizing the representation of vehicular traffic and improving prediction accuracy.

The proposed method consists of three key stages: (i) **Hexagonal Grid Representation**, where raw traffic speed data are mapped onto a structured hexagonal grid using the H3 spatial indexing system. We used H3 resolution level 9, corresponding to hexagons with an average edge length of approximately 174 meters, balancing spatial resolution with data sparsity; (ii) **Preprocessing for Compatibility**, involving upsampling, padding, and shifting operations to transform the hexagonal structure into a format suitable for ConvLSTM while preserving its original neighborhood relationships; and (iii) **Hexagonal-Constrained Convolution**, where a custom convolutional kernel enforces hexagonal neighborhood relationships by masking non-adjacent cells in the input tensor, ensuring only valid hex neighbors contribute to the convolution.. This ensures that feature extraction respects the inherent properties of hexagonal data distributions.

The HexConvLSTM architecture consists of a sequence of ConvLSTM layers adapted with a hexagonal kernel constraint, followed by fully connected layers for final speed prediction. The ConvLSTM component captures spatial-temporal dependencies in vehicle movement, leveraging recurrent convolutional operations to model long-term traffic patterns. Meanwhile, the hexagonal transformation ensures that the model exploits the benefits of hexagonal connectivity while remaining compatible with conventional deep learning frameworks.

This architecture has the ability to incorporate hexagonal grid structures by introducing hex-aware preprocessing and masking techniques, while retaining compatibility with standard ConvLSTM implementations, enabling seamless integration into existing traffic forecasting pipelines. Figure 2 shows the proposed HexConvLSTM architecture and its data processing. Details can be reviewed in [14].

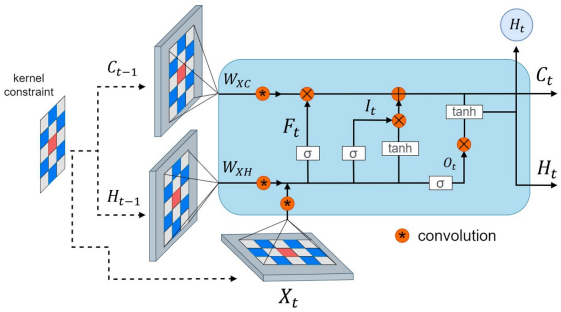

Figure 2: HexConvLSTM block at time step t from [14]. The variables are specified in [15].

3. Proposed method

The proposed model combines deep learning techniques for multimodal vehicle speed prediction in urban environments. The developed architecture integrates two complementary approaches: (1) a **HexConvLSTM** network to model the spatiotemporal dynamics of GPS sensor data and (2) a **CNN/MLP** to extract relevant features from satellite images. Each component and its integration into the final model are detailed below.

The model consists of two main branches that process different types of information before being merged into a final prediction layer. Figure 3 illustrates the overall system architecture.

3.1. HexConvLSTM Branch for Spatiotemporal Data

The first branch of the model processes GPS sensor data using a HexConvLSTM network, a variant of ConvLSTM designed to operate on a hexagonal grid instead of a square mesh. This approach enhances spatial connectivity between cells and reduces distortion in the representation of traffic patterns.

The processing flow in this branch begins with an input tensor of shape $(T, 44, 15, 1)$, where T represents the temporal sequence and $(44, 15)$ corresponds to the hexagonal grid. The data is then processed by a ConvLSTM2D layer with 128 filters and ReLU activation, constrained to a hexagonal kernel of size (5,3) to preserve spatial dependencies. Batch normalization is applied

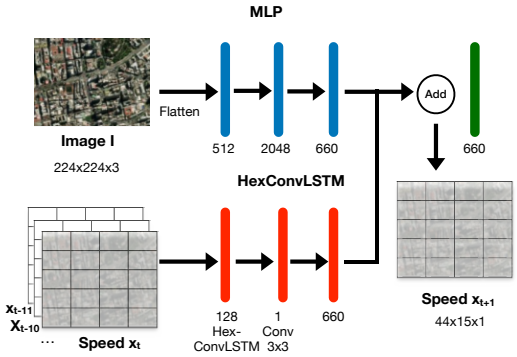

Figure 3: Architecture of the multimodal model based on HexConvLSTM and MLP.

to enhance stability and accelerate convergence during training. Subsequently, a final convolutional layer with a (3,3) kernel and a single filter refines spatiotemporal features. Finally, the output is reshaped to $(1, 44, 15, 1)$, ensuring compatibility with the multimodal integration framework.

