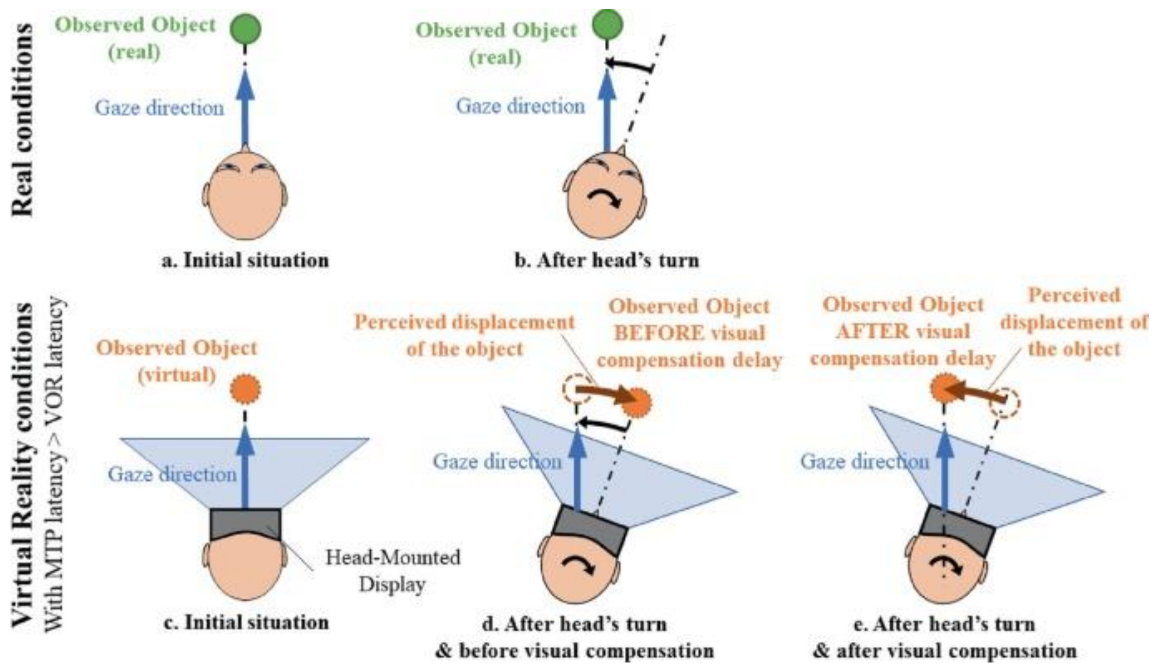
IMU Gyro Prediction for AR Headset Motion

Reducing Latency via Short-Horizon Forecasting
Forecasting

Joshua Shabanov
Omer Reuven

Background

Virtual and Augmented Reality (VR/AR) systems require extremely low latency for a comfortable, immersive experience. Predicting future head and gaze motion reduces perceived lag by enabling scene pre-rendering, but due to limited resources on the goggles, lightweight models are essential.



Motivation: Enhancing VR/AR Immersion

1

Precise Motion Forecasting

Our primary objective is to accurately predict head motion for 1 to 4 samples ahead, equating to a critical time window of approximately 20 to 80 milliseconds (52Hz).

2

Elevating User Experience

By proactively accounting for future motion, we aim to significantly enhance user comfort, minimize cybersickness, and deepen the overall sense of immersion in virtual environments.

