applied sciences

IMPACT FACTOR 2.5

CITESCORE 5.3

Review

A Survey on Visual Mamba

Hanwei Zhang, Ying Zhu, Dan Wang, Lijun Zhang, Tianxiang Chen, Ziyang Wang and Zi Ye

Special Issue
Application of Artificial Intelligence in Visual Processing

Edited by
Prof. Dr. Yinan Kong

MDPI

https://doi.org/10.3390/app14135683

applied sciences

MDPI

Review

A Survey on Visual Mamba

Hanwei Zhang [1,2,3], Ying Zhu [4], Dan Wang [4], Lijun Zhang [1], Tianxiang Chen [5], Ziyang Wang [6] and Zi Ye [2,*]

[1] Automotive Software Innovation Center, Chongqing 401331, China; zhang@depend.uni-saarland.de (H.Z.); zhanglj@ios.ac.cn (L.Z.)
[2] Institute of Intelligent Software, Guangzhou 511458, China
[3] Department of Computer Science, Saarland University, 66424 Homburg, Germany
[4] Hangzhou Institute for Advanced Study, University of Chinese Academy of Sciences, Hangzhou 310024, China; zhuying23@mails.ucas.ac.cn (Y.Z.); wangdan233@mails.ucas.ac.cn (D.W.)
[5] School of Cyber Space and Technology, University of Science and Technology of China, Hefei 230026, China; txchen@mail.ustc.edu.cn
[6] Department of Computer Science, University of Oxford, Oxford OX3 7LD, UK; ziyang.wang@cs.ox.ac.uk
* Correspondence: yezi1022@gmail.com

Abstract: State space models (SSM) with selection mechanisms and hardware-aware architectures, namely Mamba, have recently shown significant potential in long-sequence modeling. Since the complexity of transformers' self-attention mechanism is quadratic with image size, as well as increasing computational demands, researchers are currently exploring how to adapt Mamba for computer vision tasks. This paper is the first comprehensive survey that aims to provide an in-depth analysis of Mamba models within the domain of computer vision. It begins by exploring the foundational concepts contributing to Mamba's success, including the SSM framework, selection mechanisms, and hardware-aware design. Then, we review these vision Mamba models by categorizing them into foundational models and those enhanced with techniques including convolution, recurrence, and attention to improve their sophistication. Furthermore, we investigate the widespread applications of Mamba in vision tasks, which include their use as a backbone in various levels of vision processing. This encompasses general visual tasks, medical visual tasks (e.g., 2D/3D segmentation, classification, image registration, etc.), and remote sensing visual tasks. In particular, we introduce general visual tasks from two levels: high/mid-level vision (e.g., object detection, segmentation, video classification, etc.) and low-level vision (e.g., image super-resolution, image restoration, visual generation, etc.). We hope this endeavor will spark additional interest within the community to address current challenges and further apply Mamba models in computer vision.

Keywords: Mamba; computer vision; state space model; application

check for updates

Citation: Zhang, H.; Zhu, Y.; Wang, D.; Zhang, L.; Chen, T.; Wang, Z.; Ye, Z. A Survey on Visual Mamba. *Appl. Sci.* **2024**, *14*, 5683. https://doi.org/10.3390/app14135683

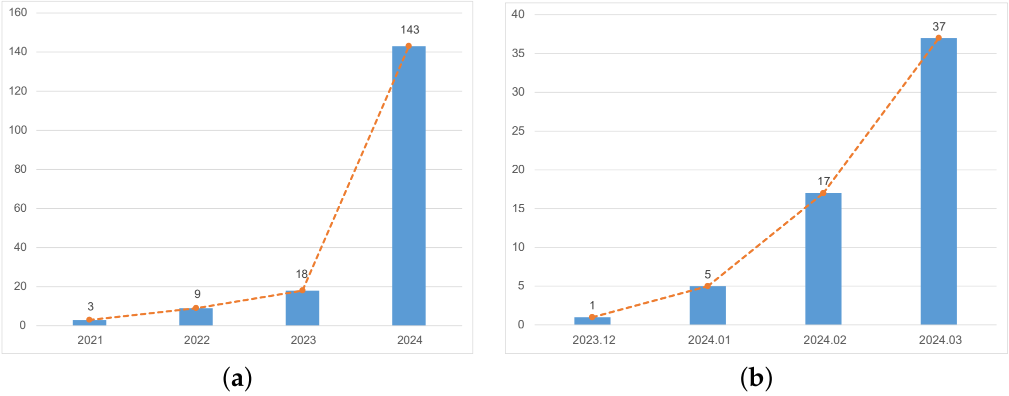
Academic Editor: Antonio Fernández-Caballero

Received: 22 May 2024
Revised: 17 June 2024
Accepted: 27 June 2024
Published: 28 June 2024

1. Introduction

Deep neural networks have exhibited remarkable performance across various artificial intelligence tasks, with the fundamental architecture playing a crucial role in determining the model's capabilities. Typically, traditional neural networks comprise multi-layer perceptron (MLP) or fully connected layers [1,2]. Convolutional neural networks (CNNs) [3,4] introduce convolutional and pooling layers, which are particularly effective for processing shift-invariant data like images. Recurrent neural networks (RNNs) [5,6] utilize recurrent cells to handle sequential or time series data. To address the existing issue of CNNs, RNNs, and Graph Neural Networks models only capturing local relationships, the transformer [7–9], introduced in 2017, excels at learning long-distance feature representations. Transformers primarily depend on attention-based attention mechanisms, e.g., self-attention and cross-attention, to extract intrinsic features and enhance their representation capability. Pre-trained massive transformer-based models, such as GPT-3 [10], deliver robust performance across various natural language processing datasets, excelling in tasks involving the generation and comprehension of natural language. The remarkable performance of transformer-based models has led to their extensive adoption in vision applications. The key to transformer models is their exceptional skill in capturing long-range dependencies, as well as maximizing the use of large datasets. The feature extraction module is the primary component of vision transformer architectures. In addition, it also processes data using a sequence of self-attention blocks, which can obviously enhance its capacity to analyze images.

Nevertheless, a primary obstacle for transformers is the substantial computational demands of the self-attention mechanism, which can increase quadratically with image resolution. The Softmax operation within the attention blocks can further intensify these computational demands, presenting significant challenges for implementing the above-mentioned models on edge and low-resource devices. Apart from that, real-time computer vision systems utilizing transformer-based models must adhere to stringent low-latency standards in order to maintain a high-quality user experience. This scenario emphasizes the continuous evolution of new architectures to enhance performance, although this usually comes with the trade-off of higher computational demands. Numerous new models on the basis of sparse attention mechanisms or innovative neural network paradigms have been put forward to further lower computational costs, while obtaining long-range dependencies and maintaining high performance. SSMs have become a central focus among these developments. As displayed in Figure 1a, the number of publications related to SSM demonstrates an explosive growth trend. Initially devised to simulate dynamic systems in areas including control theory and computational neuroscience using state variables, SSM predominantly describes linear invariant (or stable) systems adapted for deep learning.



(**a**) (**b**)

Figure 1. The number of SSM and Mamba papers published (from year 2021 to year March 2024). (**a**) SSM-based papers, (**b**) Mamba-based papers on vision.

As SSMs have evolved, a new class of selective SSMs has emerged, termed Mamba [11]. These have advanced the modeling of discrete data, such as text, with SSMs through two key improvements. Firstly, they feature an input-dependent mechanism for adjusting SSM parameters dynamically, enhancing information filtering. Secondly, Mamba employs a hardware-aware approach that processes data linearly with sequence length, boosting computational speed on modern systems. Inspired by Mamba's achievements in language modeling, several initiatives are currently aiming to adapt this success to the field of vision. Several studies have explored its integration with mixture-of-experts (MoE) techniques, as demonstrated by works like Jamba [12], MoE-Mamba [13], and BlackMamba [14], outperforming the state-of-the-art architecture transformer-MoE with fewer training steps. As depicted in Figure 1b, since the release of Mamba in December 2023, the number of research papers focusing on Mamba in the vision domain has rapidly increased, reaching a peak in March 2024. This trend suggests that Mamba is emerging as a prominent research area in vision, potentially providing a viable alternative to transformers. Therefore, a review of the current related works is necessary and timely, to provide a detailed overview of this new methodology in this evolving field.

Consequently, we present a comprehensive overview of how Mamba models are used in the vision domain. This paper aims to serve as a guide for researchers looking to delve deeper into this area. The critical contributions of our work include:

- This survey is the first attempt to offer an in-depth analysis of the Mamba technique in the vision domain, explicitly concentrating on analyzing the proposed strategies.
- An investigation on how Mamba's capabilities can be enhanced and combined with other architectures in order to achieve superior performance, by expanding upon the Naive-based Mamba visual framework.
- We offer an exploration that organizes the literature based on various application tasks. In addition, we establish a taxonomy, identify advancements specific to each task, as well as offer insights on overcoming challenges.
- To keep up with the rapid development in this field, we will regularly update this review with the latest relevant papers and develop an open-source implementation at https://github.com/ziyangwang007/Awesome-Visual-Mamba (accessed on 25 June 2024).

Here is the structure for the remaining portions of the survey. Section 2 examines the general and mathematical concepts underlying Mamba strategies. Section 3 discusses the naive Mamba visual models and how they integrate with other technologies to enhance performance, as recently proposed. Section 4 explores the application of Mamba technologies in addressing a variety of computer vision tasks. Finally, Section 5 concludes the survey.

2. Formulation of Mamba

Mamba [11] was initially introduced in the domain of natural language processing. As depicted in Figure 2, the original Mamba Block integrated a Gated MLP into the SSM architecture of H3 [15], utilizing an SSM sandwiched between two gated connections alongside a standard local convolution. For the activation function σ, SiLU [16] or Swish activation function [17] is used. The Mamba architecture consists of Mamba blocks that are repeated and interspersed with residual connections and standard normalization. An optional normalization layer (LayerNorm [18] chosen by the original Mamba) is applied in a similar location to RetNet [19].

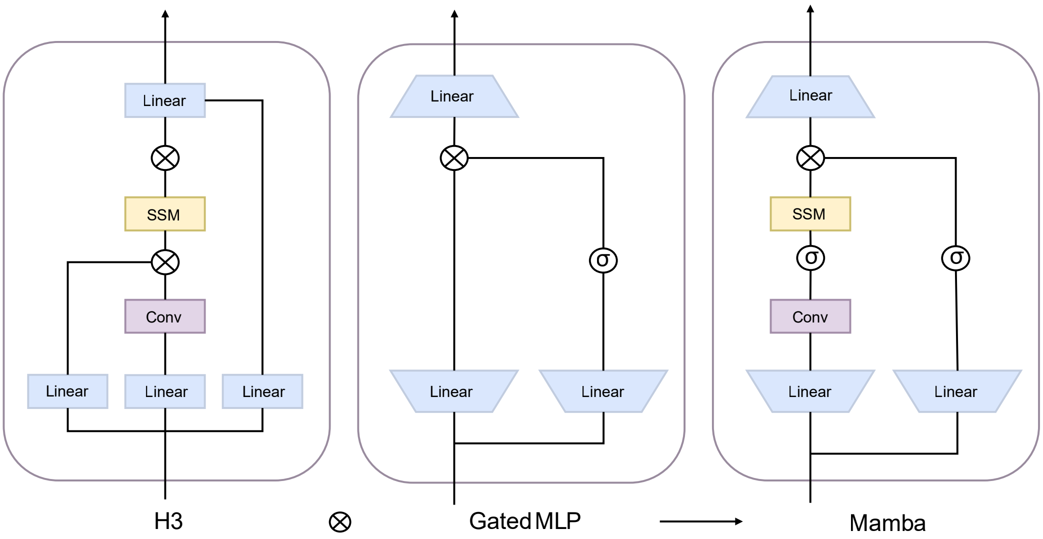


Figure 2. Graphical representation of Mamba Block [11].

2.1. State Space Model

Consider a structured SSM mapping a one-dimensional sequence $x(t) \in \mathbb{R}^L$ to $y(t) \in \mathbb{R}^L$ through a hidden state $h(t) \in \mathbb{R}^N$. With the evolution parameter $\mathbf{A} \in \mathbb{R}^{N \times N}$ and the projection parameters $\mathbf{B} \in \mathbb{R}^{N\times 1}$, $\mathbf{C} \in \mathbb{R}^{1\times N}$, such a model is formulated using linear ordinary differential equations

$$\begin{aligned} h'(t) &= \mathbf{A}h(t) + \mathbf{B}x(t), \\ y(t) &= \mathbf{C}h(t). \end{aligned} \tag{1}$$

2.1.1. Discretization

To adapt to deep learning, SSMs as continuous-time models are discretized with a zero-order hold (ZOH) assumption. Therefore, the continuous-time parameters $\mathbf{A}, \mathbf{B}$ are transformed into their discretized counterparts $\overline{\mathbf{A}}, \overline{\mathbf{B}}$ with a timescale parameter Δ according to

$$\begin{aligned} \overline{\mathbf{A}} &= \exp(\Delta\mathbf{A}), \\ \overline{\mathbf{B}} &= (\Delta\mathbf{A})^{-1}(\exp(\Delta\mathbf{A}) - \mathbf{I}) \cdot \Delta\mathbf{B}. \end{aligned} \tag{2}$$

Thus, Equation (1) can be rewritten as

$$\begin{aligned} h_t &= \overline{\mathbf{A}}h_{t-1} + \overline{\mathbf{B}}x_t, \\ y_t &= \mathbf{C}h_t. \end{aligned} \tag{3}$$

To facilitate understanding of this discretization, we have illustrated it visually in Figure 3. To enhance computational efficiency and scalability, the iterative process in Equation (3) can be synthesized through a global convolution

$$\begin{aligned} \overline{\mathbf{K}} &= (\mathbf{C}\overline{\mathbf{B}}, \mathbf{C}\overline{\mathbf{A}\mathbf{B}}, \cdots, \overline{\mathbf{A}}^{L-1}\overline{\mathbf{B}}), \\ \mathbf{y} &= \mathbf{x} * \overline{\mathbf{K}}, \end{aligned} \tag{4}$$

where L is the length of the input sequence $\mathbf{x}$, $\overline{\mathbf{K}} \in \mathbb{R}^L$ serves as the kernel of the SSM, and $*$ represents the convolution operation.

