Spatial-FWP Mamba: A Spatiotemporally Decoupled Memory Architecture with a Causal Trajectory Prior and Multiscale Frequency Forgetting

Dian Yu

Abstract

Sequence models compress history in token order, but sequence order does not always agree with physical proximity: two observations separated by many steps may come from the same location, whereas consecutive observations may be separated by a large spatial displacement. For scanning tasks with physical coordinates in two or three dimensions, we propose Spatial-FWP Mamba, which maintains temporal-semantic memory separately from coordinate-addressed spatial memory. The Mamba branch models temporal dependencies and semantic representations. The Spatial-FWP branch writes observation summaries into a fixed-size associative matrix and uses the current position key to recall historical information from the same or nearby locations.

The spatial branch relies on an explicit assumption: the scan remains inside a bounded workspace known in advance, and after coordinate normalization and frequency configuration, multiscale sinusoidal position keys exhibit reliable local similarity within that domain. Under this condition, position-triggered recall aggregates relevant history with one matrix-vector product, without constructing a sliding window or storing and traversing all previous coordinates. To regulate updates to the fixed-capacity memory, we introduce a causal trajectory-prior matrix, referred to as the Prior Hologram (PH), that records historical coordinates only. Querying PH with the current position produces a multiscale summary of the historical position distribution around that point. This query result and the current displacement are processed by a lightweight geometry-only gating network to produce a pair-shared diagonal forgetting gate and a scalar write gate. Observation content is used only as the value stored in memory and does not participate in spatial gating. Each spatial update applies forgetting first, then reads the historical state, and finally writes the current observation. The recurrence can be implemented as a PH prefix scan followed by a diagonal-affine FWP prefix scan. We present the formulation, complexity, applicability boundaries, and an evaluation protocol without reporting unverified performance figures.

Keywords: Mamba; fast weight programming; coordinate-addressed associative memory; spatial working memory; spatially selective recall; causal trajectory prior; multiscale positional encoding; multiscale forgetting; parallel prefix scan; spatiotemporal decoupling

1 Introduction

State-space models such as Mamba compress long-range context into a fixed-dimensional hidden state through input-dependent selective updates [3]. This mechanism naturally models what has occurred in sequence order, but it does not automatically convert temporal distance into physical distance. In embodied motion, variable-density image scanning, maze traversal, free-form canvas generation, and three-dimensional point-cloud scanning, each token may also carry a physical coordinate. Observations separated by many sequence steps can therefore correspond to the same location, while adjacent tokens may represent a large spatial jump. If a single state is updated only in sequence order, temporal semantics and physical proximity must share one hidden dynamics and may interfere.

We add a coordinate-addressed spatial module alongside Mamba. Mamba continues to maintain a temporal-semantic state, whereas Spatial-FWP maintains a fixed-size spatial associative matrix that answers a different question: what was previously observed at this location or nearby? Coordinates are assumed to be supplied by a dataset, localization system, odometry pipeline, or another external component. The module neither estimates pose nor reconstructs a complete metric map. Its role has two parts:

1. **Position-triggered recall.** Similarity between the current and historical position keys determines how much each stored observation contributes to the readout. Nearby positions are designed to produce stronger responses, while distant positions remain weaker within the configured workspace. FWP performs this aggregation through accumulated key-value associations without pointwise neighborhood traversal.

2. **Multiscale forgetting with geometry-only write control.** A current-position query of the causal trajectory prior produces a multiscale cue about nearby visitation history. This cue and the current displacement jointly determine the retention of different position-frequency channels and the overall write strength at the current step. Because the spatial gates do not read observation semantics, stored content and geometric memory evolution remain separate in the control path.

The two branches interact only through late fusion at the current prediction head. Spatial readouts are not fed into the next Mamba state, and the Mamba hidden state is not used to generate spatial gates. Since the forgetting and write gates depend only on the causal coordinate prefix rather than on the main FWP matrix, spatial computation can be organized as two serial associative scans: the first computes PH states and gates, and the second updates the main spatial memory. Constant per-step complexity here means constant with respect to processed sequence length; the actual cost still depends on the dimensions of the position keys and observation values.