3

Model Selection Optimization

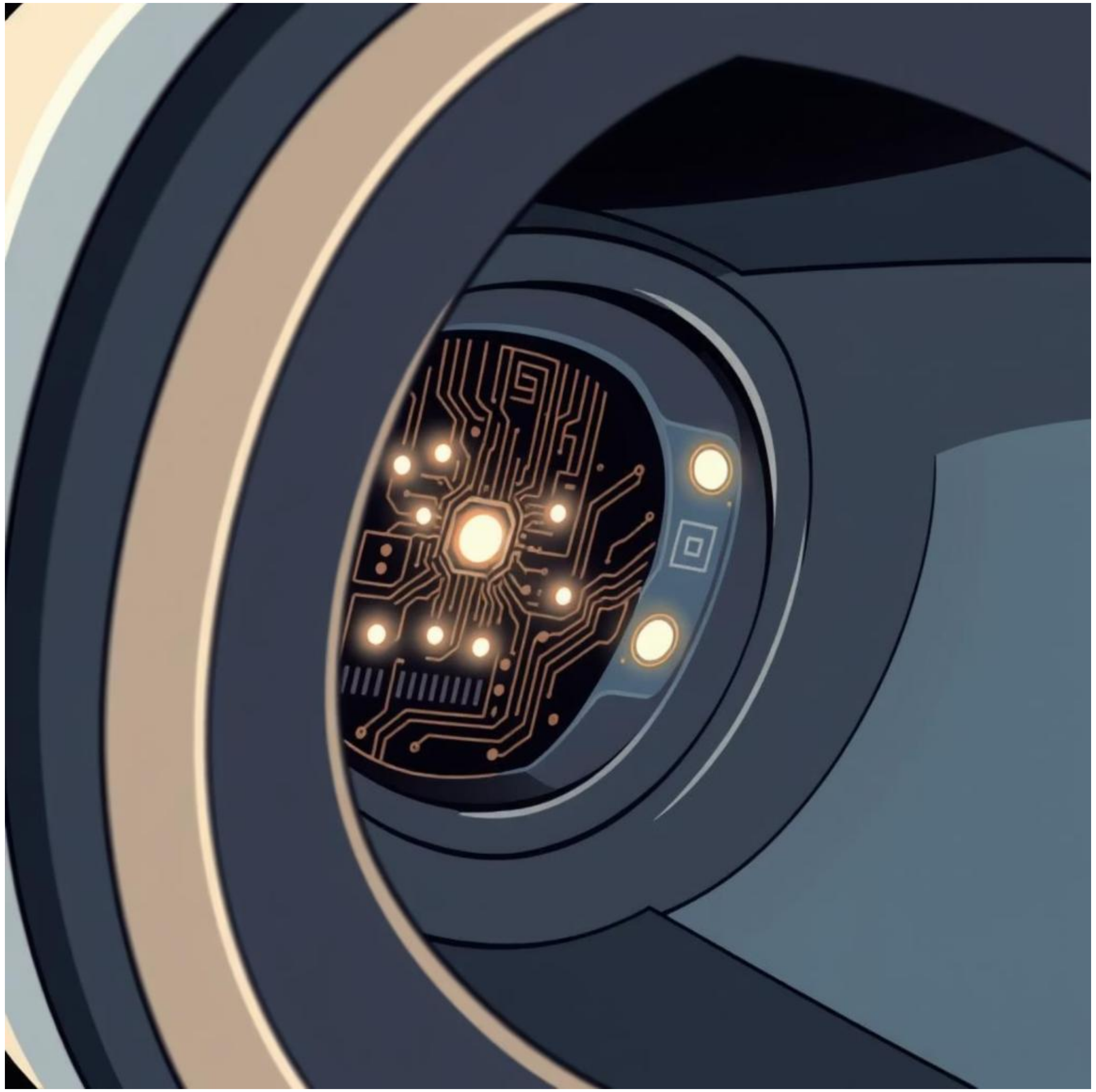
A core focus of this research is to identify the most effective and efficient model architectures specifically tailored for short-horizon motion prediction in real-time VR/AR applications.