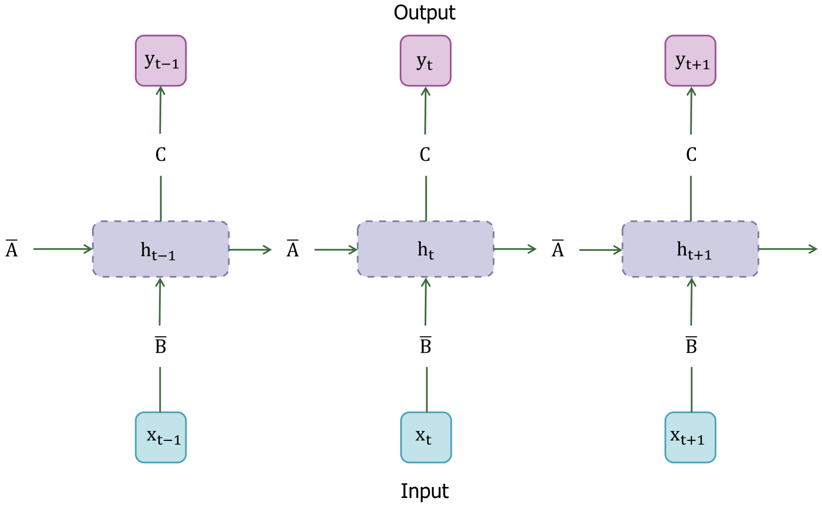


Figure 3. Graphical representation of a discretized SSM.

2.1.2. Architectures

SSMs usually serve as independent sequence transformations that can be integrated into neural network architectures that are end-to-end. Here, we introduce several fundamental architectures. Linear attention [20] approximates self-attention with a recurrence mechanism as a simplified form of linear SSM. H3 [15], as illustrated in Figure 2, places an SSM between two gated connections and inserts a standard local convolution before it.

Following H3, Hyena [21] substitutes an MLP-parameterized global convolution [22] for the SSM layer. RetNet [19] introduces an extra gate and employs simpler SSM. RetNet uses a variant of multi-head attention (MHA) in place of convolutions, providing an alternate parallelizable computing approach. Inspired by the attention-free transformer [23], the recent RNN design RWKV [24] can be interpreted as the ratio of two SSMs, owing to its primary "WKV" mechanism involving linear time invariance (LTI) recurrences.

2.1.3. Selective SSM

Traditional SSMs demonstrated linear time complexity but their representativity of sequence context is inherently limited by time-invariant parameterization. To overcome the existing constraints, selective SSMs introduce selective scan for interactions among sequential states, as shown below:

$$\begin{aligned} \mathbf{B} &= S_{\mathbf{B}}(\mathbf{x}), \\ \mathbf{C} &= S_{\mathbf{C}}(\mathbf{x}), \\ \Delta &= \tau_{\Delta}(\Delta + S_{\Delta}(\mathbf{x})). \end{aligned} \tag{5}$$

This occurs before Equations (2) and (3), so that the parameters $\mathbf{B} \in \mathbb{R}^{B \times L \times N}$, $\mathbf{C}^{B \times L \times N}$ and $\Delta^{B \times L \times D}$ are dependent on the input sequence $\mathbf{x} \in \mathbb{R}^{B \times L \times D}$, where B represents the batch size, and D represents number of channels. Normally, S_B and S_C are linear parameterized projections to dimension N, *i.e.* $Linear_N(\cdot)$, while $S_{\Delta}(\mathbf{x}) = Broadcast_D(Linear_1(\mathbf{x}))$ and $\tau_{\Delta} = softplus$. The choice of S_{Δ} and τ_{Δ} is results from the relationship with RNNs gating mechanisms, which will be explained later.

2.2. Other Key Concepts in Mamba

2.2.1. Selection Mechanism

There is a well-established link between discretizing continuous-time systems and RNN gating [25]. One example of the selection mechanism for SSMs is the traditional gating mechanism of RNNs. When $N = 1, \mathbf{A} = -1, \mathbf{B} = 1, S_{\Delta} = Linear(\mathbf{x})$ and $\tau_{\Delta} = softplus$, then the selective SSM recurrence takes the following form:

$$\begin{aligned} g_t &= \sigma(Linear(x(t))) \\ h_t &= (1 - g_t)h_{t-1} + g_t x_t. \end{aligned} \tag{6}$$

2.2.2. Scan

The selection mechanism was devised to address the constraints of linear time invariance (LTI) models. However, it reintroduces the computation issue associated with SSMs. To enhance GPU utilization and efficiently materialize the state h within the memory hierarchy, hardware-aware state expansion is enabled by selective scan. By incorporating kernel fusion and recomputation with parallel scan, the fused selective scan layer can effectively decrease the quantity of memory I/O operations, leading to a significant acceleration compared to conventional implementations.

2.2.3. Discussion

Compared to RNNs and LSTMs, which struggle with vanishing gradients and long-range dependencies, Mamba provides efficient computation and memory utilization. While transformers excel in batch processing and handling long-range dependencies through attention mechanisms, they incur high computational costs, especially during inference. Mamba introduces a selective SSM, incorporating input-dependent matrices to enhance adaptability, while maintaining the computational advantages of traditional SSMs. Mamba bridges the gap between traditional SSMs and modern neural network architectures by providing a selective dependency mechanism, optimal GPU memory utilization, and linear scalability with context length, therefore offering a promising solution for various sequential data processing tasks.

3. Mamba for Vision

The original Mamba block was designed for one-dimensional sequences, yet vision-related tasks require processing multi-dimensional inputs like images, videos, and 3D representations. Consequently, to adapt Mamba for these tasks, enhancements to the scanning mechanism and architecture of the Mamba block play a vital role in effectively handling multi-dimensional inputs.

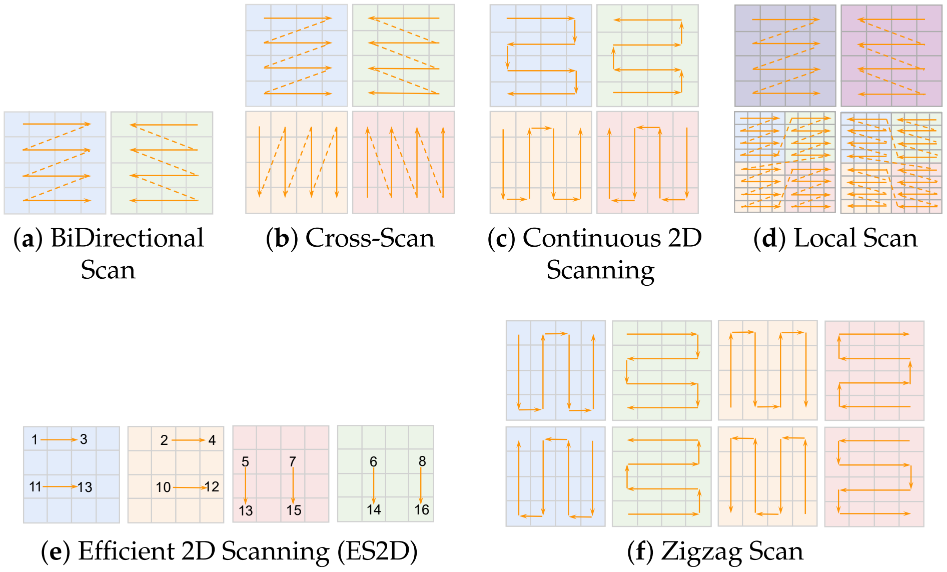
The current section presents efforts to enable Mamba to tackle vision-related tasks, while enhancing its efficiency and performance. Initially, we delve into two foundational works, including Vision Mamba [26] and VMamba [27]. These works introduced the Vision Mamba (ViM) block and visual state space (VSS) block, respectively, serving as a foundation for subsequent research endeavors. Subsequently, we explore additional works focused on refining the Mamba architecture as a backbone for vision-related tasks. Lastly, we discuss integrating Mamba with other architectures, including convolution, recurrence, and attention.

3.1. Visual Mamba Block

Drawing inspiration from the visual transformer architecture, it seems natural to preserve the framework of the transformer model, while substituting the attention block with a Mamba block and keeping the rest of the process intact. At the crux of the matter lies adapting the Mamba block to vision-related tasks. Nearly simultaneously, Vision Mamba and VMamba presented their respective solutions: the ViM block and the VSS block.

3.1.1. ViM

ViM block [26], also known as a bidirectional Mamba block, annotates image sequences with position embeddings and condenses visual representations based on a bidirectional SSM. It processes inputs both forward and backward, employing one-dimensional convolution for each direction, as displayed in Figure 4a. The Softplus function ensures non-negative Δ. Forward and backward $\mathbf{y}$ are computed via the SSM described in Equations (2) and (3), and then combined through SiLU gating to produce the output token sequence as Figure 5a.



(**a**) BiDirectional Scan

(**b**) Cross-Scan

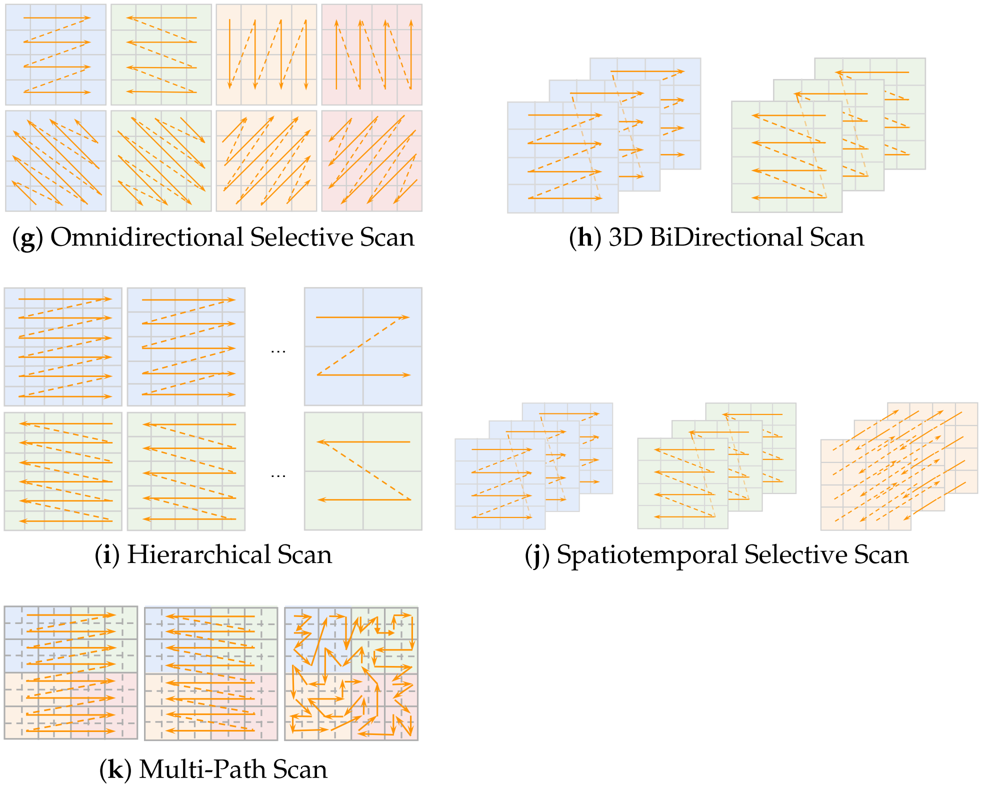
(**c**) Continuous 2D Scanning

(**d**) Local Scan

(**e**) Efficient 2D Scanning (ES2D)

(**f**) Zigzag Scan

Figure 4. *Cont.*


(**g**) Omnidirectional Selective Scan

(**h**) 3D BiDirectional Scan

(**i**) Hierarchical Scan

(**j**) Spatiotemporal Selective Scan

(**k**) Multi-Path Scan

Figure 4. Comparison between different 2D scans and the selective scan orders in Vim (**a**) [26], VMamba (**b**) [27], PlainMamba (**c**) [28], LocalMamba (**d**) [29], Efficient VMamba (**e**) [30], Zigzag (**f**) [31], VmambaIR (**g**) [32], VideoMamba (**h**) [33], Motion Mamba (**i**) [34], Vivim (**j**) [35] and RSMamba (**k**) [36].