The main contributions are:

- a fixed-capacity spatial memory alongside Mamba that uses coordinate similarity for position-triggered recall;
- geometry-only dual gating based solely on historical coordinates and current displacement, with pair-shared multiscale forgetting and scalar write control;
- a causal read-before-write update and a two-stage associative prefix-scan implementation with explicit complexity and causality constraints; and
- a distinction from explicit maps, sliding-window retrieval, and conventional sequence states, together with an evaluation protocol for spatial recall, gating, and runtime efficiency.

2 Related Work

2.1 State-Space Models and Sequence Modeling

State-space models (SSMs) compress sequence history through recurrent state updates. HiPPO established a theoretical basis for finite-dimensional projections of continuous histories, and S4 developed structured SSMs for efficient long-sequence modeling [1, 2]. Mamba makes state updates depend on the current input, while Mamba-2 simplifies computation through structured state-space duality and improves hardware utilization [3, 4]. These states evolve primarily in input order. Point Cloud Mamba and STREAM apply SSMs to point clouds and sparse geometric data [10, 11], showing that scan order, sampling interval, and physical geometry affect state updates. We preserve Mamba's temporal-semantic role and add an independent coordinate-addressed state.

2.2 Fast Weights and Associative Memory

Fast-weight models use matrices that change over a sequence as short-term plastic memory [5, 6]. In linear attention, key-value outer products accumulate over time, so one matrix-vector product can aggregate historical associations [7]. Fast Weight Programmers interpret this process as memory writing and reading [8]; Gated DeltaNet further studies gating and error-corrective updates [9]. We use additive FWP for fixed-capacity associative storage rather than to reproduce an attention window: coordinate keys provide addresses, observation summaries are stored as values, and query cost does not grow with the number of historical coordinates. Diagonal multiscale decay keeps interval composition in a low-cost closed form and avoids main-memory-dependent dense modifications.

2.3 Continuous Coordinates and Fourier Features

Random Fourier features approximate translation-invariant kernels through finite-dimensional sinusoidal inner products, providing explicit kernel features over continuous coordinates [12]. Fourier features also improve the representation of high-frequency variation in networks with low-dimensional coordinate inputs [13]. We do not use a learnable coordinate network. Instead, the two- or three-dimensional scan is assumed to remain inside a bounded workspace known in advance. Frequencies, directions, and weights are fixed from the workspace scale and the smallest neighborhood scale that the task is intended to distinguish, and the resulting Fourier key acts directly as the FWP address. Unlike simply concatenating a position encoding with a network input, the key inner product directly determines historical recall weights. We therefore discuss locality only inside the configured workspace and do not claim unbounded spatial extrapolation.

2.4 Spatial Representation and Geometric Sequences: Relation to SLAM

Simultaneous localization and mapping (SLAM) estimates pose while constructing an environmental representation. Systems may maintain sparse landmarks, metric maps, occupancy grids, or pose graphs. Cognitive Mapping and Planning, Neural Map, MapNet, and Active Neural SLAM also use explicit or semi-explicit spatial memory for learned navigation [14, 15, 16, 17]. These methods and Spatial-FWP all use location to organize history, but solve different problems.

Spatial-FWP assumes that the current coordinate is given. It neither performs localization nor replaces SLAM. It stores observation summaries acquired along the visited trajectory and answers what was previously observed near the current location:

- **Implicit memory rather than an explicit map.** The model does not allocate a separate slot to every grid cell, voxel, or landmark. It compresses continuous coordinates and observations into a fixed-dimensional associative matrix, so it cannot directly output a measurable occupancy map or preserve every historical sample exactly.
- **Matrix readout rather than neighborhood search.** The read $(M_t^-)^\top k_t$ uses linear superposition of historical outer products to aggregate all retained associations in one matrix-vector multiplication. Inference requires no sliding window, neighbor list, or Top-K coordinate search, and its cost is independent of the number of visited locations.

- **Visited experience rather than a complete environment model.** The matrix accumulates observations only along the visited trajectory and allocates no explicit storage to unvisited regions. Continuous position responses may still produce nonzero interpolation or Fourier sidelobes at unvisited coordinates, and similar positions may interfere in the fixed-dimensional representation. This lossy superposition is the trade-off for fixed capacity and stable long-sequence cost.