The actual Glasses by Everysight

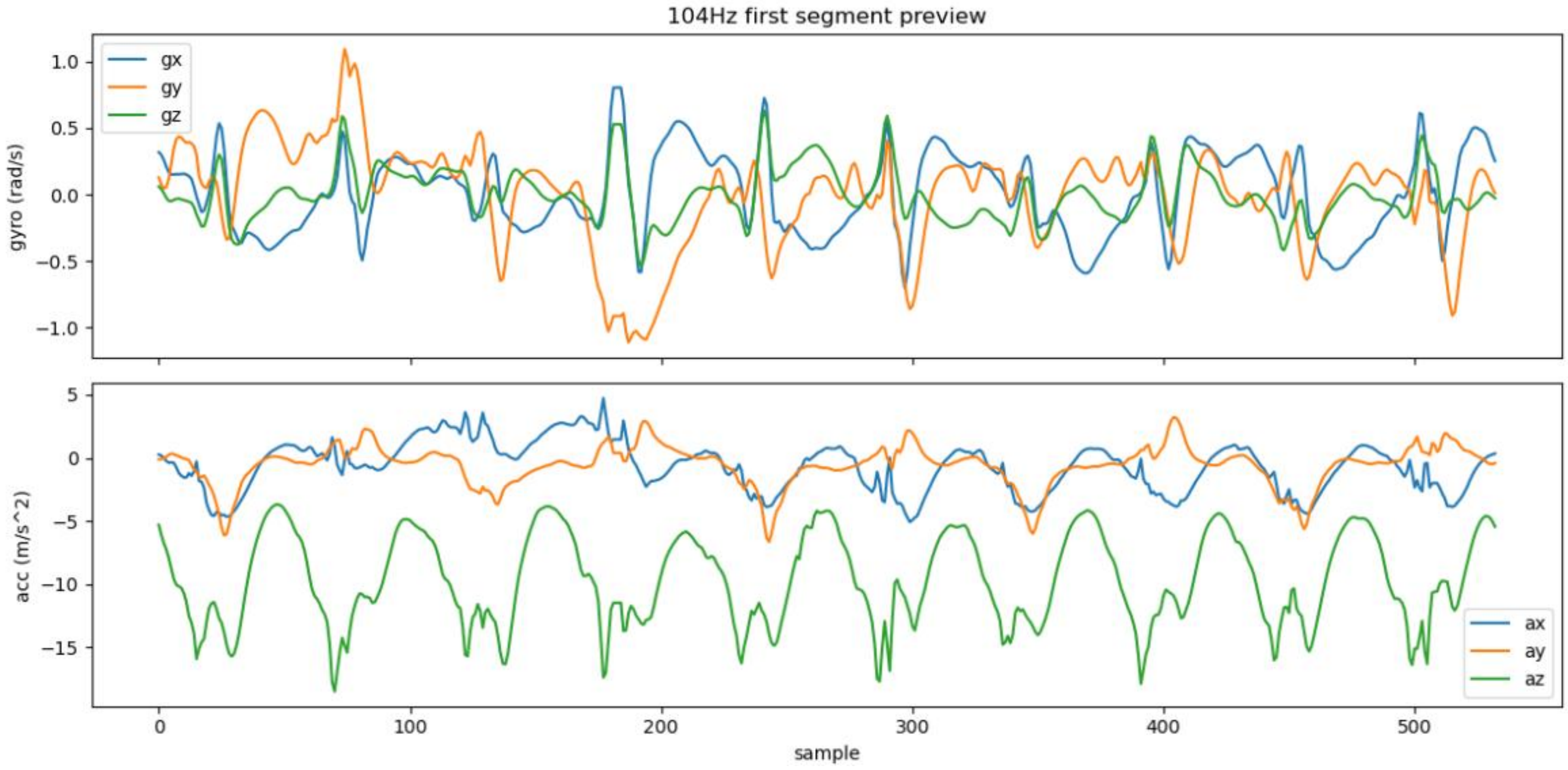
Data Acquisition and Processing

Our dataset is derived from Inertial Measurement Unit (IMU) recordings, captured directly from an AR headset for test purposes. This provides authentic user motion data in multiple scenarios.

- Key channels include Gyroscope data (gx, gy, gz) for rotational velocity, rotational velocity, with potential inclusion of accelerometer data for data for linear acceleration.
- We have two sets of 52Hz and 104 Hz recordings.
- Data is mostly continuous but have gaps, segmentation is necessary necessary



Visualizing an IMU Data Segment



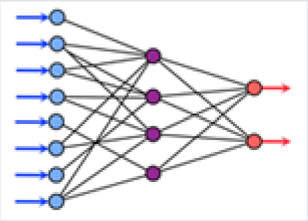
Models Architectures Under Comparison

We selected a diverse set of deep learning models, each offering unique strengths for sequence prediction and temporal pattern temporal pattern recognition.

CNN (Convolutional Neural Network)

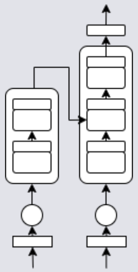
Network)

Chosen for its efficiency in extracting spatial features from sequential data, offering fast, low-latency low-latency inference due to parallel processing capabilities.



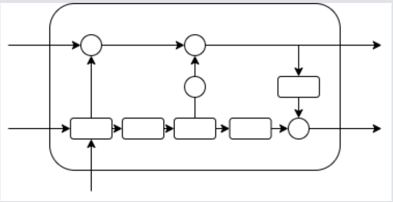
Transformer

Leverages self-attention to model relationships between all time steps in the IMU sequence, enabling accurate long-range dependency capture and flexible weighting of past sensor readings



LSTM

Long Short-Term Memory networks, designed for effective temporal modeling and handling vanishing/exploding gradients in shorter sequences.



Chosen Parameters

Transformer

Linear	**1x12x128**	**in=6, out=128**	**X2**
Pos Encoding	**1x12x128**		
Dropout	**1x12x256**		
LayerNorm	**1x12x128**		
Linear	**1x12x256**		
Dropout	**1x12x128**		
Linear	**1x12x256**		
Dropout	**1x12x128**		
LayerNorm	**1x12x128**		
Linear	**1x12x256**		
LayerNorm	**1x12x128**		
Linear	**1x3**	**in=128, out=3**	

LSTM

LSTM	**1x12x128**	**hidden=128, layers=2**
Linear	**1x3**	

CNN

Layer	**Output Shape**	**Activation**	
Conv1d	1x64x12	GELU	**X3**
GELU	1x64x12	GELU	
BatchNorm1d	1x64x12	GELU	
AdaptiveAvgPool1d	1x64x1		
Flatten	1x64		
Linear	1x3		

Rigorous Comparison Methodology

To ensure a fair and insightful evaluation, each model underwent a standardized training and testing protocol.

Identical Dataset Training

Each selected model was trained trained on precisely the same datasets, ensuring a level playing playing field for performance comparison and minimizing data-data-related biases.