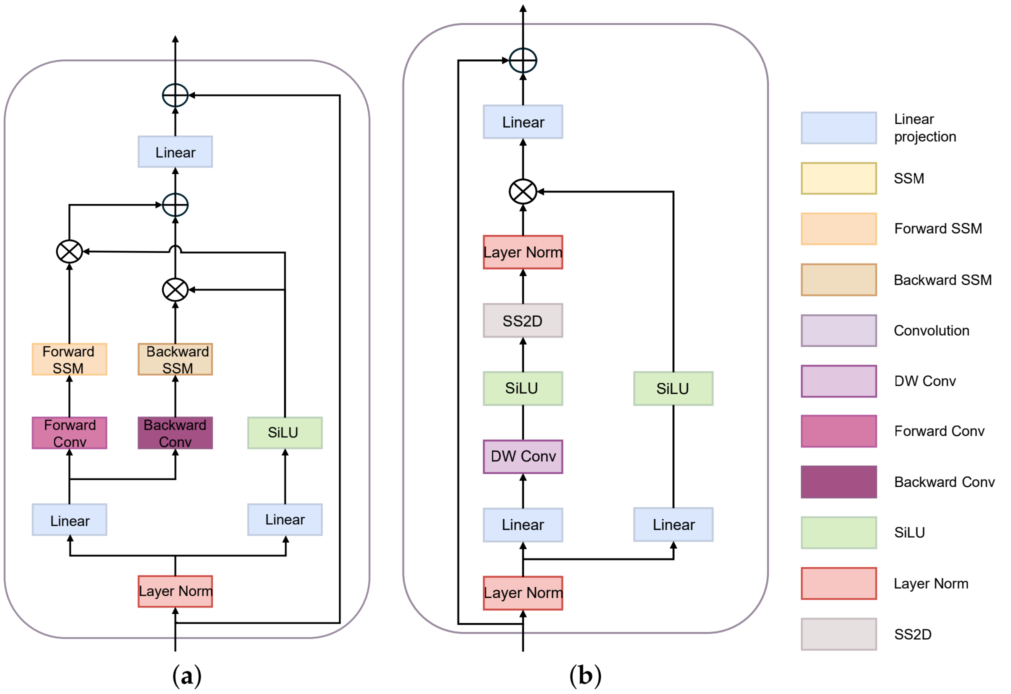


Figure 5. Graphical representation of the architecture and element functions of the ViM Block and VSS Block. (**a**) ViM block (**b**) VSS block.

3.1.2. VSS

The VSS block [27] incorporates the pivotal SSM operation. It begins by directing the input through a depth-wise convolution layer, followed by a SiLU activation function, and then through the SSM outlined in Equations (2) and (3) employing an approximate $\overline{\mathbf{B}}$. Afterward, the output of the SSM is subjected to layer normalization before being amalgamated with the output of other information streams, as in Figure 5b. To tackle the direction-sensitive problem, they presented a cross-scan module (CSM), which can traverse the spatial domain and transform any non-causal visual image into order patch sequences, as shown in (b) Figure 4. They refined the approximation of $\overline{\mathbf{B}}$ using the first-order Taylor series $\overline{\mathbf{B}} = (\Delta\mathbf{A})^{-1}(\exp(\Delta\mathbf{A}) - \mathbf{I}) \cdot \Delta\mathbf{B} \approx (\Delta\mathbf{A})(\Delta\mathbf{A})^{-1}\Delta\mathbf{B} = \Delta\mathbf{B}$.

3.2. Pure Mamba

It is clear from Figure 5 what the primary distinction is between ViM and VSS blocks: the ViM block employs separate one-dimensional convolutions for scanning different directions. Conversely, in VSS blocks, different scanning schemes share the same depth-wise convolution layer. Thus, for pure Mamba architectures, we consider designs that use individual one-dimensional convolutions for each scanning direction as an extension of ViM-based approaches, while those sharing a depth-wise convolution layer among different scanning schemes are seen as extensions of VSS-based approaches. In addition to these two approaches, researchers also consider visual data as multi-dimensional data, where the pure Mamba architecture typically relies on the original Mamba block. Therefore, in this subsection, we will introduce the pure Mamba architecture derived from these three branches, followed by a summary of 2D scanning mechanisms utilized in visual Mamba.

3.2.1. ViM-Based

Inspired by the vision transformer architecture, Vision Mamba [26] replaces the transformer encoder with a vision Mamba encoder based on ViM blocks, while retaining the remainder of the process. This involves converting the two-dimensional image into flattened patches, followed by linear projection of these patches into vectors and the addition of position embeddings. A class token represents the entire patch sequence, and subsequent steps involve normalization layers and an MLP layer to derive the final predictions.

LocalMamba [29] is built based on a Vim block, and it introduces a revolutionary approach to scanning that combines global context with localized scanning within distinct windows to capture comprehensive local information. In addition, LocalMamba searches scanning directions across various network layers to identify and utilize the most effective scanning combinations. They proposed two variants, i.e., with plain and hierarchical structures. In addition, they proposed their LocalVim Block, which includes four scanning directions (*cf*. Figure 4d) shows ViM scanning and partitioning tokens into distinct windows, in addition to their flipped equivalents, to facilitate scanning from tail to head. Additionally, the block incorporates a state space module and a spatial and channel attention module (SCAttn).

3.2.2. VSS-Based

VMamba [27] undergoes four stages after partitioning the input image into patches as Vision Mamba. VMamba stacks several VSS blocks on the feature map with a resolution $\frac{H}{4} \times \frac{W}{4}$ as Stage 1. In Stage 2, before more VSS blocks are involved, the feature map in Stage 1 undergoes a patch merge operation for downsampling, in order to build hierarchical representations, resulting in an output resolution of $\frac{H}{8} \times \frac{W}{8}$. Stage 3 and Stage 4 are the repetition of Stage 1 and Stage 2 with resolutions of $\frac{H}{16} \times \frac{W}{16}$ and $\frac{H}{32} \times \frac{W}{32}$.

The PlainMamba block [28], which is based on the VSS block, uses the following two primary mechanisms to improve its capacity to learn features from two-dimensional images: (i) using a continuous 2D scanning procedure to increase spatial continuity, ensuring tokens in the scanning sequence are adjacent, as depicted in Figure 4c; and (ii) integrating direction-aware updating to encode directional information, allowing the model to recognize spatial relationships between tokens. PlainMamba can address the issue of spatial discontinuity in the 2D scanning mechanisms of Vim and VMamba. It continues to scan in the opposite direction as it moves to a new row or column until it reaches the image's final vision token. Moreover, PlainMamba does away with the requirement for special tokens.

Within lightweight model designs, EfficientVMamba [30] improves the capabilities of VMamba with an atrous-based selective scan approach, i.e., efficient 2D scanning (ES2D). Instead of scanning all patches from various directions and increasing the total number of patches, ES2D adopts a strategy of scanning forward vertically and horizontally, while skipping patches and maintaining the number of patches unchanged, as shown in Figure 4e. Their efficient visual state space (EVSS) block comprises a convolutional branch for local features, applies ES2D as the SSM branch for global features, and all branches end with a squeeze-excitation block. They employ EVSS blocks for both Stage 1 and Stage 2, while opting for inverted residual blocks in Stage 3 and Stage 4, to enhance the capture of global representations.

3.2.3. Visual Data as Multi-Dimensional Data

Existing models for multi-dimensional data also work for visual-related tasks but often lack the capacity to facilitate inter- and intra-dimensional communication or data independence. The MambaMixer block [37] incorporates a dual selection mechanism that operates across tokens and channels. By linking sequential selective mixers via a weighted averaging mechanism, it enables layers to directly access inputs and outputs from different layers. Mamba-ND [38] expands the application of the SSM to higher dimensions by alternating sequence wandering across layers. Utilizing a similar scanning strategy as VMamba in the 2D scenario, it extends this approach to 3D. Additionally, they advocated for the use of multi-head SSMs as an analog to multi-head attention. In response to the inefficiencies and performance challenges encountered by traditional transformers in image and time series processing, a new architecture named simplified Mamba-based architecture, SiMBA [39] was proposed to incorporate the Mamba block for sequence modeling and Einstein FFT (EinFFT) for channel modeling, hoping to improve the model's stability and effectiveness when handling image and time series tasks. The Mamba block proves effective at processing long sequence data, while EinFFT represents a novel channel modeling technique. Experimental results demonstrated that SiMBA surpassed the existing SSMs and transformers across multiple benchmark tests.

3.2.4. Summary of 2D Scanning Mechanisms

Scan serves as a key component for Mamba, as when it comes to multi-dimensional inputs, the scanning mechanism matter. As shown in Figure 4, we summarize the existing 2D scanning mechanisms. In particular, direction-aware updating employs a set of learnable parameters $\{\Theta_k\}$ to represent both the four cardinal directions and a special begin direction for the initial token, reformulating Equation (3) as follows:

$$\begin{aligned} h'_k(t) &= \overline{\mathbf{A}}_t h_k(t) + (\overline{\mathbf{B}}_t + \overline{\Theta}_{k,t}) x(t), \\ y'(t) &= \sum_{k=1}^{4} \mathbf{C}_t h'_k(t), \\ y(t) &= y'(t) \odot z(t), \end{aligned} \tag{7}$$

where $\odot$ denotes the element-wise multiplication, and $z(t)$ is a gating mechanism that modulates the output. Expanding on the fundamental structure of Mamba in Equation (7), we can devise the additional scanning mechanisms depicted in Figure 4.

As a vital element of Mamba, scanning mechanisms not only help the efficiency but also provide information in the scenario of visual-related tasks. In this study, we summarize the usage of different scanning mechanisms in existing works in Table 1. Cross-scan [27] and bidirectional scan [26] stand out as the most widely adopted scanning mechanisms. Nevertheless, various other scanning mechanisms serve specific purposes. For example, 3D bidirectional scan [33] and spatiotemporal selective scan [35] are tailored for video inputs. Local scan [29] focuses on gathering local information, while ES2D [30] prioritizes efficiency.

3.3. Mamba with Other Architectures

Mamba, being a novel component compared to convolution, recurrence, and attention, offers opportunities for synergistic combinations with other architectures that are still relatively underexplored. The combination of Mamba with other architectures typically occurs through two approaches: (i) designing blocks that integrate the concepts of other architectures alongside Mamba blocks, or (ii) incorporating Mamba blocks into existing architectures. In this section, we examine existing exploratory findings on such combinations. We begin by presenting the first type of combination, followed by summarizing the second type in Table 2.

Table 1. Summary of the Scanning mechanisms used in visual Mamba.

Scanning Mechanisms	Method
BiDirectional Scan [26]	Vision Mamba [26], Motion Mamba [34] HARMamba [40], MMA [41], VL-Mamba[42] Video Mamba Suite [43], Point Mamba [44] LMa-UNet [45] Motion-Guided Dual-Camera Tracker [46]
Cross-Scan [27]	VMamba [27],VL-Mamba[42], VMRNN [47] RES-VMAMBA [48], Sigma [49], ReMamber [50] Mamba-UNet [51], Semi-Mamba-UNet [52] VMambaMorph [53], ChangeMamba [54] H-vmunet [55], MambaMIR [56], MambaIR [57] Serpent [58], Mamba-HUNet [59], TM-UNet [60] Swin-UMamba [61], UltraLight VM-UNet [62] VM-UNet [63], VM-UNET-V2 [64] MedMamba [65], MIM-ISTD [66], RS3Mamba [67]
Continuous 2D Scanning [28]	PlainMamba [28]
Local Scan [29]	LocalMamba [29], FreqMamba [68]
Efficient 2D Scanning (ES2D) [30]	EfficientVMamba [30]
Zigzag Scan [31]	ZigMa [31]
Omnidirectional Selective Scan [32]	VmambaIR [32], RS-Mamba [69]
3D BiDirectional Scan [33]	VideoMamba [33]
Hierarchical Scan [34]	Motion Mamba [34]
Spatiotemporal Selective Scan [35]	Vivim [35]
Multi-Path Scan [36]	RSMamba [36]

3.3.1. Mamba with Convolution

Convolution, being a widely employed technique, possesses the advantageous property of capturing local information. Consequently, it is frequently integrated with Mamba to augment its capabilities. To construct a new fundamental block, the identity branch of the residual block and convolution layers are commonly employed to enhance the Mamba block. This integration is aimed at enhancing the representational capability of the model and its effectiveness in tasks requiring a thorough comprehension of visual data, by combining localized details captured by Mamba blocks with overarching global features. By incorporating a residual learning framework into the VMamba model, RES-VMAMBA [48] was the first to utilize both local and global state features that were part of the original VMamba architectural design. MambaIR [57] introduced the residue state space (RSS) block, incorporating scale residual connections, a convolutionary layer, and channel attention atop the VSS block. LMa-UNet [45] incorporates residual connections with ViM at both pixel-level and patch-level. nnMamba [70] introduced the Res-Mamba block, which merges the Mamba block with a convolution layer, batch normalization, ReLU activation, residual connections, and weight sharing among channels and spatial dimensions using a Siamese input. SegMamba [71] introduced the TSMamba block, which can enhance the Tri-orientated Mamba with layer normalization, gated spatial convolutional layers, and residual connections. MambaMIR [56] introduced the AMSS block group, which enhances feature extraction for reconstruction and uncertainty estimation by incorporating a convolutional layer and layer normalization ahead of a VSS-based Mamba block called AMSS. MedMamba [65] introduced the SS-Conv-SSM block, which comprises a convolution branch and a VSS branch.