3 Spatial-FWP Architecture and Formulation

Spatial-FWP Mamba consists of a temporal-semantic stream and a spatial-memory stream. Mamba maintains the hidden state h_t, whereas Spatial-FWP maintains the spatial associative matrix M_t. Both branches receive the same observation but serve different purposes. Mamba uses it to update temporal semantics. The spatial branch uses coordinates to generate addressing keys and control signals, while treating the projected observation v_t only as the payload to store. Its operations comprise positional encoding, PH lookup, geometry-only gating, historical decay, position-triggered recall, and current-step writing. Figure 1 summarizes the architecture. The modules indicate data dependencies rather than requiring every local projection to run serially. Quantities such as k_t, v_t, and δp_t depend only on current inputs and can be prepared in parallel, but the gates must wait until the PH scan has produced PY_t; they cannot bypass the trajectory prior.

3.1 Multiscale Positional Keys in a Bounded Workspace

At time step t, the model receives a physical two- or three-dimensional coordinate p_t. Let Ω be a bounded workspace whose coordinate limits are known in advance and which the scan trajectory does not leave. Along coordinate axis r, we use

$$\bar{p}_{t,r} = \frac{p_{t,r} - b_r^-}{b_r^+ - b_r^-} \in [0,1] \tag{1}$$

to place all axes on a common scale. This rescaling does not discretize the continuous space into a grid. The trajectory may continue for an arbitrary number of steps inside Ω; repeated visits to the same physical location deliberately produce the same address and are not a temporal periodicity error.

The position key $k_t = K(p_t)$ is constructed from fixed multiscale sine–cosine pairs and requires no trainable coordinate encoder. Let $a_j > 0$ denote the weight of frequency ω_j. A direct definition is

$$K(p) = \frac{1}{\sqrt{\sum_{j=1}^{m} a_j}} \big[\sqrt{a_j}\cos(2\pi\omega_j^\top \bar{p}), \sqrt{a_j}\sin(2\pi\omega_j^\top \bar{p})\big]_{j=1}^{m}. \tag{2}$$

Each frequency contributes one sine–cosine pair, so $d_k = 2m$, and the normalization gives $\|K(p)\|_2 = 1$. Lower frequencies describe broad spatial relations, while higher frequencies add local discrimination. In two or three dimensions, each scale may use several fixed directions so that the encoding is not sensitive to only one coordinate axis. After normalization, low-frequency pairs with periods longer than the workspace span provide a coarse spatial basis; higher frequencies are then added according to the smallest spatial scale that the task should distinguish. The scales need not share one period, and their weights can limit the influence of fine-scale sidelobes. Since Ω is known, these parameters are fixed once when the model is configured and remain unchanged no matter how long the scan continues within the workspace.

The association strength between two positions is determined by the inner product of their keys:

$$\kappa(p,q) = K(p)^\top K(q) = \frac{\sum_{j=1}^{m} a_j \cos\big(2\pi\omega_j^\top(\bar{p}-\bar{q})\big)}{\sum_{j=1}^{m} a_j}. \tag{3}$$

This expression depends only on coordinate displacement. If the selected scales and directions form a dominant response around zero displacement within the configured workspace, the current position preferentially recalls history from the same and nearby locations. A bounded workspace makes the largest relevant distance and the smallest required resolution explicit, so the frequencies can be chosen for that range without requiring extrapolation to unseen coordinates. Finite sinusoidal features may nevertheless retain distant sidelobes. We therefore assume only that nearby locations usually produce stronger responses within the configured workspace; we do not claim that similarity decreases monotonically with Euclidean distance over all space.

3.2 Trajectory-Prior Matrix PH

The trajectory-prior matrix, referred to as the Prior Hologram (PH), records only position keys encountered in the past. It stores no observation content and does not represent walls or reachability. Let P_t denote the PH state immediately before the t-th position is queried, so it contains only previous positions. PH uses a fixed retention factor γ_{PH}, and older positions gradually weaken.