3.2. CNN/MLP Branch for Satellite Images

The second branch of the model leverages both a **Multilayer Perceptron (MLP)** and convolutional neural networks (CNNs) to extract spatial features from satellite images. Given the relatively small and static nature of the input data (target representation: 44×15), an MLP can offer a computationally efficient alternative by avoiding unnecessary spatial convolutions while still capturing relevant feature structures.

The processing flow in this branch begins with RGB input images resized to $(224, 224, 3)$ pixels. The images are then flattened into a one-dimensional vector, followed by two fully connected layers with 512 and 2048 neurons, both using ReLU activation. Finally, the output layer is adjusted to match the hexagonal grid, consisting of 660 neurons with a linear activation function. Flattening preserves spatial context because each pixel index maps to a fixed geo-coordinate, letting the MLP learn location-specific weights.

3.3. Multimodal Fusion and Training Regime

Once the two branches finish their forward passes, their feature maps are added element-wise to produce a tensor of shape $(1, 44, 15, 1)$ that exactly mirrors the input hexagonal grid. Keeping this layout intact simplifies downstream error visualisation and ensures that no spatial information is lost during fusion.

The HexConvLSTM branch is first trained on the GPS-only subset and then frozen; initial tests showed that letting its weights update in the multimodal stage worsened validation accuracy. Consequently, the only trainable parts in the full network are (i) a lightweight MLP fed with the flattened $224 \times 224 \times 3$ satellite image, converting it into a 660-element vector that matches the grid, and (ii) the fusion bias term.

For comparison, we also tested a CNN-based visual branch (InceptionV3, EfficientNetB7, Xception), where the image retains its spatial structure and a global-average-pooling layer feeds a dense layer of 1024 units, followed by a 660-dimensional output. This branch is fine-tuned end-to-end, including the custom regression head.

4. Data collection and preprocessing

This study focuses on predicting vehicle speeds in urban environments by integrating spatiotemporal data from GPS sensors with visual features extracted from satellite imagery. The data pipeline consists of two primary stages: (1) data collection, which involves vehicle trajectories and satellite images; and (2) data preprocessing and treatment.

4.1. Data Collection

Two primary sources of information were used for model construction, ensuring a comprehensive and multimodal approach to vehicle speed prediction by integrating both spatiotemporal and visual data.

The first source was **GPS Sensor Data**, provided by the Transport and Logistics Center of Universidad Andrés Bello (CTL-UNAB). This dataset recorded the speed of freight vehicles operating in downtown Santiago and included essential attributes such as date, time, latitude, longitude, speed, and vehicle direction. The data spans from January 4th to July 25th, 2020 , covering a total of 157 days , with the exception of April, for which no records are available. Measurements were taken at an hourly frequency between 8:00 a.m. and 7:00 p.m. , resulting in 12 time steps per day . In total, approximately 22 million records were collected, providing a rich temporal dataset that captures variations in traffic conditions across different hours of the day, days of the week, and seasons of the year.

The second source of data consisted of **Satellite Images**, extracted from Google Earth Engine using the Python library ee. These images represented the urban environment with high spatial resolution, capturing road networks, infrastructure, and other environmental features that influence vehicle speed and traffic flow. The

images were specifically selected to align with the GPS sensor locations, ensuring a meaningful correlation between visual and numerical data. The region of interest was defined based on the highest density of GPS records, covering an area of central Santiago with heavy traffic activity.

4.2. Data Preprocessing

Data preprocessing was essential to ensure the quality and representativeness of the information fed into the model. To achieve this, a series of steps were carried out to refine and structure the data effectively.