Table 2. Summary of visual Mamba with other architectures.

Other Architecture	Mamba Method	Capability
Convolution	RES-VMAMBA [48]	Food vision tasks
	MedMamba [65]	Medical images classification tasks
	HSIMamba [72]	Hyperspectral images classification tasks
	MambaMIR [56] MambaMIR-GAN [56]	Medical images reconstruction tasks
	MambaIR [57]	Image restoration tasks
	VMambaMorph [53]	3D images registration tasks
	FreqMamba [68]	Image deraining tasks
	Pan-Mamba [73]	Pan-sharpening tasks
	MambaTalk [74]	Gesture synthesis tasks
	Samba [75]	Images semantic segmentation tasks
	Semi-Mamba-UNet [52], Swin-UMamba [61] H-vmunet [55], UltraLight VM-UNet [62] Weak-Mamba-UNet [76] LMa-UNet [45], SegMamba [71], T-Mamba [77] Vivim [35], nnMamba [70], ProMamba [78]	Medical images segmentation tasks
Recurrence	VMRNN [47]	Video prediction tasks
	VMambaMorph [53]	3D images registration tasks
Attention	SSM-ViT [79]	Event camera-based tasks
	MMA [41]	Image super-resolution tasks
	ViS4mer [80]	Long movie clip classification tasks
	FDVM-Net [81]	Images exposure correction tasks
	CMViM [82]	3D multi-modal representation tasks
	Motion-Guided Dual-Camera Tracker [46]	Endoscopy skill evaluation tasks
	MambaIR [57]	Image restoration tasks
	FreqMamba [68]	Image deraining tasks
	3DMambaComplete [83]	Point cloud completion tasks
	VM-UNET-V2 [64], Weak-Mamba-UNet [76] UltraLight VM-UNet [62], ProMamba [78]	Medical images segmentation tasks
U-Net	U-Mamba [84], UVM-Net [85], Mamba-UNet [51] TM-UNet [60], Semi-Mamba-UNet [52] Swin-UMamba [61], Weak-Mamba-UNet [76] LMa-UNet [45], LightM-UNet [86] UltraLight VM-UNet [62], VM-UNET-V2 [64] H-vmunet [55], Mamba-HUNet [59] VM-UNet [63]	Medical images tasks
	MambaMIR-GAN [56]	Medical images reconstruction tasks
	VmambaIR [32]	Image restoration tasks
	Motion Mamba [34]	Generation tasks
	MambaMorph [87]	Multi-modality registration tasks
	FreqMamba [68]	Image deraining tasks
	RS-Mamba [69]	Dense image prediction tasks
Diffusion	DiS [88], ZigMa [31], Motion Mamba [34] SSM-based diffusion model [89]	Generation tasks
	MD-Dose [90]	Radiation dose prediction tasks

In particular, specialized Mamba-based blocks have been devised to capture frequency-based information, leveraging techniques from convolutional networks. T-Mamba [77] introduced the Tim block, which integrates frequency-based bandpass filtering atop convolutional shared dual position encoding compensation and a gate selection unit. HSI-Mamba [72] introduced the HyperspectralBiNetworks block, which was derived from ViM but tailored for spectral inputs. Vivim [35] introduced the Temporal Mamba Block, integrating a spatiotemporal version of ViM named ST-Mamba, along with detail-specific FFN, convolution, and layer normalization. FreqMamba [68] introduced the FreqSSM block, incorporating a convolutional layer for the spatial branch, discrete wavelet transformation with the SSM block for the frequency band branch, and a Fourier modeling branch implemented with a convolutional layer.

A branch of research directly integrates Mamba blocks with convolutional layers into existing architectures. VMambaMorph [53] utilizes a hybrid approach, employing half of a 3D VSS block and half of a 3D CNN to construct a U-shaped network to serve as the registration module. Pan-Mamba [73] and UltraLight VM-UNet [62] incorporate convolutional layers at the start and end of their architecture to enhance feature quality. RS3Mamba [67] employs VSS blocks to construct an auxiliary encoder and utilizes convolution-based CCM modules and residual blocks as the main encoder for semantic segmentation of remote sensing images. Samba [36] utilizes convolution as a stem at the beginning of the architecture. H-vmunet [55] integrates high-order VSS blocks into a CNN-based U-shaped network for medical image segmentation tasks. Swin-UMamba [61] substitutes attention with VSS blocks in a Swin model [91], incorporating hierarchical Mamba with shifted windows. Additionally, it integrates a CNN-based U-shaped network architecture. Semi-Mamba-UNet [52] combines Mamba-based and CNN-based U-shaped branches to segment medical images in a semi-supervised manner. Furthermore, Weak-Mamba-UNet [76] enhances performance by incorporating an additional ViT-based U-shaped branch. MambaTalk [74] employs two convolutional-based audio feature extraction networks along with a Mamba model for motion synthesis. ProMamba [78] consists of a ViM-based image encoder, a Transformer-based prompt encoder, and a CNN-based mask decoder.

3.3.2. Mamba with Recurrence

To harness the long-sequence modeling capabilities of Mamba blocks and the spatiotemporal representation prowess of LSTMs, the VMRNN [47] Cell eliminates all weights and biases in ConvLSTM [92] and employs VSS blocks to learn spatial dependencies vertically. Long-term and short-term temporal dependencies are captured in the VMRNN Cell by updating the information on cell states and concealed states from a horizontal perspective. Building upon the VMRNN Cell, two variants have been proposed: VMRNN-B and VMRNN-D. VMRNN-B mainly focuses on stacking VMRNN layers, while VMRNN-D incorporates more VMRNN Cells and introduces patch merging and patch expanding layers. By downsampling the data and lowering its spatial dimensions, the patch merging layer helps to capture more abstract, global features, while also lowering the computational complexity. In contrast, upsampling employs a patch-expanding layer to increase the spatial dimensions in order to recover detail and facilitate accurate feature localization during the reconstruction stage. Ultimately, the reconstruction layer creates the predicted frame for the subsequent time step by scaling the concealed state from the VMRNN layer back to the input size. Integrating downsampling and upsampling processes offers important benefits for a predictive architecture. By making the input representation simpler through downsampling, the model can process higher-level features with minimal computational cost. This is especially helpful for grasping the intricate linkages and patterns in the data more abstractly. In addition, in VMambaMorph [53], a recursive registration framework integrated a hybrid VSS and CNNs-based VMambaMorph as the registration module.

3.3.3. Mamba with Attention

Attention mechanisms, like self-attention and cross-attention, empower transformers to concentrate on pertinent segments of the input sequence. This attention-driven strategy boosts the model's capacity to assess the significance of various elements, resulting in more intricate and contextually informed representations. Cross-attention stands out as the most prevalent element for integrating with Mamba to facilitate information exchange. FD-Vision Mamba [81] introduced the C-SSM block, which merges the SSM block with cross-attention to facilitate information exchange between the amplitude and phase branches. SpikeMba [93] integrated the SNN block with simplified cross-attention in a spiking saliency detector to enable information exchange between text features and relevant slots. Subsequently, it employs multi-modal relevant Mamba blocks to bolster long-range dependency. MambaIR [57] uses channel attention as part of its RSS block. The meet more areas (MMA) block introduced in [41] adopts a MetaFormer-style architecture, comprising two layer normalization layers, a token mixer (consisting of a channel attention mechanism and a ViM block in parallel), and an MLP block for deep feature extraction. Instead of using cross-attention, the SSM-ViT block [79] comprises three main components: a self-attention block (Block-SA), a dilated attention block (Grid-SA), and an SSM block. The block-SA focuses on immediate spatial relations and provides a detailed representation of nearby features. Grid-SA offers a global perspective, capturing comprehensive spatial relations and overall input structure. The SSM block ensures temporal consistency and a smooth information transition between consecutive time steps. By integrating SSMs with self-attention, the SSM-ViT block enables faster training and parameter timescale adjustment for temporal aggregation.

Few works have leveraged attention at the architectural level to enhance Mamba's performance. In ViS4mer [80], self-attention is employed to process each frame and obtain features for the SSN-based multi-scale decoder. Following the extraction of the short-range spatiotemporal features by the normal transformer encoder, the long-term temporal reasoning is captured by the Mamba-based multi-scale temporal S4 decoder. Thus, ViS4mer achieves decent performance in understanding long videos. CMViM [82] incorporates a single cross-attention layer after the online ViM encoder to facilitate information exchange between MRI and PET branches. 3DMambaComplete [83] integrates attention blocks into the HyperPoint generation process to enhance features extracted from incomplete point clouds and FPS. FreqMamba [68] exploits the distinctive data-dependent characteristic of Mamba alongside attention to identify potential degradation locations at different granular levels. ProMamba [78] employs self-attention and prompt-to-image attention mechanisms within the prompt encoder. UltraLight VM-UNet [62] employs spatial and channel attention mechanisms to facilitate weight sharing. Conversely, VMUNetV2 [64] introduces an SDI block, which computes attention scores for both spatial and channel dimensions. This block is positioned between the encoder, composed of VSS blocks, and the decoder, consisting of fusion blocks. In the motion-guided dual-camera tracker [46], two crucial elements are employed: a cross-camera mutual template strategy (CMT) and a Mamba-based motion-guided prediction head (MMH). Inspired by cross-attention, CMT aggregates features from dual cameras, while MMH utilizes a ViM block to capture motion tokens. The integration of vision and motion is facilitated by a cross-attention module.

3.3.4. Others

The U-shape net and diffusion architectures serve as fundamental frameworks frequently combined with Mamba blocks, particularly in the medical field. Given their prevalence, we believe that it is important to highlight them, so we have compiled related works in Table 2.

3.4. Comparison of Mamba Models and Other State-of-the-Art Models

In this section, we have thoroughly summarized the performance of various visual Mamba backbone networks on standard benchmarks and conducted an in-depth comparison with the performance of some backbones from CNN and Transformer models on the same datasets, as shown in Tables 3–6. Our analysis focused on three public datasets: ImageNet-1K for classification, COCO for object detection, ADE10K for semantic segmentation. By analyzing the performance and computational complexity of models with different architectures, we can gain insights into each model's advantages, guiding model selection and providing valuable references for future research and practical applications in computer vision.

According to these tables, we clearly observe that compared to CNN and Transformer models, Mamba either achieved better performance or required less computational resources. In Table 3, the best performance achieved by CNN was 81.7% with 39M parameters, 84.5% with 88M parameters by transformer, while Mamba achieved 84.7% with 40M parameters. Similar superior performances by Mamba are observed in Tables 4 and 6. An exception is noted in Table 5, where a Transformer achieved the best performance of 51.9% with 145M parameters and 982 FLOPs, whereas Mamba, with a maximum of 69M parameters, achieved an 49.9% average precision (AP).

3.4.1. Analysis and Comparison in Image Classification Tasks

As shown in Table 3, the ImageNet-1K dataset is designed for image classification tasks, where accuracy is the key measure of performance. Therefore, we used the Top-1 Accuracy metric to compare the classification capabilities of different models. The comparison shows that CNN models have moderate parameters and computational complexity but relatively low Top-1 accuracy, mostly below 80%. The highest Top-1 accuracy among CNN models was 81.7% for RegNetY-8G [94]. In contrast, most Transformer models have more parameters and greater computational complexity. For example, the ViT-L/16 [95] model has 307M parameters and 190.7G FLOPs, making it suitable for high-performance computing environments. In such environments, Transformer models generally achieve higher Top-1 accuracy, such as 83.7% for ViL-Base-RPB [96] and 83.8% for Focal-Base [97]. Mamba models exhibit various parameters and computational complexities, accommodating various application needs. They include lightweight models like EfficientVMamba-T [30] and more complex models like VMamba-B [27]. Many Mamba models achieved Top-1 accuracy above 83%, such as LocalVMamba-S [29] and SiMBA-B [39].

From Table 3, it is evident that CNN models do not perform as well in terms of accuracy compared to Transformer and Mamba models. Let us focus on comparing the Transformer and Mamba models. Among the Transformer models, Swin-B [91] performed the best, with a Top-1 accuracy of 84.5%. Of the Mamba models, SiMBA-B (MLP) [39] performed the best, with a Top-1 accuracy of 84.7%. Notably, Swin-B used an image size of 384^2, whereas SiMBA-B (MLP) used an image size of 224^2. Swin-B can provide more detailed information, but SiMBA-B (MLP) still outperformed Swin-B. Additionally, because Swin-B uses a larger image size, its parameters and computational complexity are significantly higher than SiMBA-B's (MLP).

Based on the above analysis, we can summarize the advantages and disadvantages of these three types of models in image classification tasks. CNN models have relatively low parameters and computational complexity, making them suitable for environments with limited resources, but perform worse for accuracy. Transformer models have relatively high accuracy, higher parameters, and computational complexity, requiring substantial resources, thus fitting high-performance computing environments. Mamba models are more diverse, fitting different computational resources and application requirements, with many models being competitive in accuracy. However, researchers must select the appropriate model based on specific application scenarios.