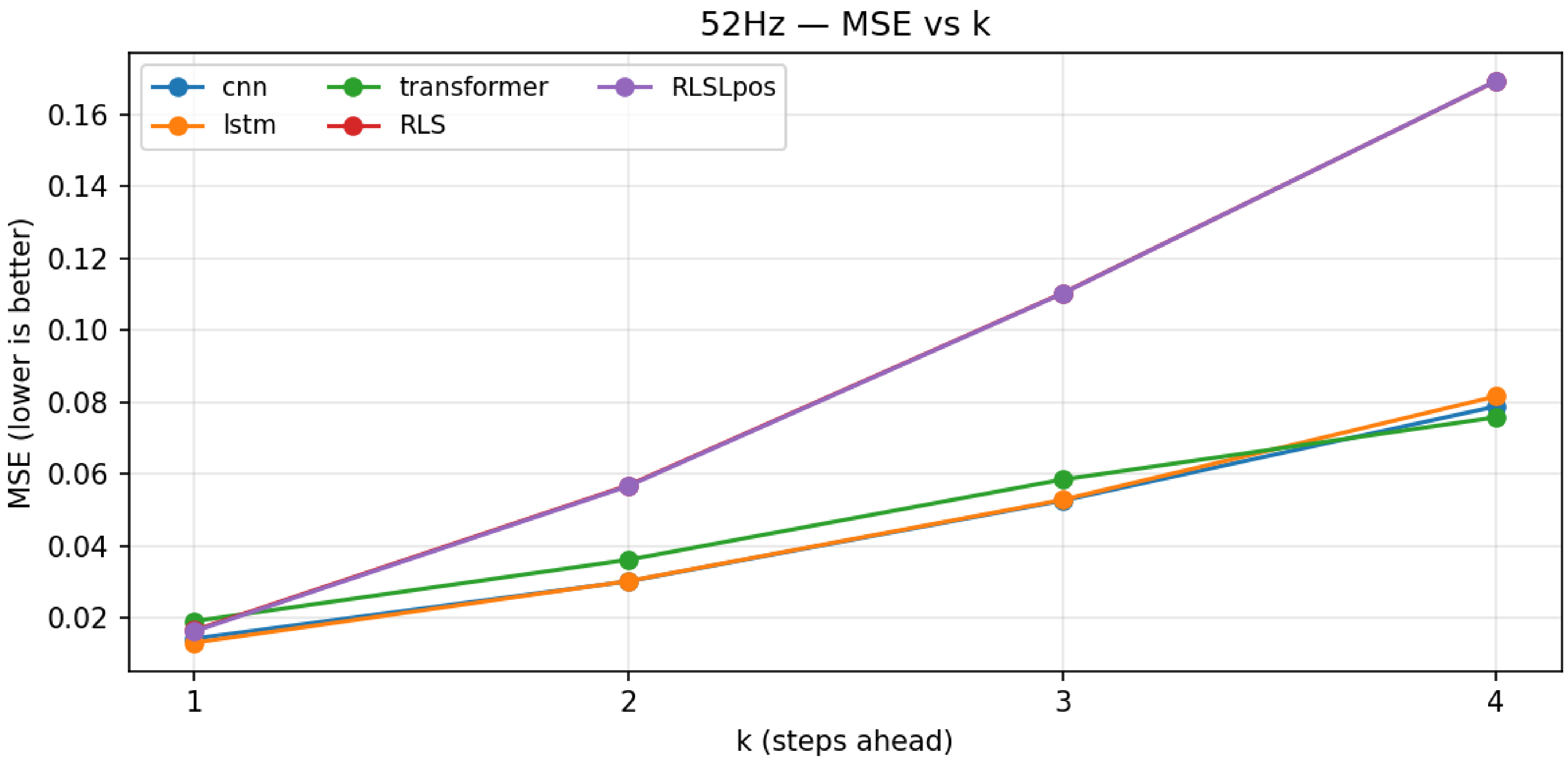
Validation MSE Evaluation

Performance was quantitatively quantitatively assessed using Mean Mean Squared Error (MSE) on test test data across prediction horizons horizons of 1, 2,3 and 4 steps, providing a clear metric of accuracy. accuracy.