Geospatial filtering was applied to select only records within the study area, defined between the coordinates [-33.4331, -70.6253] and [-33.4524, -70.6655]. This selection ensured that the dataset accurately represented the urban region of interest and excluded extraneous data points that could introduce noise into the predictions. From an initial dataset of approximately 22 million GPS records, only those relevant to the study area were retained for further processing. Additionally, records with a speed of zero were removed, as they did not contribute useful information for velocity prediction. The dataset was further refined by excluding incomplete data entries, ensuring consistency in the features used by the model.

To enhance spatial representation, the h3 library [16] was employed to transform GPS coordinates into a hexagonal grid, where each hexagonal cell aggregated multiple velocity readings. This conversion optimized spatial segmentation by reducing the distortions introduced by traditional square grids, which often fail to capture continuous spatial relationships effectively. The hexagonal structure provided a more precise spatial representation, improving the model's ability to learn traffic patterns across different areas.

Normalization was performed using MinMax Scaling, which transformed velocity values into a standardized range between 0 and 1. This process improved model stability by ensuring numerical consistency across input features and preventing large disparities in scale that could hinder the learning process. The final training dataset consisted of 1,306 sequences, each containing 12 time steps representing hourly velocity readings, while validation and test sets contained 270 and 272 sequences, respectively. Each sequence corresponded to a grid of 44×15 hexagonal cells, preserving the spatial-temporal structure of the data.

Parallel to the preprocessing of GPS data, satellite images were processed to align with the input requirements of the neural network. Each image was resized to 224×224 pixels, a commonly used dimension in deep learning applications that balances computational efficiency with sufficient detail retention. The images, originally obtained in multiple resolutions, were uniformly adjusted and converted to RGB format to maintain color consistency across different captures. Subsequently, the images were flattened and normalized using MinMax Scaling before being reshaped back into their original format. These preprocessing steps ensured compatibility with the neural network and facilitated multimodal integration by standardizing both spatial and temporal inputs.

5. Results

To evaluate the performance of the proposed model, experiments were conducted on a dataset obtained from GPS sensors and satellite imagery in the city of Santiago, Chile. The evaluation focused on comparing the multimodal model based on HexConvLSTM + MLP with traditional approaches, such as the exclusive use of HexConvLSTM networks. The results were analyzed using standard time series prediction metrics and visualization of errors on spatial maps. A demo code is available in https://drive.google.com/file/d/1yT59eh0v9fuRBtI4y6skpzXWtDXzUWlc/view?usp=sharing

5.1. Experimental setting

5.1.1. Hardware Specifications

The experiments were performed on a virtual machine with the following resources: a **GPU** composed of one Tesla T4 and three Tesla P40, totaling 80 GB of graphics memory; and a **RAM Memory** of 125 GB.

5.1.2. Model Training and Evaluation

The model was trained using a data partitioning scheme with 70% for training, 15% for validation, and 15% for testing. The optimization process focused on minimizing the Mean Squared Error loss (MSE).

To improve training stability, several optimization strategies were implemented. Early stopping was applied to halt training if validation loss did not improve for 15 consecutive epochs. Additionally, we decrease the learning rate by a factor of 0.5 if the loss did not improve within 5 epochs. The model was optimized using the Adam optimizer, with an empirically tuned initial learning rate of 0.0002.

The model's performance was evaluated using standard time-series prediction metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and Coefficient of Determination (R^2)

5.2. CNN Model Selection

To evaluate the predictive capability of different convolutional neural network (CNN) architectures, an extensive

experiment was conducted, comparing multiple models in terms of training and test loss. Widely used architectures in the literature were analyzed, including VGG16, Xception, EfficientNetB7, InceptionV3, and InceptionResNetV2.

Table 1 summarizes the averaged results obtained after three training iterations for each model. Two key metrics are reported: Validation RMSE and Train RMSE, which reflect the model's generalization ability and fit to the training data. EfficientNetB7, Xception, and InceptionV obtained the best performance in terms of validation RMSE.