3.4.2. Analysis and Comparison in Object Detection and Instance Segmentation Tasks

Since the COCO dataset is used for object detection and instance segmentation tasks, it not only requires recognizing object categories but also accurately localizing them. Therefore, average precision (AP) metrics are needed to comprehensively evaluate a models' performance at different intersection over union (IoU) thresholds. To fully measure the performance of models under different training configurations, we have provided two tables, which show the performance of Mask R-CNN under the a 1× schedule and 3 × *MS* schedule for object detection and instance segmentation tasks on the COCO dataset, as shown in Tables 4 and 5.

Table 3. Comparison of Mamba models and other state-of-the-art models on t ImageNet-1K dataset.

Model	Backbone	Image Size	Params (M)	FLOPs (G)	Top-1 ACC (%)
CNN	ResNet-50 [98]	224^2	25.5	4.1	76.50
	ResNet-50-D [99]	224^2	25.0	4.3	77.16
	ResNet-101 [98]	224^2	44.6	7.8	77.4
	ResNet-152 [98]	224^2	60.2	11.6	78.3
	ResNeXt-50-32 × 4d [100]	224^2	25	4.1	77.8
	ResNeXt-101-32 × 4d [100]	224^2	44	7.8	78.8
	RegNetY-4G [94]	224^2	21	4.0	80.0
	RegNetY-8G [94]	224^2	39	8.0	81.7
Transformer	ViT-B/16 [95]	384^2	86	55.4	77.9
	ViT-L/16 [95]	384^2	307	190.7	76.5
	DeiT-S [101]	224^2	22	4.6	79.8
	DeiT-B [101]	224^2	86	17.6	81.8
	DeiT-B [101]	384^2	86	55.4	83.1
	Swin-T [91]	224^2	29	4.5	81.3
	Swin-S [91]	224^2	50	8.7	83.0
	Swin-B [91]	224^2	88	15.4	83.5
	Swin-B [91]	384^2	88	47.0	84.5
	ViL-Small-APE [96]	224^2	24.6	4.9	82.0
	ViL-Small-RPB [96]	224^2	24.6	4.9	82.4
	ViL-Medium-APE [96]	224^2	39.7	8.7	83.3
	ViL-Medium-RPB [96]	224^2	39.7	8.7	83.5
	ViL-Base-APE [96]	224^2	55.7	13.4	83.2
	ViL-Base-RPB [96]	224^2	55.7	13.4	83.7
	Focal-Tiny [97]	224^2	29.1	4.9	82.2
	Focal-Small [97]	224^2	51.1	9.1	83.5
	Focal-Base [97]	224^2	89.8	16.0	83.8
Mamba	Vim-Ti [26]	224^2	7	-	76.1
	Vim-S [26]	224^2	26	-	80.5
	VMamba-T [27]	224^2	22	4.5	82.2
	VMamba-S [27]	224^2	44	9.1	83.5
	VMamba-B [27]	224^2	75	15.2	83.2
	PlainMamba-L1 [28]	224^2	7	3.0	77.9
	PlainMamba-L2 [28]	224^2	25	8.1	81.6
	PlainMamba-L3 [28]	224^2	50	14.4	82.3
	LocalVim-T [29]	224^2	8	1.5	76.2
	LocalVim-S [29]	224^2	28	4.8	81.2
	LocalVMamba-T [29]	224^2	26	5.7	82.7
	LocalVMamba-S [29]	224^2	50	11.4	83.7
	EfficientVMamba-T [30]	224^2	6	0.8	76.5
	EfficientVMamba-S [30]	224^2	11	1.3	78.7
	EfficientVMamba-B [30]	224^2	33	4.0	81.8
	Mamba-2D-S [38]	224^2	24	-	81.7
	Mamba-2D-B [38]	224^2	92	-	83.0
	SiMBA-S (Monarch) [39]	224^2	18.5	3.6	81.1
	SiMBA-S (EinFFT) [39]	224^2	15.3	2.4	81.7
	SiMBA-S (MLP) [39]	224^2	26.5	5.0	84.0
	SiMBA-B (Monarch) [39]	224^2	26.9	5.5	82.6
	SiMBA-B (EinFFT) [39]	224^2	22.8	4.2	83.0
	SiMBA-B (MLP) [39]	224^2	40.0	9.0	84.7
	SiMBA-L (Monarch) [39]	224^2	42	8.7	83.8
	SiMBA-L (EinFFT) [39]	224^2	36.6	7.6	83.9

As seen in Table 4, when using the Mask R-CNN 1× schedule, the precision metrics of CNN models improved as the number of parameters and FLOPs increased. Compared to Transformer and Mamba models, CNN models had moderate parameters and FLOPs but relatively lower performance. Among the Transformer models, the ViT-Adapter-B [102] performed excellently but with relatively high parameters. Transformer models offer good performance but generally have higher parameters and FLOPs. On the other hand, Mamba models, while maintaining lower parameters and computational complexity, can deliver performance very close to or even surpassing some Transformer models, particularly with the LocalVMamba [29] and VMamba [27] backbones.

Table 4. Comparison of Mamba Models and other state-of-the-art models on COCO dataset (Mask R-CNN 1× schedule).

Model	Backbone	$\mathbf{AP}^b$	$\mathbf{AP}^b_{50}$	$\mathbf{AP}^b_{75}$	$\mathbf{AP}^m$	$\mathbf{AP}^m_{50}$	$\mathbf{AP}^m_{75}$	Params (M)	FLOPs (G)
CNN	ResNet-50 [98]	38.2	58.8	41.4	34.7	55.7	37.2	44	260
	ResNet-101 [98]	38.2	58.8	41.4	34.7	55.7	37.2	63	336
	ResNeXt101-32 × 4d [100]	41.9	-	-	37.5	-	-	63	340
	ResNeXt101-64 × 4d [100]	42.8	-	-	38.4	-	-	102	493
Transformer	ViT-Adapter-T [102]	41.1	62.5	44.3	37.5	59.7	39.9	28.1	-
	ViT-Adapter-S [102]	44.7	65.8	48.3	39.9	62.5	42.8	47.8	-
	ViT-Adapter-B [102]	47.0	68.2	51.4	41.8	65.1	44.9	120.2	-
	Swin-Tiny [91]	42.2	-	-	39.1	-	-	48	264
	Swin-Small [91]	44.8	-	-	40.9	-	-	69	354
	PVT-Tiny [103]	36.7	59.2	39.3	35.1	56.7	37.3	32.9	-
	PVT-Small [103]	40.4	62.9	43.8	37.8	60.1	40.3	44.1	-
	PVT-Medium [103]	42.0	64.4	45.6	39.0	61.6	42.1	63.9	-
	PVT-Large [103]	42.9	65.0	46.6	39.5	61.9	42.5	81.0	-
Mamba	VMamba-T [27]	46.5	68.5	50.7	42.1	65.5	45.3	42	262
	VMamba-S [27]	48.2	69.7	52.5	43.0	66.6	46.4	64	357
	VMamba-B [27]	48.5	69.6	53.0	43.1	67.0	46.4	96	482
	PlainMamba-Adapter-L1 [28]	44.1	64.8	47.9	39.1	61.6	41.9	31	388
	PlainMamba-Adapter-L2 [28]	46.0	66.9	50.1	40.6	63.8	43.6	53	542
	PlainMamba-Adapter-L3 [28]	46.8	68.0	51.1	41.2	64.7	43.9	79	696
	EfficientVMamba-T [30]	35.6	57.7	38.0	33.2	54.4	35.1	11	60
	EfficientVMamba-S [30]	39.3	61.8	42.6	36.7	58.9	39.2	31	197
	EfficientVMamba-B [30]	43.7	66.2	47.9	40.2	63.3	42.9	53	252
	LocalVMamba-T [29]	46.7	68.7	50.8	42.2	65.7	45.5	45	291
	LocalVMamba-S [29]	48.4	69.9	52.7	43.2	66.7	46.5	69	414
	SiMBA-S [39]	46.9	68.6	51.7	42.6	65.9	45.8	60	382

Table 5. Comparison of Mamba models and other state-of-the-art models on COCO dataset (Mask R-CNN 3 × *MS* schedule).

Model	Backbone	$\mathbf{AP}^b$	$\mathbf{AP}^b_{50}$	$\mathbf{AP}^b_{75}$	$\mathbf{AP}^m$	$\mathbf{AP}^m_{50}$	$\mathbf{AP}^m_{75}$	Params (M)	FLOPs (G)
CNN	ConvNeXt-T [104]	46.2	67.9	50.8	41.7	65.0	44.9	48	262
Transformer	Swin-T [91]	50.5	69.3	54.9	43.7	66.6	47.1	86	745
	Swin-S [91]	51.8	70.4	56.3	44.7	67.9	48.5	107	838
	Swin-B [91]	51.9	70.9	56.5	45.0	68.4	48.7	145	982
	ViT-Adapter-T [102]	46.0	67.6	50.4	41.0	64.4	44.1	28.1	-
	ViT-Adapter-S [102]	48.2	69.7	52.5	42.8	66.4	45.9	47.8	-
	ViT-Adapter-B [102]	49.6	70.6	54.0	43.6	67.7	46.9	120.2	-
	PVT-Tiny [103]	39.8	62.2	43.0	37.4	59.3	39.9	32.9	-
	PVT-Small [103]	43.0	65.3	46.9	39.9	62.5	42.8	44.1	-
	PVT-Medium [103]	44.2	66.0	48.2	40.5	63.1	43.5	63.9	-
	PVT-Large [103]	44.5	66.0	48.3	40.7	63.4	43.7	81.0	-
	ViL-Tiny-RPB [96]	44.2	66.4	48.2	40.6	63.2	44.0	26.9	199
	ViL-Small-RPB [96]	47.1	68.7	51.5	42.7	65.9	46.2	45.0	277
	ViL-Medium-RPB [96]	48.9	70.3	54.0	44.2	67.9	47.7	60.1	352
	ViL-Base-RPB [96]	49.6	70.7	54.6	44.5	68.3	48.0	76.1	439
	Focal-Tiny [97]	47.2	69.4	51.9	42.7	66.5	45.9	48.8	291
	Focal-Small [97]	48.8	70.5	53.6	43.8	67.7	47.2	71.2	401
	Focal-Base [97]	49.0	70.1	53.6	43.7	67.6	47.0	110.0	533
Mamba	VMamba-T [27]	48.5	69.9	52.9	43.2	66.8	46.3	42	262
	VMamba-S [27]	49.7	70.4	54.2	44.0	67.6	47.3	64	357
	LocalVMamba-T [29]	48.7	70.1	53.0	43.4	67.0	46.4	45	291
	LocalVMamba-S [29]	49.9	70.5	54.4	44.1	67.8	47.4	69	414

From Table 5, it is evident that the characteristics of the three models become more pronounced under the Mask R-CNN 3 × *MS* schedule. In object detection and instance segmentation tasks, CNN models, while suitable for resource-constrained environments, are somewhat less accurate. Transformer models are suited for high-performance computing environments, demanding significant computational resources but achieving excellent performance. Mamba models balance performance and computational complexity well, providing outstanding performance with relatively lower computational resource requirements.

3.4.3. Analysis and Comparison in Semantic Segmentation Tasks

The ADE20K dataset was used for semantic segmentation tasks. As shown in Table 6, we used the mIoU metric to compare and analyze the performance of various models on this dataset. mIoU, or mean intersection over union, is the standard metric for measuring model performance in segmentation tasks, reflecting the model's accuracy in handling complex scenes and objects of different scales.

Table 6. Comparison of Mamba models and other state-of-the-art models on ADE20K datasets.

Model	Backbone	Image Size	Params (M)	FLOPs (G)	mIoU (SS)	mIoU (MS)
CNN	ResNet-50 [98]	512^2	67	953	42.1	42.8
	ResNet-101 [98]	512^2	85	1030	42.9	44.0
	ConvNeXt-T [104]	512^2	60	939	46.0	46.7
	ConvNeXt-S [104]	512^2	82	1027	48.7	49.6
	ConvNeXt-B [104]	512^2	122	1170	49.1	49.9
Transformer	Swin-T [91]	512^2	60	945	44.4	45.8
	Swin-S [91]	512^2	81	1039	47.6	49.5
	Swin-B [91]	512^2	121	1188	48.1	49.7
	Focal-T [97]	512^2	62	998	45.8	47.0
	Focal-S [97]	512^2	85	1130	48.0	50.0
	Focal-B [97]	512^2	126	1354	49.0	50.5
	DeiT-S + MLN [105]	512^2	58	1217	43.8	45.1
	DeiT-B + MLN [105]	512^2	144	2007	45.5	47.2
Mamba	Vim-Ti [26]	512^2	13	-	41.0	-
	Vim-S [26]	512^2	46	-	44.9	-
	VMamba-T [27]	512^2	55	939	47.3	48.3
	VMamba-S [27]	512^2	76	1037	49.5	50.5
	VMamba-B [27]	512^2	110	1167	50.0	51.3
	VMamba-S [27]	640^2	76	1620	50.8	50.8
	PlainMamba-L1 [28]	640^2	35	174	44.1	-
	PlainMamba-L2 [28]	640^2	55	285	46.8	-
	PlainMamba-L3 [28]	640^2	81	419	49.1	-
	LocalVim-T [29]	512^2	36	181	43.4	44.4
	LocalVim-S [29]	512^2	58	297	46.4	47.5
	LocalVMamba-T [29]	512^2	57	970	47.9	49.1
	LocalVMamba-S [29]	512^2	81	1095	50.0	51.0
	EfficientVMamba-T [30]	512^2	14	230	38.9	39.3
	EfficientVMamba-S [30]	512^2	29	505	41.5	42.1
	EfficientVMamba-B [30]	512^2	65	930	46.5	47.3
	SiMBA-S [39]	512^2	62	1040	49.0	49.6

In semantic segmentation tasks, CNN models exhibit relatively stable mIoU performance under both single-scale (SS) and multi-scale (MS) settings. ConvNeXt [104] models perform slightly better than ResNet [98] models, with ConvNeXt-B achieving a mIoU (SS) of 49.1 and a mIoU (MS) of 49.9. Compared to other models, CNNs offer stable and reliable performance, making them suitable for resource-limited applications, though they may fall short in tasks requiring high precision.