At each step, the current key first queries the historical position distribution. The current position is written only afterward for use by later steps:

$$\begin{aligned} PY_t &= P_t k_t, && \text{query history first;} \\ P_{t+1} &= \gamma_{PH} P_t + k_t k_t^\top, \quad P_1 = 0, && \text{then update PH.} \end{aligned} \tag{4}$$

Expanding the PH recurrence makes the query interpretation explicit:

$$PY_t = \sum_{s<t} \gamma_{PH}^{t-1-s}\kappa(p_s,p_t)k_s. \tag{5}$$

Every previous position p_s contributes its key k_s. Its contribution is controlled jointly by $\kappa(p_s,p_t)$, which

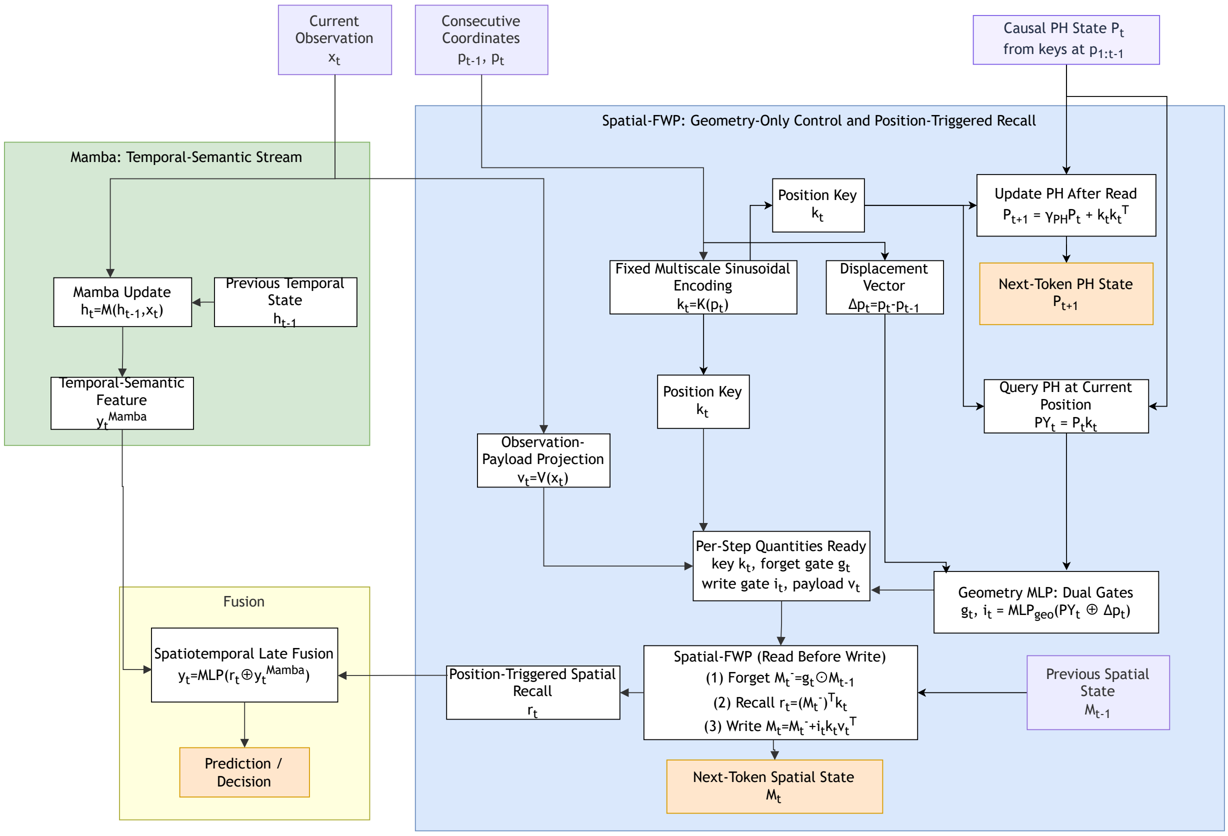

Figure 1: Overall architecture of Spatial-FWP Mamba. Mamba maintains the temporal-semantic state. The spatial branch first queries the causal trajectory prior at the current position to obtain a compressed description of nearby position history. It combines this result with the current displacement to generate geometry-only dual gates, then updates a fixed-capacity FWP by forgetting, reading, and writing in that order. The two branches interact only through late fusion at the current prediction head.

measures positional similarity to the current coordinate, and γ_{PH}^{t-1-s}, which discounts older visits. Thus, PY_t is not a global trajectory summary that is independent of the current position. It is a compressed historical query centered at p_t; querying the same P_t from another position generally produces a different result.