Table 1
RMSE for CNN Architectures Based on Training and Validation datasets

CNN	Validation RMSE	Train RMSE
Xception	6.27	5.11
VGG16	6.29	5.12
EfficientNetB7	6.25	5.12
InceptionV3	6.27	5.14
InceptionResNetV2	6.30	5.12

Table 2 presents the average test set performance of these three best-performing CNN architectures in Table 1, evaluated over three independent iterations.

The results indicate that InceptionV3 consistently yields the best performance, achieving the lowest values for MAE (2.542), and RMSE (6.109), while matching EfficientNetB7 in terms of coefficient of determination ($R^2 = 0.825$).

Table 2
Average validation set performance over three iterations for the top-3 CNN architectures.

CNN Architecture	MAE	RMSE	$\mathbf{R^2}$
EfficientNetB7	2.547	6.119	0.825
Xception	2.556	6.129	0.824
InceptionV3	**2.542**	**6.109**	**0.825**

5.3. MLP Parameter Selection

To assess the impact of the number of neurons on model accuracy, experiments were conducted by varying the number of units in each layer of the MLP network, considering a total of two layers. Table 3 presents the results of different configurations in terms of training and validation RMSE. Notably, the best configuration from the first layer was used in the second layer.

Validation set results indicate that the combination of 512 and 2048 neurons in the first and second layers, respectively, provides the best balance between accuracy and computational efficiency. Specifically, this configuration achieves a **validation RMSE of 6.26** and a **train RMSE of 5.11**, demonstrating a high generalization capacity without significant overfitting.

Table 3
RMSE Results for the MLP Architecture with Different Neuron Configurations

First Layer				
Neurons	256	512	1024	2048
Validation RMSE	6.50	**6.50**	6.51	6.53
Train RMSE	5.24	**5.24**	5.25	5.26
Second Layer				
Neurons	256	512	1024	2048
Validation RMSE	6.28	6.27	6.27	**6.26**
Train RMSE	5.13	5.12	5.11	**5.11**

5.4. Model Comparison

Table 4 presents the results obtained for each model on both the training and test sets. The reported values correspond to the average performance across three independent runs for each model.

Table 4
Comparison of Metrics for the Evaluated Models

Model	MAE	RMSE	R^2
HexConvLSTM	2.505	6.371	0.805
Inception + HexConvLSTM	2.299	5.857	0.840
MLP + HexConvLSTM	**2.296**	**5.849**	**0.838**

The results show that the multimodal model based on HexConvLSTM + MLP achieves superior performance across all metrics compared to other approaches. Specifically, it reduces the mean absolute error by 8.3% compared to HexConvLSTM and provides a marginal improvement over the CNN + HexConvLSTM model.

5.5. Error Heat-Map Visualization

To visualize the error distribution, heatmaps representing the MAE in each hexagonal grid cell within the study area were generated. Figure 4 illustrates the errors in the test set.

The spatial analysis reveals that the highest errors are concentrated in areas with high variability in vehicle speed, such as intersections and major avenues. In contrast, in regions with more stable traffic flow, the model achieves more accurate predictions.

The conducted experiments validate the hypothesis that combining spatiotemporal and visual data enhances

vehicle speed prediction. The proposed model demonstrates advantages in terms of accuracy and stability, and the results suggest that future improvements could be achieved by incorporating additional dynamic data, such as weather conditions and real-time traffic events.

Figure 4: Heatmap of MAE in the Study Area for the Test Set .

6. Discussion

The results obtained in this study confirm that the proposed multimodal model, based on the combination of HexConvLSTM and MLP, outperforms conventional approaches in vehicle speed prediction. In terms of MAE and RMSE, the multimodal model achieved a significant error reduction compared to HexConvLSTM and CNN, validating the hypothesis that integrating satellite imagery improves predictive accuracy.