Among Transformer models, Swin-B [91] stands out on the ADE20K dataset, with a mIoU (SS) of 48.1 and a mIoU (MS) of 49.7. Focal [97] models also perform well, particularly Focal-B, which achieved mIoU (SS) and mIoU (MS) scores of 49.0 and 50.5, respectively. However, Transformer models generally demand significant computational resources. For instance, while delivering excellent performance, Focal-B required 126M parameters and 1354G FLOPs. Similarly, DeiT-B + MLN [105] required 144M parameters and 2007G FLOPs.

In the Mamba model category, the VMamba [27] backbones showed outstanding performance under both single-scale and multi-scale tests, particularly VMamba-S (640^2), which achieved a mIoU (SS) and mIoU (MS) of 50.8. The PlainMamba [28] and LocalVim [29] backbones also provide high mIoU scores with lower parameters and FLOPs, such as PlainMamba-L3, which achieved a mIoU (SS) of 49.1.

Therefore, while both Transformer and Mamba models perform well in semantic segmentation tasks, Transformer models require high computational resources, making them suitable for high-performance computing environments. Mamba models, on the other hand, offer a diverse range of options, including the high-performance VMamba [27] backbones and the low-complexity yet high-performance LocalVim [29] backbones, catering to various computational resource scenarios. Mamba models balance performance and computational complexity well, making them ideal for applications requiring high precision but with limited computational resources.

4. Visual Mamba in Application Fields

Mamba-based modules increase the efficiency of processing sequential data, adeptly capturing long-range dependencies and seamlessly integrating into existing systems. In medical visual tasks and remote sensing images, where inputs usually entail high-resolution data, Mamba emerges as a pivotal tool for augmenting various visual tasks, especially those pertinent to medical applications.

In the current section, we began by highlighting the contributions of Mamba-based modules in enhancing general visual-related tasks. Then, we delved into their specific impact on medical visual tasks and remote-sensing images.

4.1. General Visual Mamba

General vision-related tasks are categorized into high/mid-level vision and low-level vision. high/mid-level vision includes recognition tasks for different input formats (pictures, videos, and 3D representation), including segmentation, object detection, classification, and prediction. By contrast, low-level vision includes restoration, generation etc., as shown in Table 7.

Table 7. Representative works of general visual mamba.

Category	Sub-Category	Method	Efficiency	Code
Backbone	Visual Mamba	Vision Mamba [26]	Params Vim-Ti: 7, Vim-S: 26	✓
		VMamba [27]	FLOPs Base: 15.2 Small: 9.1, Tiny: 4.5	✓
		PlainMamba [28]	FLOPs PlainMamba-L1: 3.0 PlainMamba-L2: 8.1 PlainMamba-L3: 14.4	✓
		LocalMamba [29]	FLOPs LocalVMamba-T: 5.7 LocalVMamba-S: 11.4	✓
		Mamba-ND [38]	Params Mamba-2D: 24 Mamba-3D: 36	✓
		SiMBA [39]	-	✓
		RES-VMAMBA [48]	-	✓
	Efficient Mamba	EfficientVMamba [30]	-	✓
		MambaMixer [37]	-	✓

Table 7. *Cont.*

Category	Sub-Category	Method	Efficiency	Code
High/Mid-level vision	Object detection	SSM-ViT [79]	Params 17.5	✗
	Segmentation	ReMamber [50]	-	✗
		Sigma [49]	-	✓
	Video classification	ViS4mer [80]	Memory 5273.6	✓
	Video understanding	Video Mamba Suite [43]	-	✓
		VideoMamba [33]	FLOPs VideoMamba-Ti: 7.1 VideoMamba-S: 28 VideoMamba-M: 83.1	✓
		SpikeMba [93]	-	✗
	Multi-Modal understanding	Cobra [106]	-	✓
		ReMamber [50]	-	✗
		VL-Mamba [42]	-	✗
	Video prediction	VMRNN [47]	Params 2.6, FLOPs 0.9	✓
		HARMamba [40]	FLOPs PAMAP2:279.21 UCI:237.83 UNIMIB HAR:238.36 WISDM:256.52	✗
Low-level vision	Image super-resolution	MMA [41]	-	✗
	Image restoration	MambaIR [57]	Params 16.7	✓
		SERPENT [58]	-	✗
		VmambaIR [32]	Params 10.50, FLOPs 20.5	✓
	Image dehazing	UVM-Net [85]	Params 19.25	✓
	Image derain	FreqMamba [68]	Params 14.52	✗
	Image deblurring	ALGNet [107]	FLOPs 17	✗
	Visual generation	MambaTalk [74]	-	✗
		Motion Mamba [34]	-	✓
		DiS [88]	-	✓
		ZigMa [31]	-	✓
	Point cloud	3DMambaComplete [83]	Params 34.06, FLOPs 7.12	✗
		3DMambaIPF [108]	-	✗
		Point Cloud Mamba [109]	Params 34.2, FLOPs 45.0	✗
		POINT MAMBA [44]	Memory 8550	✓
		SSPointMamba [110]	Params 12.3, FLOPs 3.6	✓
	3D reconstruction	GAMBA [111]	-	✗
	Video generation	SSM-based diffusion model [89]	-	✓

For the efficiency, inference speed is in ms, memory is in MB, Params is in M, and FLOPS is in G.

4.1.1. High/Mid-Level Vision

The visual Mamba backbone [26–29,38] had decent performance in classification, object detection, and segmentation. SSM-ViT [79] was designed for object detection using event cameras. Differently from standard frame-based cameras, event cameras record per-pixel relative brightness changes in a scene as they occur. Therefore, object detection with event cameras requires processing an asynchronous stream of events in a four-dimensional spatio-temporal space. Earlier studies used RNNs architectures with convolutional or attention mechanisms to develop models exhibiting superior performance on downstream tasks using event camera data. However, these models usually suffer from slow training. As a response, the SSM-ViT block was introduced by leveraging an SSM for efficient event-based information processing. It explores two strategies to mitigate aliasing effects when deploying the model at higher frequencies.

For referencing image segmentation (RIS), a difficult problem in the field of multi-modal comprehension, ReMamber [50] was introduced, utilizing Mamba's notable advances in efficient training and inference with linear complexity. Distinguished from conventional segmentation, RIS entails identifying and segmenting specific objects in images according to textual descriptions. The ReMamber architecture comprises several Mamba Twister blocks, each featuring multiple VSS blocks and a Twisting layer. The Mamba Twister block is a multi-modal feature fusion block that blends textual and visual features into a single output, which is the fused multi-modal feature representation. The last segmentation mask is generated by retrieving intermediate features after every Mamba Twister block and feeding them into a flexible decoder. The VSS layers are tasked with extracting visual features, while the Twisting layer primarily captures effective visual-language interactions. The experiments conducted by authors on various RIS datasets produced cutting-edge outcomes. Sigma [49] presented a novel network tailored for multimodal semantic segmentation tasks. Following each Mamba Twister block, intermediate features are retrieved and input into a flexible decoder, which generates the segmentation mask at the end. Furthermore, a channel-aware Mamba decoder and an attention-based Mamba fusion mechanism were presented. During the decoding phase, the fused features undergo further enhancement through channel-aware VSS (CVSS) blocks, adept at capturing multi-scale long-range information and facilitating cross-modal information integration.

Unlike transformers that depend on quadratic complexity attention mechanisms, Mamba, as a pure SSM-based model, excels in handling long sequences with linear complexity and is particularly adept at processing lengthy videos at high resolutions. ViS4mer [80] serves as a model primarily used for recognizing and classifying long videos, especially for understanding and categorizing lengthy movie clips. ViS4mer is composed of two primary parts: a multi-scale temporal S4 decoder suited for further long-range temporal reasoning, and a standard Transformer encoder intended for short-distance spatiotemporal feature extraction from videos. The multi-scale temporal S4 decoder is based on SSM and makes use of the ability of the core SSM to identify long-range correlations in consecutive data, in order to reduce the computational cost of the model.

The Video Mamba Suite [43] is not a novel method; rather, it investigates and evaluates SSM's potential, embodied by Mamba, in tasks related to comprehending videos. The decomposed bidirectionally Mamba (DBM) block is an improved version of the ViM block that shares the SSM parameters in both scanning directions, while allowing the input projector to be separated. They classify Mamba into four distinct roles for modeling videos and compare it with existing Transformer-based models to evaluate its effectiveness in various video understanding tasks. The 14 models and modules that make up the Video Mamba Suite were used to assess performance on 12 different video comprehension tasks. The experiments showed that Mamba is applicable in video analysis and can be used for more complex, multimodal video understanding challenges. Apart from the Video Mamba Suite, VideoMamba [33] was proposed for video understanding tasks, with a specific focus on addressing the following two major challenges: local redundancy and global dependencies. The study evaluated VideoMamba's capabilities across four key aspects: scalability in the video domain, sensitivity to short-term action recognition, advantages in long-term video understanding, and compatibility with other modalities. To enhance model scalability in the visual domain, a self-distillation strategy is used in VideoMamba. This approach significantly enhances the model's performance as both the model and input sizes increase, without the need for pretraining on large-scale datasets. While the ViM block enhances the model's spatial perception capabilities, VideoMamba extends this capability to 3D video understanding by including spatio-temporal bidirectional scanning. By extending the ViM block, VideoMamba achieves a significant elevation in processing speed and a reduction in computational resource consumption without compromising performance. SpikeMba [93] presents a pioneering multimodal video content understanding framework geared towards the task of temporal video localization. The proposed framework amalgamates spiking neural networks (SNNs) with SSM blocks in order to discern intricate relationships within multimodal input features. The spike saliency detector (SSD) leverages SNNs' thresholding mechanism to generate sets of saliency proposals, denoting highly pertinent or salient instances in a video via spikes. Furthermore, based on SSM, the multimodal relevance Mamba block (MRM) retains linear complexity within respect to input size, while increasing long-range dependency modeling.

Multimodal large language models (MLLMs), on the basis of transformers, have demonstrated significant success across diverse domains, albeit with secondary computational intricacy. In order to improve these models' efficiency, Mamba has been included in this work. In order to produce the most effective multimodal representation, Cobra [106] combines an effective Mamba language model into the visual modality and investigates several modal fusion strategies. It comprises three parts, including a visual encoder, a projector, and a Mamba backbone. The visual encoder extracts the visual representation of the image, while the projector adjusts the dimensions of the visual representation to align with the Mamba language model's tokens. The Mamba backbone consists of 64 identical basic blocks, maintaining connectivity and RMSNorm, and transforms the combined visual and textual embeddings into target token sequences in an autoregressive manner. VL-Mamba [42] comprises a pretrained visual coder, a randomly initialized MMC, and a pretrained Mamba LLM. The visual coder takes the original picture and uses the ViT architecture to create a series of patch features. Regarding the MMC, it introduces a 2D visual selective scanning mechanism tailored for computer vision tasks, the state-space model is designed for 1D sequential data with causality, while visual sequences from the visual coder are 2D non-causal data. The study explored three multimodal connector variants including MLP, VSS-MLP, and VSS-L2. Initially, input images are processed into visual characteristics by the coder. After feeding these visual series to the MMC, the output vectors that are produced are coupled with a tokenized text question and sent to the Mamba LLM, which produces the appropriate response. Through the synergistic combination of these components, the integration and processing of visual and verbal information are optimized . ReMamber [50] addresses the referential image segmentation (RIS) task, including identifying and dividing particular elements in a picture according to written descriptions. The architecture combines Mamba with multimodal Mamba Twister blocks to simulate image–text interactions explicitly through a distinctive channeling and spatial warping mechanism, therefore fusing textual and visual features. After each Mamba Twister block, ReMamber extracts intermediate features and passes them through a versatile decoder to produce the segmentation mask at the end.

To tackle the unparalleled challenge of predicting temporal and spatial dynamics for spatio-temporal forecasting in videos, the VMRNN cell [47] introduced a novel recurrent unit designed to efficiently handle spatio-temporal prediction tasks. By recognizing the challenges in processing extensive global information, the VMRNN cell integrates VSS blocks with an LSTM architecture to leverage the long-sequence modeling abilities of VSS blocks and the spatio-temporal representation capabilities of LSTM. This integration can enhance the accuracy and efficiency of spatio-temporal predictions. The model performs image-level analysis by segmenting each frame into patches, which are subsequently flattened and processed through an embedding layer. Moreover, this process enables the VMRNN layer to extract and predict spatio-temporal features effectively. HARMamba [40] builds on ViT blocks for activity recognition and achieves superior performance, lowering reducing computational and memory overhead in activity recognition tasks.