Under the local-similarity assumption, several historical positions near p_t tend to reinforce related frequency channels, whereas an unvisited region usually yields a weaker response or lacks the same coherent multiscale pattern. Consequently, PY_t carries cues about how historical positions are distributed around the current point, how recent those visits are, and at which spatial scales they match the current position. It is not an exact neighborhood map, a coordinate list, or a visit count, but a compact local-history description for the gating network. Because querying precedes writing, the current coordinate does not enter its own prior and no future coordinate is used. PH does not preserve directed transition order, so motion direction and step size are supplied separately by the current displacement. During training, the fixed PH recurrence can be evaluated with the prefix scan described in Section 4.

3.3 Geometry-Only Dual Gating

The spatial controller reads only PY_t and the current displacement $\Delta p_t = p_t - p_{t-1}$. It does not read x_t, the observation payload v_t, or the Mamba state h_t. The displacement is rescaled along each coordinate axis to comparable units and is denoted by δp_t. A lightweight network directly produces the forgetting and write gates:

$$(g_t, i_t) = \mathrm{MLP}_{\mathrm{geo}}\big(PY_t \oplus \delta p_t\big). \tag{6}$$

The two inputs answer different questions. The PH query PY_t describes how the neighborhood of the current position has been visited, including the strength and multiscale composition of the relevant history. The displacement δp_t describes how the current step arrived there, including motion direction and travel distance. Displacement alone cannot distinguish a jump back into a visited region from a jump into a new one, while PY_t alone does not describe the current motion. Together, they provide evidence for distinguishing continuous local scanning, cross-region jumps, returns to old areas, and entry into new areas without reading the complete coordinate history or observation semantics.

Accordingly, $PY_t \oplus \delta p_t$ supplies two complementary geometric inputs to the controller: one summarizes the spatial trace already present around the current position, and the other describes how the scan reached that position. The MLP uses these cues to produce frequency-wise retention and the overall write strength. This means that the inputs contain geometry directly relevant to the gating decision; it does not treat PY_t as a lossless map or prescribe a hand-designed gating rule.

The vector g_t is a diagonal forgetting gate that controls retention in each positional-frequency channel. The sine and cosine channels of the same frequency share one gate value, allowing that frequency to decay as a pair without disturbing its phase relationship. Different frequencies may have different retention times, producing multiscale forgetting; the mechanism cannot precisely delete one physical location.

The scalar i_t determines how strongly the current observation is written as a whole. It depends only on historical visitation and current motion, not on image quality or semantic importance. Both gates use bounded activations between zero and one. The forgetting gate additionally has an upper bound below one to prevent indefinite accumulation under additive writing.

3.4 Read-Before-Write and Position-Triggered Recall

The observation is first compressed into a value vector $v_t = V(x_t)$. It may contain a visual or semantic summary, but serves only as the stored payload and is not supplied to the spatial controller. The spatial matrix M_t is updated by forgetting, reading, and writing in that order:

$$\begin{aligned} M_t^- &= \mathrm{diag}(g_t)M_{t-1}, && \text{decay history;} \\ r_t &= (M_t^-)^\top k_t, && \text{read at the current position;} \\ M_t &= M_t^- + i_t k_t v_t^\top, && \text{write the current observation.} \end{aligned} \tag{7}$$

Here, M_t^- is the historical state visible to the current read. In implementation, g_t only scales matrix rows; the diagonal matrix need not be constructed explicitly. Because the read precedes the current write, r_t does not contain v_t.

Historical content is generally easier to retrieve when its position key is more similar to the current key. As the gates continue to act, older information weakens, and different frequencies may remain active for different durations. PH and the main FWP use the same positional similarity but query different associations: PH summarizes historical coordinates into PY_t for gate generation, whereas the main FWP retrieves coordinate-observation associations into r_t for the current prediction. All surviving writes are compressed into the fixed-size matrix M_t^-. A query therefore requires only one matrix-vector multiplication and no sliding window, neighbor list, or traversal over historical coordinates. Figure 2 illustrates the latter process: the current position automatically recalls nearby historical observations from the main FWP.