The comparative analysis demonstrates that incorporating visual information from the urban environment through satellite images allows the model to capture spatial patterns that traditional models do not consider. The proposed architecture improves predictions in areas with regular traffic conditions, although challenges were observed in maintaining accuracy during abrupt speed fluctuations caused by unpredictable events, such as accidents or sudden congestion.

Additionally, the use of hexagonal grids in the HexConvLSTM branch offers a potentially improved spatial representation of GPS data, mitigating some of the distortions commonly associated with square-grid structures. This feature has been crucial to ensuring model stability in urban traffic analysis. A somewhat *surprising* observation was that the MLP outperformed more advanced CNN-based architectures such as Inception. This result is likely due to the static nature of the satellite image, where convolutional models may not fully exploit their inherent translational invariance. Given the relatively small resulting feature map size (44×15), the advantages of convolutional operations become less pronounced, reducing the expected performance gap between CNNs and fully connected networks.

Previous studies in the literature have explored traffic prediction using LSTM, CNN, and hybrid models with geospatial data. However, most of them do not explicitly consider the integration of sensor data with satellite images in a multimodal framework. Compared to previous works, our model offers a more comprehensive integration of spatiotemporal information. Unlike approaches that rely solely on historical traffic data, our model incorporates the geographic context of the road environment, providing a more dynamic and context-aware prediction.

Despite the positive results, our method has several limitations that open promising research avenues. First, generalisability is still unproven: the model was trained solely on downtown Santiago traffic, so its behaviour in cities with different network layouts or demand patterns must be validated. Second, prediction accuracy may deteriorate where only low-resolution imagery is available or where rapid infrastructure changes outpace the satellite update cycle, calling for dynamic image-quality checks. Finally, the hexagonal tessellation—though uniform and rotation-invariant—aggregates roads of different functional classes and directions within a single cell, blurring lane- or direction-specific congestion (e.g., a stalled freeway lane next to a free-flow local road). Consequently, the current design is best suited to area-level tasks such as fleet dispatch or hotspot screening; applications needing direction separation should combine the grid with road-graph or edge-level GNN features, an integration we leave for future work.

7. Conclusions

This study developed a multimodal predictive model that integrates spatiotemporal data from GPS sensors with satellite imagery, leveraging HexConvLSTM and MLP neural networks. The model was trained and evaluated using traffic data from downtown Santiago, demonstrating significant improvements in prediction accuracy compared to conventional approaches that rely solely on historical data.

Overall, the results indicate that incorporating satellite imagery into traffic prediction models enhances the accuracy of vehicle speed estimations. Specifically, the HexConvLSTM + MLP multimodal model achieved lower Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) than traditional methods, highlighting the benefits of combining spatial and temporal information. Furthermore, the proposed methodology is adaptable to other urban environments, provided that data preprocessing and hyperparameter tuning are adjusted accordingly.

For future work, we aim to assess the generalization of the model across different urban settings with varying traffic conditions. Additionally, we plan to integrate meteorological data, urban events, and social media in-

formation to improve the model's adaptability to sudden traffic fluctuations. From a technical perspective, we will explore attention-based models and Graph Neural Networks (GNNs) to better capture complex relationships within geospatial data. Furthermore, we intend to incorporate the YOLO network for satellite image processing, enabling more precise identification of road structures, vehicle densities, and other key environmental features that influence traffic flow. This enhancement will refine the integration of visual data, further improving the model's predictive performance in dynamic urban scenarios.

Acknowledgments

B. Peralta and H. Lobel appreciate the support of the National Center for Artificial Intelligence CENIA FB210017, Basal ANID.

References

[1] D. Schöder, F. Ding, J. K. Campos, The impact of e-commerce development on urban logistics sustainability, Open Journal of Social Sciences 4 (2016) 1–6. URL: https://tancuarku.com/lander/tancuarku.com/index.php?paperid=64089&_=%2Fjournal%2Fpaperinformation%23Z0x%2FkvlZXYFNnfiVfd428GUNP8E%3D. doi:10.4236/jss.2016.43001.