4.1.2. Low-Level Vision

In the realm of image super-resolution, meet more areas (MMA) [41] stands out as a novel model designed for super-resolution tasks. By building on the ViM block, MMA aims to enhance performance by activating a wider range of areas within images. On this basis, MMA adopts several key strategies, including adding ViM to modules in MetaFormer style, pre-training ViM on larger datasets, and employing complementary attention mechanisms. MMA comprises the following three primary modules: shallow feature extraction, deep feature extraction, and high-quality reconstruction. By leveraging the ViM module, MMA effectively models global information and further expands the activation region through attention mechanisms.

Existing restoration backbones often confront a dilemma between global receptive fields and efficient computation, hindering their application in practice, while Mamba has a lot of promise for linear complexity long-range dependency modeling, which can also offer an efficient a way to resolve the above dilemma. MambaIR [57] aimed to address the problem by introducing local enhancement and channel attention mechanisms to enhance the standard Mamba model. The methodology of the model mainly consists of three stages: shallow feature extraction, deep feature extraction, and high-quality image reconstruction. Among them, the deep feature extraction stage utilizes multiple residual state space blocks (RSSBs) for feature extraction, adding a VSS block before the channel attention block designed in previous transformer-based restoration networks. SERPENT [58] has a hierarchical architecture, processing input images in a multi-scale manner, including processing steps such as segmentation, embedding, downsampling, and upsampling, and introduces jump connections to facilitate information flow. Among them, the Serpent block is the main processing unit, consisting of multiple VSS blocks stacked on top of each other. Serpent reduces the computational effort, GPU memory demand, and model size significantly, while preserving good reconstruction quality by combining the benefits of transformers and convolutional networks. VmambaIR [32] put forward the OSS module to comprehensively and efficiently model image features from six directions. In addition, the omnidirectional selective scanning mechanism overcomes the unidirectional modeling limitation of SSMs and accomplishes thorough pattern identification and modeling by simulating the three-dimensional visual information flow.

UVM-Net [85] refers to a novel single-image defogging network architecture exhibiting effective performance by merging the long-range dependency modeling capacity of SSMs with the local feature extraction of convolutional layers. The method employs an encoder–decoder network architecture, and the critical component is the ViM block, which leverages the long-range modeling capability of SSMs through rolling the feature map over the channel domain. Differently from U-Mamba [84] and Mamba-UNet [51], with long-range dependencies on the non-channel domain, a different feature map dimension is established using the ViM block.

Images lose important frequency information under the influence of raindrops, affecting the performance of visual perception and advanced visual tasks. FreqMamba [68] is a novel image de-raining method combining Mamba modeling and frequency analysis techniques to deal with the image de-raining problem. Specifically, FreqMamba contains three branching structures, including spatial Mamba, frequency band Mamba, and Fourier global modeling. Spatial Mamba processes raw image features to extract details and correlations within the image. Frequency band Mamba employs wavelet packet transform (WPT) to decompose the input features into spectral features in different frequency bands and scan them over the frequency dimension. Fourier modeling, i.e., processing an input using Fourier transform, captures the global degradation patterns that can affect an image. Numerous tests have shown that FreqMamba works better than current state-of-the-art techniques in terms of both visual and quantitative aspects.

Image deblurring is a traditional issue in low-level computer vision, with the goal of restoring crisp, high-quality images from hazy input photographs. ALGNet [107] is an efficient image deblurring network utilizing selective state-space models (SSM) to aggregate rich and accurate features. The network consists of multiple ALGBlocks, each of which contains a CLGF module that captures local and global features and a feature aggregation module FA. The CLGF module captures long-range dependent features using a SSM and employs a channel-attention mechanism to lower local pixel forgetting and channel redundancy. Through weight calibration, the FA module highlights the significance of local features in recovery.

The efficiency of Mamba makes a significant contribution to mitigating the high computational complexity associated with training generation tasks. To address the change in generating long and diverse sequences with low latency, MambaTalk [74] implements a two-stage modeling strategy with discrete motion priors to improve the quality of gestures and employs a mamba block to enhance gesture diversity and rhythm through multimodal integration. Motion Mamba [34] was introduced to construct a motion generation model based on Mamba, leveraging an efficient hardware-aware design. Motion Mamba consists of the following two main components: a hierarchical temporal Mamba (HTM) block for temporal data handling; and for analyzing latent postures, a bidirectional spatial Mamba (BSM) block. To maintain motion consistency across frames, the HTM block employs several separate SSM modules within a U-Net architecture that is balanced. Meanwhile, the BSM block enhances the accuracy of motion generation within a temporal frame by processing latent poses bidirectionally. Diffusion state space odels (DiS) [88] substitute the conventional U-Net backbone in diffusion models with SSM. All inputs are taken into account by this system, including time, conditions, and tokens and noisy image patches. To address the oversight of spatial continuity in the scanning scheme of existing Mamba-based vision methods, Zigzag Mamba [31] was introduced as a straightforward, plug-and-play solution inspired by DiT-style approaches. Essentially, it retains the scanning scheme of plain Mamba but expands it from four to eight schemes by incorporating mirror flipping schemes, as displayed in Figure 4f. Then, Zigzag Mamba was integrated using the framework of the stochastic interpolant, forming ZigMa, to investigate the diffusion model's scalability using high-resolution visual datasets. GAMBA [111] introduces a sequential network based on Mamba, allowing for linear sequence length scalability and context-dependent reasoning. This architecture accommodates many Gaussians for the 3D Gaussian splatting process. To deal with the issue of the increase in quadratic memory consumption with sequence length in traditional attention-based video generative diffusion models, an SSM-based diffusion model [89] was introduced for generating longer video sequences. Similarly to ViS4mer [80], the SSM-based diffusion model re-imagines the attention modules within the conventional temporal layers of video diffusion models. It can replace them with a ViM block designed to record the temporal changes of video data and an MLP to improve model performance. Moreover, this innovative approach significantly mitigates memory consumption for extended sequences.

In 3D vision tasks, the irregularity and sparsity of point cloud data present considerable hurdles. While transformers exhibit promise in exploring point cloud data, due to their strong global information modeling capability, but their computational complexity escalates significantly as the input length increases. This limitation restricts their applicability, particularly in long sequence models. SSPoint Mamba [110] employs embedded point blocks as inputs and enhances an SSM's capacity for global modeling by rearranging the structure to produce a geometric scanning order that is more logical. Then, rearranged point tokens undergo processing via multiple Mamba blocks in order to causally represent the structure of the point cloud, showcasing effectiveness across different point cloud analysis tasks. 3DMambaComplete [83] tackles the computational complexity challenges of point cloud completion by leveraging the Mamba framework. The method involves downsampling incomplete point clouds, enhancing feature learning with a Mamba encoder, predicting and refining hyperpoints, dispersing hyperpoints to various 3D locations through learned offsets, and performing point deformation to generate complete point clouds. Structured state-space modeling optimizes shape reconstruction by predicting hyperpoints and controlling the deformation at each hyperpoint location. 3DMambaIPF [108] concentrates on denoising large-scale point cloud data. Integrating Mamba into a filtering module, Mamba-Denoise, can enable accurate and fast modeling of long sequences of point cloud features. By employing iterative point cloud filtering and loss functions, including reconstruction loss and differentiable rendering loss, it minimizes the distance between noisy and real point clouds, optimizing visual boundaries, and enhancing denoising realism. Point Cloud Mamba [109] combines local and global modeling frameworks and introduced a consistent traversal serialization (CTS) approach to convert 3D point cloud data into 1D point sequences, while preserving spatial adjacency. Moreover, it incorporates point cuing and position encoding based on spatial coordinate mapping to improve Mamba's efficiency in processing point sequences and injecting position information. Point Mamba [44] addresses causality in SSM for point cloud data through introducing an octree-based ordering strategy. Additionally, it integrates bi-directional selective scanning mechanisms into point Mamba blocks to adjust sequence order dependency, thus enhancing its adaptability to point cloud structures.

4.2. Medical Visual Mamba

Transformers [8] have exerted a deep influence on the field of medical imaging with their ability to master complex data representations. They have led to notable advancements across various imaging modalities, including radiography [112], endoscopy [113], computed tomography (CT) [114], ultrasound images [115], and magnetic resonance imaging (MRI) [116]. However, since most medical images are high-resolution and detailed, transformer models typically require considerable computational resources, which scale quadratically with image resolution.

Recently, the medical imaging field has experienced a surge in the development of Mamba-based methodologies, particularly following the introduction of VMamba. The current section provides detailed examples of these design choices, further dividing them into 2D and 3D-based approaches based on the input type, as displayed in Table 8.

Table 8. Representative works of medical visual mamba.

Category	Sub-Category	Method	Efficiency	Code
2D	Segmentation	Mamba-UNet [51]	-	✓
		H-vmunet [55]	Memory 0.676 Params 8.97	✓
		Mamba-HUNet [59]	-	✗
		P-Mamba [117]	Inference speed 23.49 Memory 12.22 Params 183.37 FLOPs 71.81×10^{9}	✗
		ProMamba [78]	Params 102	✗
		TM-UNet [60]	Params 14.86 Total Params 8.41 FLOPs 3.42	✗
		Semi-Mamba-UNet [52]	-	✓
		Swin-UMamba [61]	Params 28 FLOPs 18.9	✓
		UltraLight VM-UNet [62]	Params 0.049 GFLOPs 0.060	✓
		U-Mamba [84]	-	✓
		VM-UNet [63]	Params 34.62 FLOPs 7.56 FPS 20.612	✓
		VM-UNET-V2 [64]	Params 17.91 FLOPS 4.40 FPS 32.58	✓
		Weak-Mamba-UNet [76]	-	✓
	Radiation dose prediction	MD-Dose [90]	Inference speed 18 Params 30.47	✓
	Classification	MedMamba [65]	-	✓
		MambaMIL [118]	-	✓
	Image reconstruction	MambaMIR /MambaMIR-GAN [56]	-	✓
	Exposure correction	FDVM-Net [81]	Inference speed 22.95	✓

Table 8. *Cont.*

Category	Sub-Category	Method	Efficiency	Code
3D	Segmentation	LMa-UNet [45]	-	✓
		LightM-UNet [86]	Params 1.87 FLOPs 457.62 × 10^9	✓
		SegMamba [71]	Inference speed 151	✓
		T-Mamba [77]	-	✓
		Vivim [35]	FPS 35.33	✓
	Classification	CMViM [82]	Params 50	✓
	Motion tracking	Motion-Guided Dual-Camera Tracker [46]	-	✓
	Backbone	nnMamba [70]	Params 15.55 FLOPs 141.14	✗
	Image registration	VMambaMorph [53]	Inference speed 19 Memory 3.93 Params 9.64	✓
		MambaMorph [87]	Inference speed 27 Memory 7.60 Params 7.59	✓

For efficiency, inference speed is in ms, memory is in Gb, Params is in M, and FLOPS is in G.

4.2.1. Two-Dimensional Medical Images

Mamba has exhibited impressive potential in 2D medical segmentation, as displayed in Figure 6. Here, we discuss in detail some methods that explore the use of mamba to model the global structure information of 2D medical segmentation.

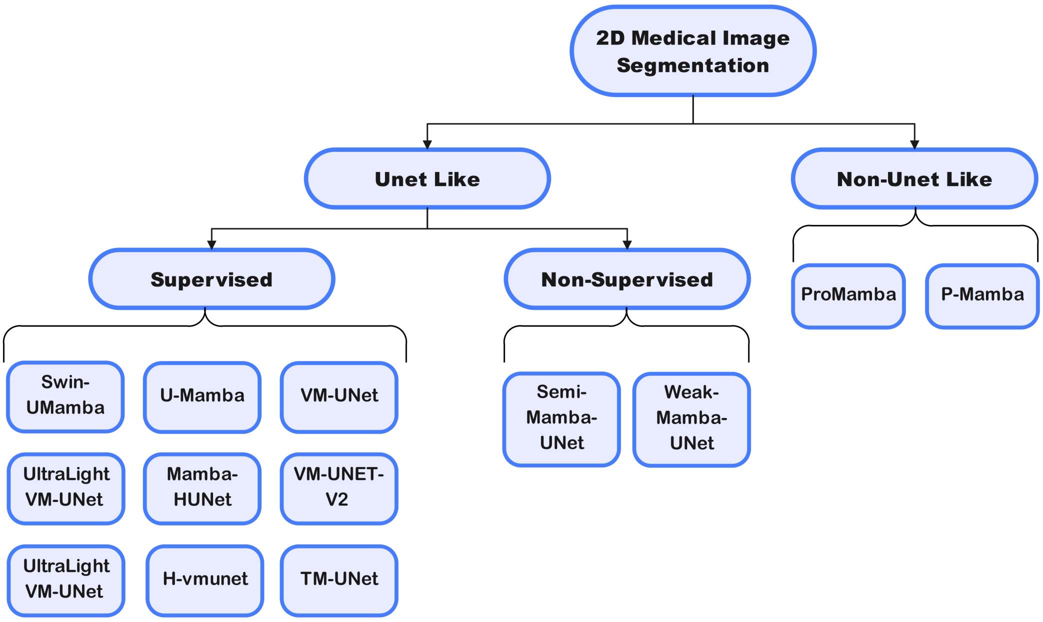


Figure 6. An overview of Mamba models used for segmentation task in 2D medical images.