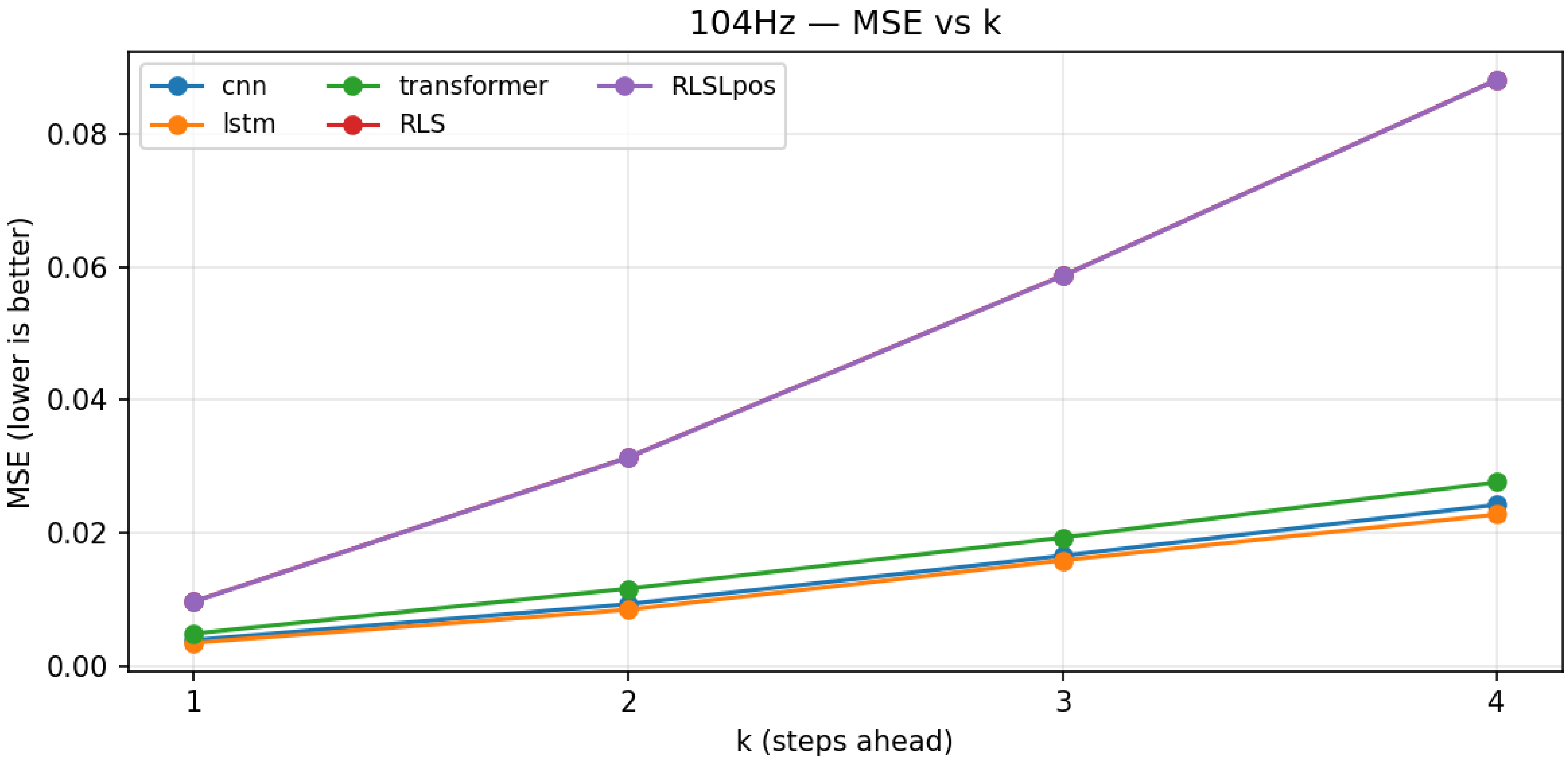
Inference Time Measurement

Crucially, the real-time inference time time for each model was meticulously meticulously recorded. This metric is metric is paramount for determining determining their practical feasibility in feasibility in latency-critical VR applications.