The blue intensity illustrates only the intended tendency that nearby locations usually produce stronger responses; it is not the exact response of a particular Fourier configuration. Actual recall also depends on write strength and accumulated forgetting. Both PH and the main spatial memory follow query-before-write, so the current observation cannot enter its own recall. Bounding the forgetting gate helps prevent unbounded accumulation under repeated additive writes, but does not eliminate interference in fixed-capacity memory.

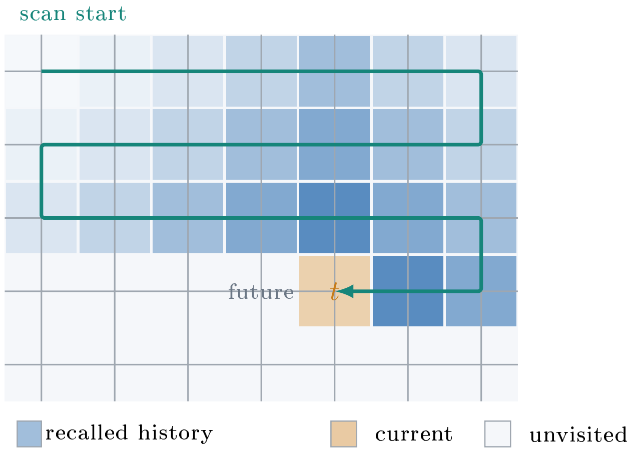

Figure 2: Automatic spatial-neighborhood recall during a two-dimensional serpentine scan. The dark trajectory is the causal prefix, and the orange cell is the current position t. Previously scanned regions are blue; darker cells are closer to the current position and are expected to produce stronger spatial responses. No explicit sliding window or traversal over historical coordinates is required.

4 Cascaded Dual Parallel Prefix Scans

Here, "parallel" does not mean removing temporal causality. During training, a tree-structured prefix scan can evaluate an entire sequence without a serial token-by-token loop. The quantities k_t, v_t, and Δp_t are first generated in parallel across time steps. The spatial chain then performs the PH scan, gate generation, and FWP scan in sequence. Mamba does not depend on PH and can run independently. During online inference, P_t and M_t are still updated one step at a time. Figure 3 summarizes this distinction.

Why prefix composition is possible. PH uses a fixed scalar retention factor, whereas the main FWP uses a diagonal forgetting vector. Both updates first scale the previous state and then add the current write, so adjacent intervals can be composed by a fixed rule. Let $B_t = i_t k_t v_t^\top$. Two consecutive main-FWP intervals combine as

$$(g_2, B_2)\circ(g_1, B_1) = (g_2\odot g_1,\ g_2[:, \mathrm{None}]\odot B_1+B_2). \tag{8}$$

The later interval first scales the write retained from the earlier interval, then adds its own write. Because g_t is a vector rather than a dense transition matrix, composition requires only elementwise scaling and addition and does not produce a dense state transition.

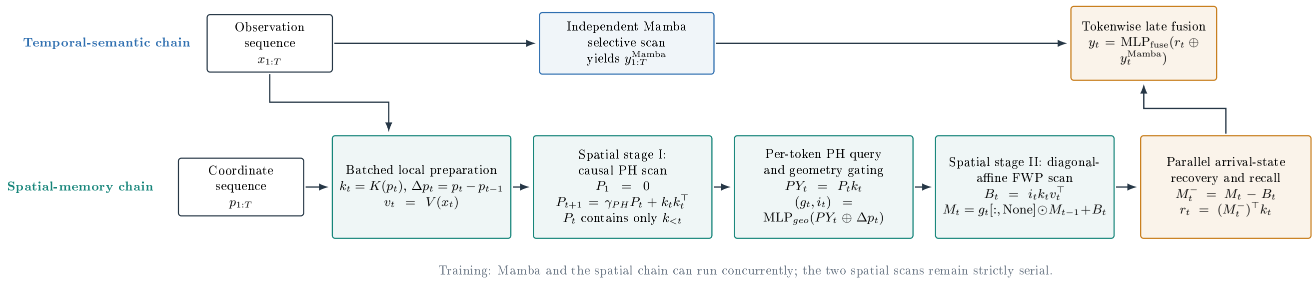

Figure 3: Dual-stream execution plan during training. Mamba runs independently, whereas the spatial chain performs the PH scan, geometry-only gate generation, and FWP scan in sequence, followed by position-triggered recall and late fusion.