[2] S. Afandizadeh, S. Abdolahi, H. Mirzahossein, Deep learning algorithms for traffic forecasting: A comprehensive review and comparison with classical ones, Journal of Advanced Transportation 2024 (2024) 1–30. URL: https://doi.org/10.1155/2024/9981657. doi:10.1155/2024/9981657.

[3] R. Rajha, S. Shiode, N. Shiode, Improving traffic-flow prediction using proximity to urban features and public space, Sustainability 17 (2025) 68. URL: https://www.mdpi.com/2071-1050/17/1/68. doi:10.3390/su17010068.

[4] V. Stienen, D. D. Hertog, J. C. Wagenaar, J. F. Zegher, Better routing in developing regions: Weather and satellite-informed road speed prediction, CentER Discussion Paper (2023).

[5] D. Guo, C. Zhao, H. Shuai, J. Zhang, X. Zhang, Enhancing sustainable traffic monitoring: Leveraging nanosight–yolo for precision detection of micro-vehicle targets in satellite imagery, Sustainability 16 (2024) 7539.

[6] H. Cheng, H. Lei, S. Zourlidou, M. Sester, Traffic control recognition with an attention mechanism using speed-profile and satellite imagery data, in: ISPRS - Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci., volume XLIII-B4, 2022, pp. 287–293.

[7] K. Chowdhury, M. Sarwat, Deep learning with spatiotemporal data: A deep dive into geotorchai, in: Proceedings of the 40th IEEE International Conference on Data Engineering (ICDE), IEEE, 2024.

[8] W. Wang, R. D. J. Samuel, C.-H. Hsu, Prediction architecture of deep learning assisted short-long term neural network for advanced traffic critical prediction system using remote sensing data, European Journal of Remote Sensing 54 (2021) 65–76.

[9] M. Adamiak, Y. Grinblat, J. Psotta, N. Fulman, H. Mazumdar, S. Tang, A. Zipf, Deep learning enhanced road traffic analysis: Scalable vehicle detection and velocity estimation using planetscope imagery, arXiv preprint arXiv:2410.14698 (2024). URL: https://arxiv.org/abs/2410.14698.

[10] A. Sheehan, A. Beddows, D. C. Green, S. Beevers, City scale traffic monitoring using worldview satellite imagery and deep learning: A case study of barcelona, Remote Sensing 15 (2023). URL: https://www.mdpi.com/2072-4292/15/24/5709. doi:10.3390/rs15245709.

[11] A. A. Kashyap, S. Raviraj, A. Devarakonda, S. R. N. K, S. K. V, S. Bhat, Traffic flow prediction models – a review of deep learning techniques, Cogent Engineering 9 (2021). URL: https://doi.org/10.1080/23311916.2021.2010510. doi:10.1080/23311916.2021.2010510.

[12] A. Mystakidis, P. Koukaras, C. Tjortjis, Advances in traffic congestion prediction: An overview of emerging techniques and methods, Smart Cities 8 (2025) 25. URL: https://www.mdpi.com/2624-6511/8/1/25. doi:10.3390/smartcities8010025.

[13] X. He, W. Jia, Hexagonal structure for intelligent vision, in: 2005 International conference on information and communication technologies, IEEE, 2005, pp. 52–64.

[14] F. Bahamondes, B. Peralta, O. Nicolis, A. Bronfman, Á. Soto, Convlstm neural network based on hexagonal inputs for spatio-temporal forecasting of traffic velocities, in: Proceedings of the 3rd International Workshop on Spatio-Temporal Reasoning and Learning (STRL 2024) co-located with the 33rd International Joint Conference on Artificial Intelligence (IJCAI 2024), 2024, pp. 45–55. URL: https://ceur-ws.org/Vol-3827/paper5.pdf.

[15] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, W.-c. Woo, Convolutional lstm network: A machine learning approach for precipitation nowcasting, Advances in neural information processing systems 28 (2015).

[16] I. Brodsky, H3: Uber's hexagonal hierarchical spatial index, https://eng.uber.com/h3/, 2018. Available from Uber Engineering website. Accessed: 22 June 2019.