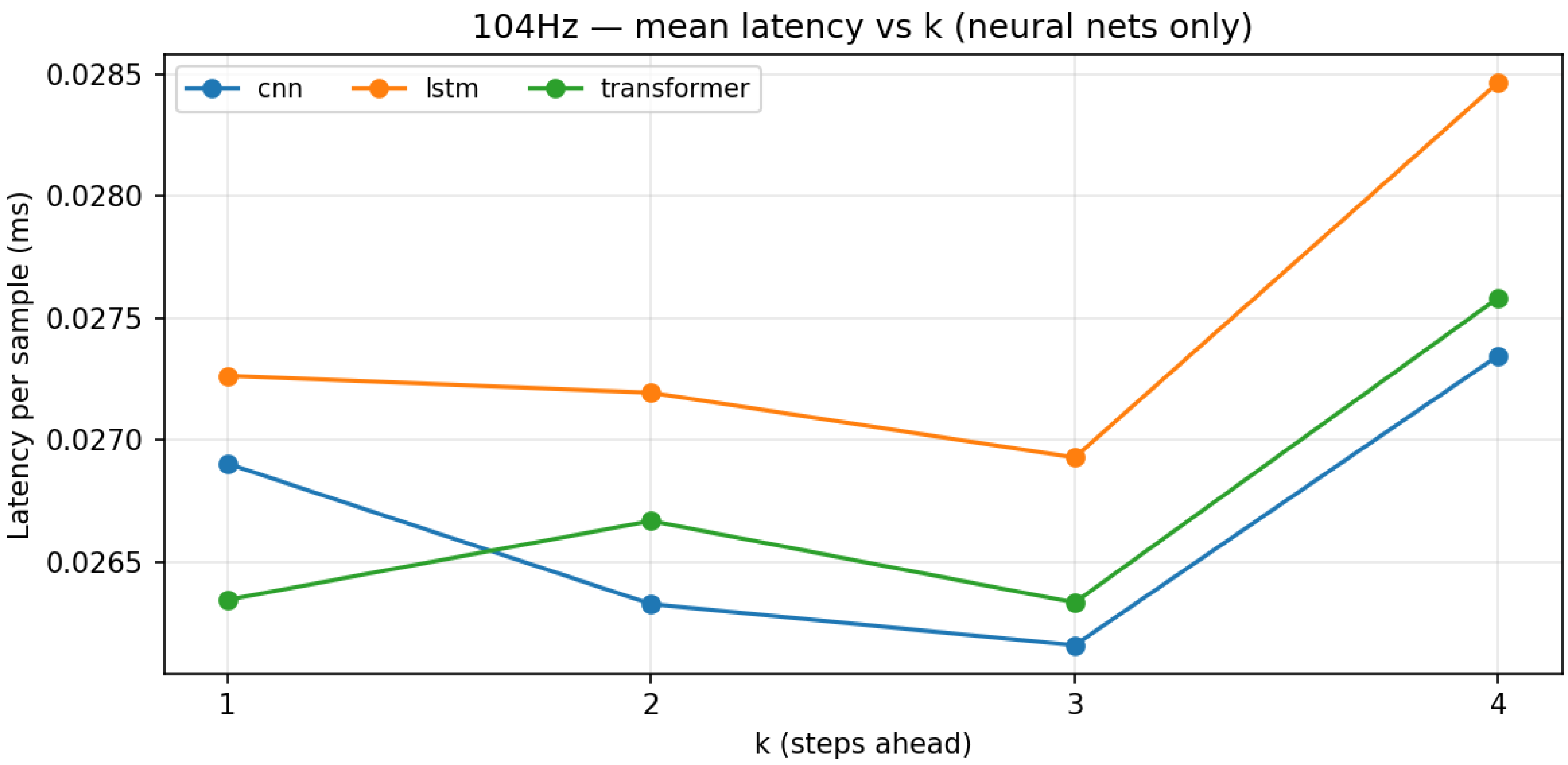
Results: Loss



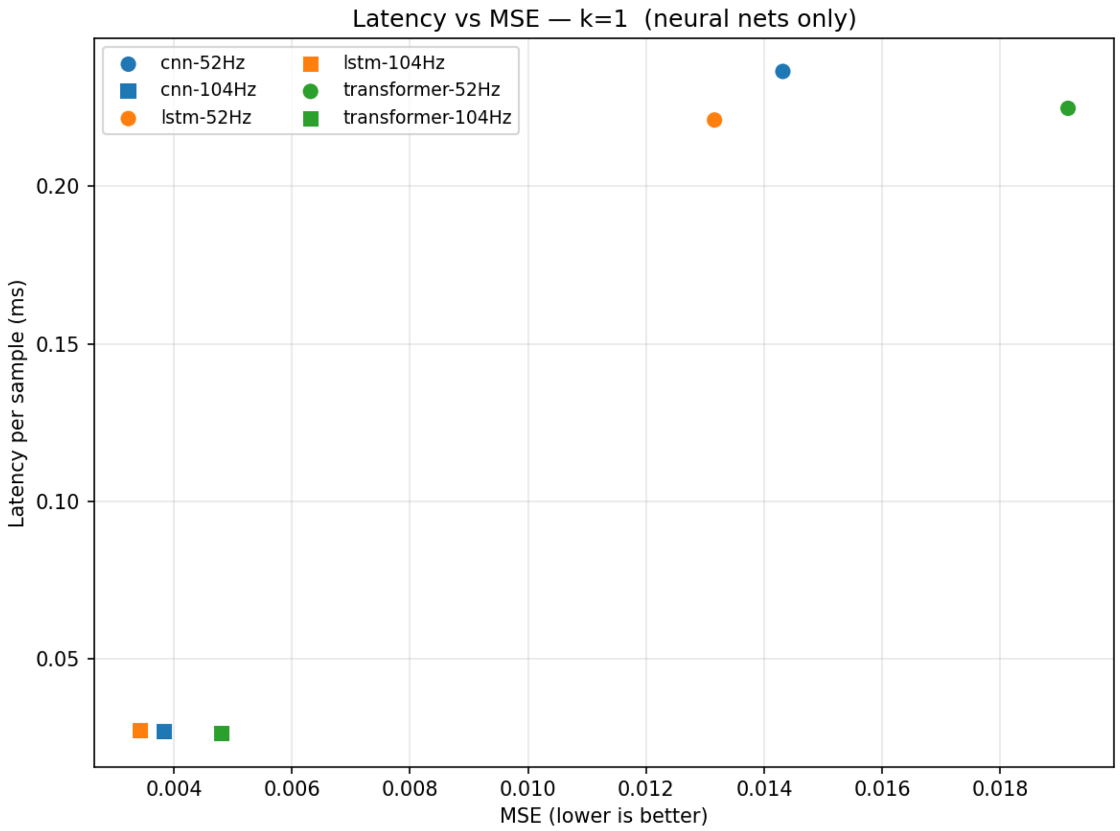
Results: Loss

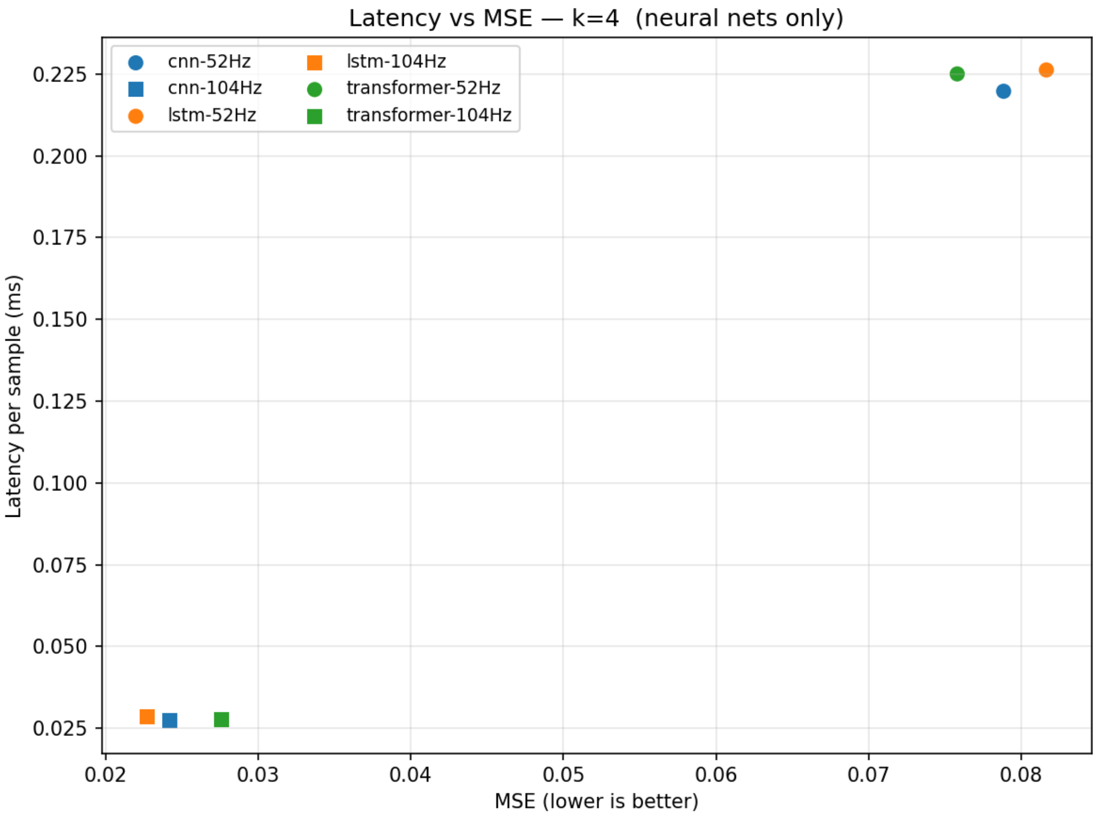


Results: Latency

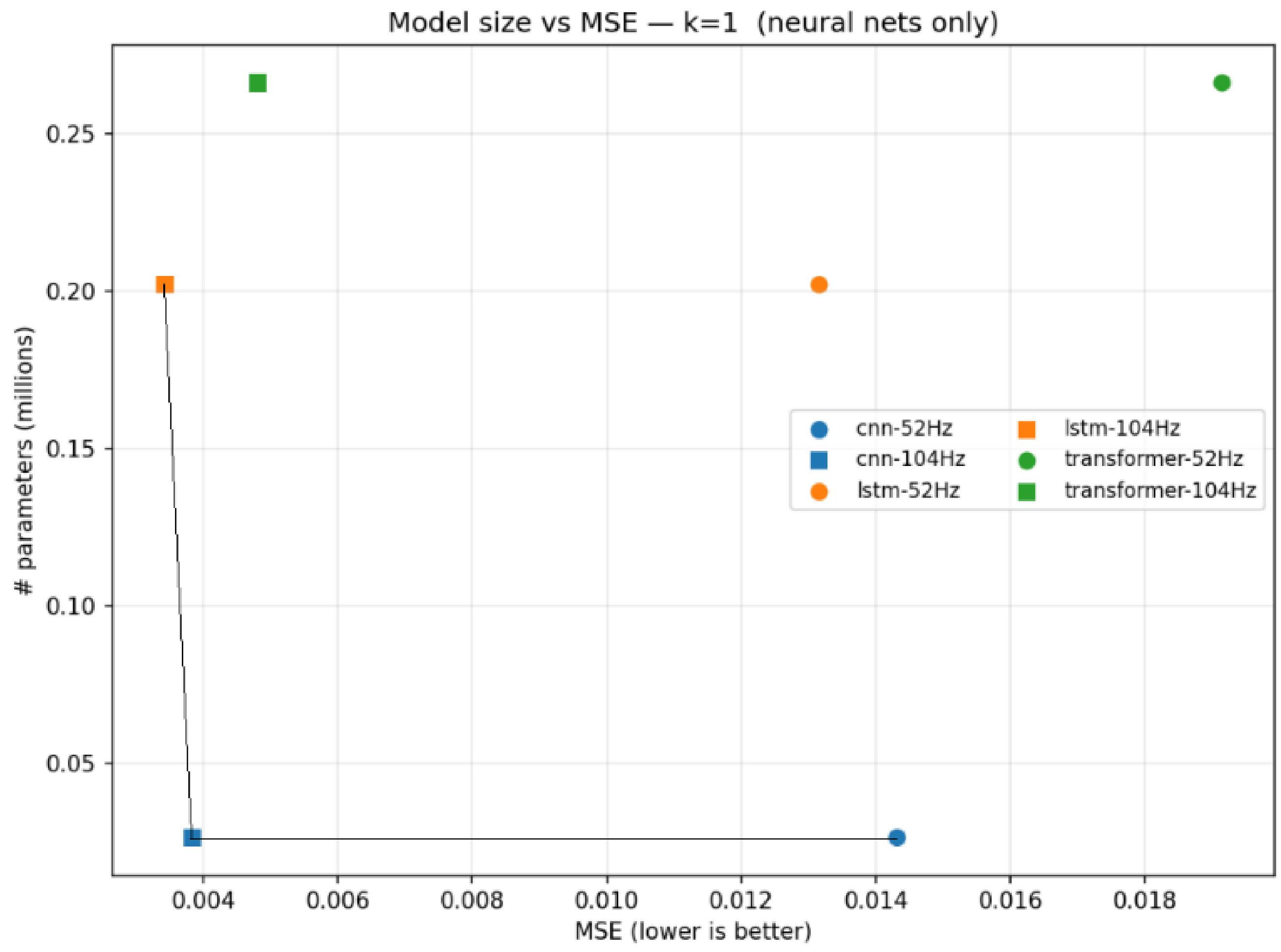


Results: Latency

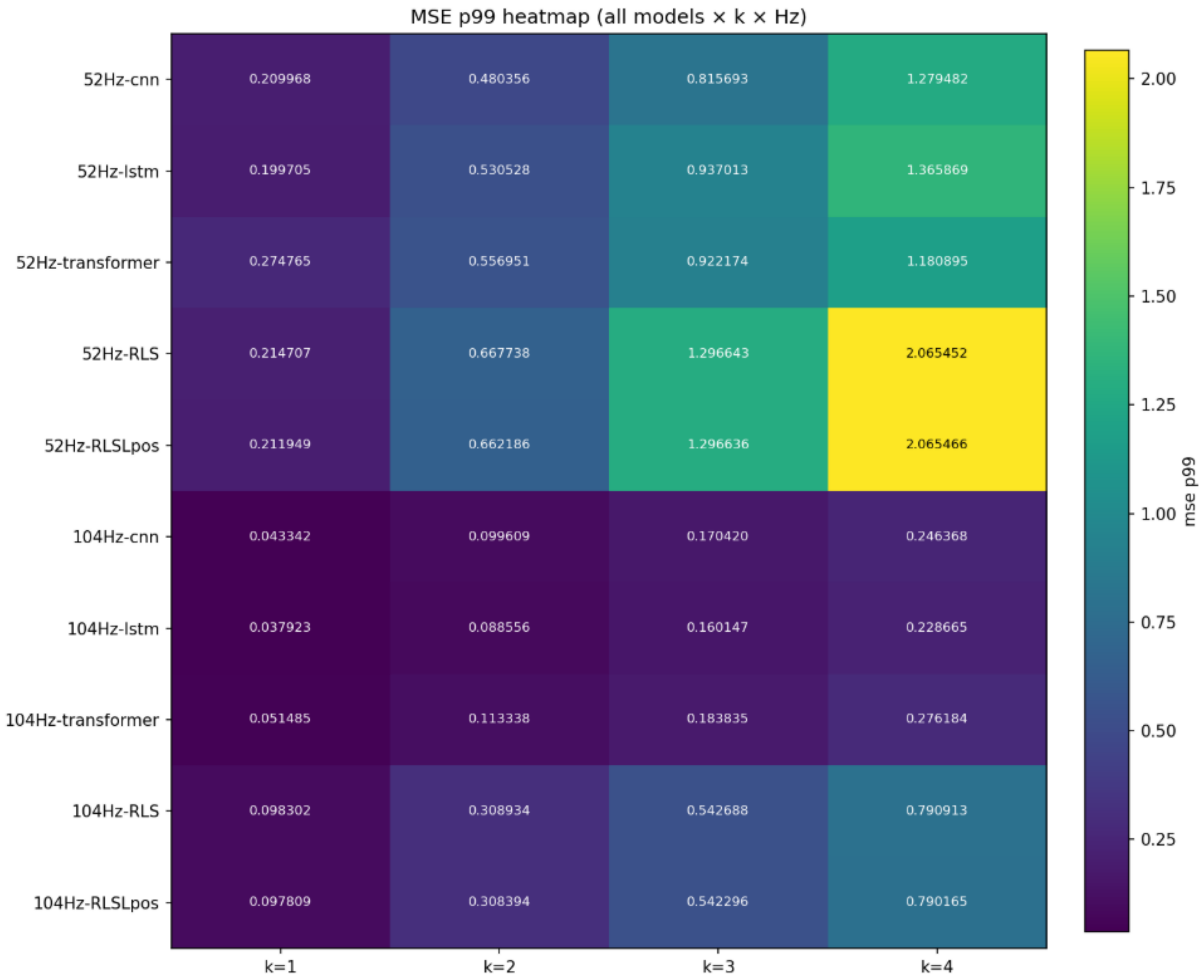




Results: parameters



Results: Edge Cases



Conclusion

- **LSTM in 104Hz achieved superior results in loss reduction, slightly better than CNN with similar latency.**
- **However, takes a lot of memory to store relative to traditional RLS method and CNN.**
- **RLS is less consistent in loss on edge cases while both LSTM and CNN in 104Hz are far more stable.**
- **LSTM and CNN are both viable options for replacing classic RLS choosing between them depends on the device's memory limits.**

Future Directions & Optimization

Hybrid Architectures

Explore combining CNN or RLS and RNN strengths for enhanced feature extraction and temporal modeling.

Probabilistic Approach

Estimate the confidence or uncertainty associated with each predicted sample to improve decision-making.

Horizon Expansion

Enhance prediction to cover longer horizons and interpolate values for intermediate steps.

Hardware Integration

Conduct rigorous real-time testing on target VR/AR hardware hardware platforms for practical validation.

Thank You

Questions?