The scan first produces the post-write states M_t. To preserve read-before-write semantics, the arrival state used for recall can be recovered afterward:

$$M_t^- = M_t - B_t, \qquad r_t = (M_t^-)^\top k_t. \tag{9}$$

This recovery is independent at each time step and requires no additional temporal recurrence.

Computation and state size. Online inference stores only P_t and M_t, so state size does not grow with trajectory length. The principal per-step cost of PH is $O(d_k^2)$, while FWP update and recall cost $O(d_k d_v)$. During training, total work grows linearly with sequence length, and both scans admit logarithmic parallel depth. The spatial chain must nevertheless complete the PH scan before running the FWP scan. The architecture is a fixed-capacity, lossy memory mechanism; it does not eliminate the additional computation introduced by the spatial branch.

5 Decoupled Spatiotemporal Streams and Late Fusion

The Mamba backbone follows its standard selective state-space recurrence:

$$h_t = \mathcal{M}(h_{t-1}, x_t). \tag{10}$$

Let its current-step output be

$$y_t^{\text{Mamba}} = O(h_t, x_t). \tag{11}$$

The spatial recall vector r_t is not fed into the next Mamba state, and the Mamba state is not used to generate g_t or i_t. The branches interact only through late fusion at the output:

$$y_t = \text{MLP}_{\text{fuse}}\big(r_t \oplus y_t^{\text{Mamba}}\big). \tag{12}$$

Here, "decoupled" refers to the control pathways rather than a complete absence of shared information. Both branches receive x_t, and the spatial payload v_t may contain a semantic summary. The distinction is that observation content does not determine the spatial gates, while spatial recall is not written directly into the next temporal state. Recall errors therefore do not accumulate recurrently in the Mamba state through the fusion path.

6 Experimental Design

6.1 Spatial Revisit Task

We use a single spatial-revisit task to evaluate the architecture. In a bounded continuous workspace, the model receives observations along a trajectory. A cue relevant to a later decision appears at one location; after leaving that region, the trajectory returns to the same or a nearby position, where the model must act on the earlier cue. This task places a long temporal separation and renewed spatial proximity in one process, directly testing whether coordinate-triggered recall contributes useful information to the final decision.

We retain only three comparisons: Mamba, Mamba given the same coordinate encoding, and the complete Spatial-FWP Mamba. They use the same observation encoder and comparable state sizes. Decision accuracy is the primary measure, accompanied by state size and online per-step cost. This compact comparison separates the contribution of spatial memory from the simpler effect of providing coordinates, without turning position-key diagnostics, hidden-state probes, and multiple ablations into separate experiments.

6.2 Limitations

The method has a clearly defined operating range. Sinusoidal position keys depend on a predefined bounded workspace and a corresponding frequency configuration, and we assume that scan coordinates remain inside this domain. Knowing the workspace makes it possible to configure frequencies from its maximum distance and the smallest required resolution, but it does not eliminate all distant sidelobes of a finite-dimensional Fourier kernel. If the operating range changes, coordinates must be renormalized and the frequencies reconfigured; the original encoding should not be treated as a lossless position store for unbounded space. FWP is also a fixed-capacity distributed associative matrix. As writes accumulate, different position keys and observations may overlap and interfere. Diagonal forgetting changes the retention of frequency scales but cannot delete the content of one physical location exactly.

PH records past visitation rather than obstacles, connectivity, or reachability. Two coordinates that are close in Euclidean distance but separated by a wall may still influence one another. Localization noise and accumulated drift can alter position keys and reduce revisit reliability. A geometry-only write gate also cannot detect blurred images, sensor failures, or semantically unimportant observations. This is the cost of keeping spatial control separate from content semantics. In dynamic environments, old and new observations at the same location may be compressed into the same associative direction and require forgetting or a later extension.

Computationally, PH adds a fixed state of size $O(d_k^2)$, and the main FWP scan must wait for the PH scan and gate generation. The method therefore does not have the same constant factors as a single Mamba scan. Its advantage is that state size does not grow with history length, not that spatial memory introduces no additional computation.

6.3 Future Research Directions

Multiscale sine–cosine position keys allow the spatial FWP to write and retrieve spatial history through positional bases at different frequencies. The diagonal forgetting gate used in this paper can adjust these frequency channels separately, but its control rule remains open to further study. Future work could generate a more suitable diagonal forgetting matrix from the geometry of the current environment, observation scale, and motion state, allowing each frequency scale to receive an environment-conditioned forgetting rate. This may improve on fixed or uniform forgetting when multiscale information must be retained. The useful conditioning signals and the effect of such gates on memories at other positions must be established experimentally.

Potential settings include movement among objects, variable-density image or point-cloud scanning, robot navigation, maze exploration, and generation across a two-dimensional canvas.

7 Conclusion

This paper introduced Spatial-FWP Mamba, which augments the temporal-semantic state of Mamba with a fixed-capacity coordinate-addressed memory. Within a predefined bounded workspace, multiscale position keys selectively retrieve history from the current coordinate. The PH query supplies a compressed cue about the surrounding historical position distribution, and the current displacement supplies the motion context; together they generate a paired diagonal forgetting gate and a scalar write gate. Observation content enters the spatial matrix only as a value vector; it does not control the spatial gates, and the spatial readout is not fed into the next Mamba state. Read-before-write prevents the current observation from entering its own recall. The affine forms of PH and the main FWP allow the spatial branch to use two serial associative prefix scans. The module does not replace localization, mapping, or explicit neighborhood search; it provides a fixed-capacity and necessarily lossy historical context queried by physical coordinates. Its practical value must be established through position-kernel diagnostics, controlled revisits, downstream decisions, and runtime measurements.

References

[1] Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. HiPPO: Recurrent Memory with Optimal Polynomial Projections. *Advances in Neural Information Processing Systems*, 33, 2020.

[2] Albert Gu, Karan Goel, and Christopher Ré. Efficiently Modeling Long Sequences with Structured State Spaces. *International Conference on Learning Representations*, 2022.

[3] Albert Gu and Tri Dao. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. *First Conference on Language Modeling*, 2024.

[4] Tri Dao and Albert Gu. Transformers are SSMs: Generalized Models and Efficient Algorithms through Structured State Space Duality. *Proceedings of the 41st International Conference on Machine Learning*, 2024.

[5] Jürgen Schmidhuber. Learning to Control Fast-Weight Memories: An Alternative to Dynamic Recurrent Networks. *Neural Computation*, 4(1):131–139, 1992.

[6] Jimmy Ba, Geoffrey E. Hinton, Volodymyr Mnih, Joel Z. Leibo, and Catalin Ionescu. Using Fast Weights to Attend to the Recent Past. *Advances in Neural Information Processing Systems*, 29, 2016.

[7] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention. *Proceedings of the 37th International Conference on Machine Learning*, 2020.

[8] Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. Linear Transformers Are Secretly Fast Weight Programmers. *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[9] Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated Delta Networks: Improving Mamba2 with Delta Rule. *International Conference on Learning Representations*, 2025.

[10] Tao Zhang, Haobo Yuan, Lu Qi, Jiangning Zhang, Qianyu Zhou, Shunping Ji, Shuicheng Yan, and Xiangtai Li. Point Cloud Mamba: Point Cloud Learning via State Space Model. *arXiv preprint arXiv:2403.00762*, 2024.

[11] Mark Schöne, Yash Bhisikar, Karan Bania, Khaleelulla Khan Nazeer, Christian Mayr, Anand Subramoney, and David Kappel. STREAM: A Universal State-Space Model for Sparse Geometric Data. *arXiv preprint arXiv:2411.12603*, 2024.

[12] Ali Rahimi and Benjamin Recht. Random Features for Large-Scale Kernel Machines. *Advances in Neural Information Processing Systems*, 20, 2007.

[13] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains. *Advances in Neural Information Processing Systems*, 33, 2020.

[14] Saurabh Gupta, James Davidson, Sergey Levine, Rahul Sukthankar, and Jitendra Malik. Cognitive Mapping and Planning for Visual Navigation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[15] Emilio Parisotto and Ruslan Salakhutdinov. Neural Map: Structured Memory for Deep Reinforcement Learning. *International Conference on Learning Representations*, 2018.

[16] João F. Henriques and Andrea Vedaldi. MapNet: An Allocentric Spatial Memory for Mapping Environments. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[17] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning to Explore Using Active Neural SLAM. *International Conference on Learning Representations*, 2020.