Most of the innovative architectures that have been developed are based on U-Net, it has shown outstanding results in a variety of medical image segmentation challenges. U-Mamba [84] refers to the first extension of the mamba model to the U-Net framework for visual segmentation in biomedical imaging, addressing the challenge of long-range dependency modeling, which is captured by a hybrid CNNs-SSM block. Wu et al. introduced the High-order Vision Mamba UNet (H-vmunet) [55], an improvement on U-Mamba, which employs a high-order 2D-selective scan at each interaction to bolster the learning of local features, while minimizing the incorporation of redundant information. Shortly after their initial release, the team expanded their work by introducing the UltraLight VM-UNet [62]. This new iteration was developed through in-depth analysis of the critical factors affecting parameter efficiency within the Mamba framework. This resulted in a significantly more lightweight model with a mere 0.049M parameters and a computational efficiency of only 0.060 GFLOPs. Moreover, Mamba-UNet [51] combines the encoder–decoder architecture of U-Net with the capabilities of Mamba and maintains spatial information at different network scales based on jump connections. A Visual Mamba-based VSS block is used, which utilizes linear embedding layers and deep convolution to extract features, while downsampling and upsampling are facilitated by multiple merge operations and extension layers for comprehensive feature learning.

Both Pyramid ViT (PVT) and Swin-Unet are pioneering hierarchical designs that apply visual tasks and propose progressive shrinking pyramids and spatial-reduction attention. By drawing inspiration from PVT and Swin-Unet, Ruan et al. introduced VM-UNet [63], a foundational model for purely SSM-based segmentation in medical imaging. This model shows the capabilities of SSMs in medical image segmentation and is made up of three primary components: an encoder, a decoder, and skip connections. By building on their previous work, the team proposed VM-UNET-V2 [64]. The purpose of the visual state space (VSS) block was to obtain a greater range of contextual information. A semantics and detail infusion (SDI) mechanism was also implemented to enhance the fusion of low-level and high-level features. Mamba-HUNet [59], as another multi-scale hierarchical upsampling network, incorporates the Mamba technique. Additionally, it preserves spatial information by extracting hierarchical characteristics through the use of patch merging layers and visual state space blocks. TM-UNet [60] introduced improvements to the bottleneck layer. This architecture proposes triplet SSMs as the bottleneck layer, marking the first attempt to combine spatial and channel data using a pure SSM technique. Current Mamba-based models miss out on possible advantages, because they are mostly developed from scratch. As a new Mamba-based model tailored explicitly for medical image segmentation tasks, Swin-UMamba [61] leveraged the strengths of ImageNet-based pretraining.

Previous discussions have primarily concentrated on supervised learning methods, but other supervisory approaches have also been explored. Semi-Mamba-UNet [52] combines a visual Mamba-based U-shape encoder–decoder with a traditional CNNs-based UNet in a semi-supervised learning framework. In order to enhance feature learning, especially with regard to unlabeled data, it presents a self-supervised pixel-by-pixel contrastive learning approach that makes use of two projectors. Weak-Mamba-UNet [76] refers to a novel weakly-supervised learning structure for medical image segmentation, combining CNNs, ViT, and the VMamba. It employs a cooperative, cross-supervisory approach using pseudo labels for iterative network learning and improvement, with a focus on scribble-based annotations.

Some segmentation approaches diverge from UNet architectures. P-Mamba [117] presents a novel dual-branch framework for highly efficient left ventricle segmentation in pediatric echocardiograms. This model features an innovative DWT-based encoder branch equipped with Perona–Malik diffusion (PMD) blocks. Moreover, to bolster computational and memory efficiency, P-Mamba adopts vision mamba layers within its vision mamba encoder branch. PromptMamba [78] represents a groundbreaking integration of vision Mamba and prompt technologies, marking a significant milestone as the first model to leverage the Mamba framework for the specific task of polyp segmentation.

In addition, mamba has also expanded into research in 2D medical imaging beyond segmentation, enhancing the precision and speed of image analysis to support diagnosis and treatment planning. Classification is a vital and fundamental task in the area of analysis of medical images. Yue et al. invented Vision Mamba for this purpose, which was also known as MedMamba [65]. They created a brand-new module called Conv-SSM, which combines SSM's long-range dependency capture with a convolutional layers' local feature extraction. This enables efficient modeling of medical images from different modalities. Furthermore, MambaMIL [118] introduced the Sequence Reordering Mamba (SR-Mamba), a model recognizing the order and distribution of instances in long sequences, to effectively harness valuable embedded information. Since accurate and efficient clinical decision-making depends on high-quality and high-fidelity medical pictures, image reconstruction plays a critical role in improving diagnostic processes. Huang et al. [56] developed MambaMIR, a model leveraging Mamba technology for the reconstruction of medical images, alongside its advanced counterpart, MambaMIR-GAN, which incorporates generative adversarial networks. Zheng et al. introduced FDVision Mamba (FDVM-Net) [81], a frequency-domain-based structure that effectively corrects image exposure by restoring endoscopic pictures' frequency domain, as recorded endoscopic images often suffer from exposure abnormalities. As shown in specialized areas, MD-Dose [90], a cutting-edge diffusion model based on the Mamba architecture, was designed to accurately predict radiation therapy dose distribution for thoracic cancer patients.

4.2.2. Three-Dimensional Medical Images

Three-dimensional image analysis in medical imaging enables more accurate and comprehensive diagnoses by providing a detailed view of complex anatomical structures. Gong et al. presented nnMamba [70], an innovative architecture designed for 3D medical imaging applications, which integrates local and global relationship modeling via the MICCSS (Mamba-In-Convolution with Channel-Spatial Siamese input) module. nnMamba was tested on a comprehensive benchmark of six datasets for three crucial tasks, including segmentation, classification, and landmark detection, showcasing its capability for long-range relationship modeling at channel and spatial levels.

Precise 3D segmentation results can alleviate physicians' diagnostic workloads in disease management. SegMamba [71], a cutting-edge architecture, is the first technique to use Mamba expressly for precise 3D medical imaging segmentation. It introduced a tri-orientated Mamba (ToM) module for modeling 3D features from three directions and a gated spatial convolution (GSC) module to enhance spatial feature representation before each ToM module. By similarly employing a U-shaped architecture, LightM-UNet [86] uses the residual vision Mamba layer alone in a Mamba-only method to model large-scale spatial dependencies and extract deep semantic features in a lightweight framework. Both LMa-UNet [45] and T-Mamba [77] built upon the foundation of SegMamba, with improvements made to the Mamba block. A notable aspect of T-Mamba [77] was creating a gate selection unit to adaptively combine two features in the spatial domain with one feature in the frequency domain, whereas LMa-UNet [45] refers to its use of large windows, which outperformed small kernel-based CNNs and small window-based Transformers in local spatial modeling. This marks the first instance of incorporating frequency-based features into the vision Mamba framework. The issue of long-term temporal dependency in video scenarios was also addressed by developing a general framework called Vivim [35], built on Mamba for video vision. Based on a specifically engineered temporal Mamba block, this model effectively compresses long-term spatiotemporal data into sequences of different scales.

In terms of image registration tasks, MambaMorph [87] introduced a groundbreaking multi-modality deformable registration framework that enhances medical image analysis by combining a Mamba-based registration module with an advanced feature extractor for efficient spatial correspondence and feature learning. The VMambaMorph [83] model further enhanced its VMamba-based block by incorporating a 2D cross-scan module, redesigned to process 3D volumetric features in an efficient way.

In other domains, the Contrastive Masked Vim Autoencoder (CMViM) [82] tackles Alzheimer's disease (AD) classification by incorporating vision Mamba (ViM) into a masked autoencoder for 3D multi-modal data reconstruction. Regarding endoscopy skill evaluation, a low-cost motion-guided dual-camera tracker [34] provided reliable endoscope tip feedback, and a Mamba-based motion-guided prediction head (MMH) merged visual tracking with historical motion data based on a SSM.

4.2.3. Challenge

Here, we explore some promising future research directions for vision Mamba in medical image analysis. Challenges include the need for pretraining on large datasets, which could enhance the interpretability of Mamba-based medical imaging approaches, as well as improve robustness against adversarial attacks. There is a need to design efficient Mamba architectures suitable for real-time medical applications and to address the challenges in deploying Mamba-based models in distributed settings.

4.3. Remote Sensing Image

The progress of remote sensing methods has sparked interest in high-resolution Earth observation, with the transformer model providing an optimal solution through its attention mechanism. Its quadratic complexity, however, presents problems with memory consumption and modeling efficiency. The SSM addresses these issues by establishing long-distance dependencies with near-linear complexity, and Mamba can further enhance efficiency through hardware optimization and time-varying parameters. Representative recent work is presented in Table 9.

Table 9. Representative mamba work in remote sensing image.

Category	Method	Highlight	Efficiency	Code
Pan-sharpening	Pan-Mamba [73]	channel swapping Mamba; cross-modal Mamba	Params 0.1827 FLOPs 3.0088	✓
Infrared Small Target Detection	MIM-ISTD [66]	Mamba-in-Mamba architecture	Params 1.16 FLOPs 1.01 Inference speed 30 Memory 1774	✓
Classification	RSMamba [36]	multi-path activation	-	✓
	HSIMamba [72]	process data bidirectionally	Memory 136.53	✓
Image dense prediction	RS-Mamba [69]	omnidirectional selective scan	-	✓
Change detection	ChangeMamba [54]	cross-scan mechanism	-	✓
Semantic segmentation	RS3Mamba [67]	dual-branch network	FLOPs 31.65 Params 43.32 Memory 2332	✓
	Samba [75]	encoder-decoder architecture	Params 51.9	✓

For the Efficiency, Inference speed is in ms, Memory is in MB, Params is in M, and FLOPS is in G.

By drawing inspiration from TNT, Chen et al. introduced a new Mamba-in-Mamba (MiM-ISTD) [66] architecture to enhance the efficiency of infrared tiny target detection. In the proposed approach, local patches are considered "visual sentences", while outer Mamba is utilized to extract global information. Regarding remote sensing image classification, RSMamba [36] features a dynamic multi-path activation mechanism to improve Mamba's capability for handling non-causal data. RS-Mamba [69] is adept at handling very-high-resolution (VHR) remote sensing images for dense prediction tasks, built on an omnidirectional selective scan module to model images from various angles comprehensively. In remote sensing research, it is challenging to classify hyperspectral images because of their high-dimensional complicated data. HSIMamba [72] was designed with a module dedicated to spatial analysis, including multiple spectral bands and three-dimensional spatial structures to take advantage of the rich multidimensional nature of the hyperspectral data and to improve the feature representation capability using linear transformations and activation functions. Through the use of forward and backward spectral dependence capture, HSIMamba uses a bi-directional processing approach that enhances the network's capacity to represent and use spectrum information. In addition, Pan-Mamba [73] offers a novel network in the pansharpening space and modifies two essential elements, channel switching Mamba and cross-modal Mamba, both of which are skillfully designed for effective cross-modal information fusion and interchange. For the first time, ChangeMamba [54] investigated the Mamba architecture's potential for distant sensing change detection (CD) activities. For binary change detection (BCD), semantic change detection (SCD), and building damage assessment (BDA) tasks, the MambaBCD, MambaSCD, and MambaBDA network frameworks were built. Three spatio-temporal connection modeling mechanisms were proposed to completely learn spatio-temporal features. ChangeMamba employs selective state-space modeling to capture long-range dependent features and maintains linear computational complexity while providing a visual Mamba architecture to learn global spatial context information. Semantic segmentation of remotely sensed images is crucial for geoscientific research. RS3Mamba [67] is a novel two-branch model developed for this purpose. The model incorporates visual state space (VSS) models, particularly the Mamba architecture, aiming to improve long-range relational modeling capabilities. A co-completion module (CCM) was proposed for feature fusion. The experimental results demonstrated that RS3Mamba had significant advantages over CNNs and transformer-based approaches. With an encoder–decoder architecture, Samba [75] is a revolutionary semantic segmentation system designed especially for high-resolution remote sensing images. Samba blocks act as encoders to extract multilevel semantic information, and Mamba blocks adopt SSMs for capturing global semantic information with linear computational complexity.

5. Conclusions

Mamba is gaining prominence in computer vision for its capability for managing long-range dependencies and its significant computational efficiency relative to transformers. As detailed in recent surveys, various methods have been developed to harness and investigate Mamba's capabilities, reflecting ongoing advancements in the field.

We began by discussing the foundational concepts of SSM and Mamba architectures, followed by a comprehensive analysis of various competing methodologies across a spectrum of computer vision applications. Our survey encompassed state-of-the-art Mamba models designed explicitly for backbone architectures, high/mid-level vision, low-level vision, medical imaging, as well as remote sensing. Moreover, this survey is the first review paper on the recent developments in SSMs and Mamba-based techniques, explicitly concentrating on computer vision challenges. Our goal was to generate more interest among the vision community in utilizing the possibilities of Mamba models, finding solutions to their current limitations.

5.1. Challenges and Limitations

Currently, Mamba has some limitations that vision Mamba aims to address. The original Mamba's one-dimensional selective scanning struggles to capture spatial information in high-dimensional visual data. While the existing methods attempt to mitigate this issue with enhanced scanning mechanisms, they are still insufficient and require further exploration to effectively retain spatial relationships within the Mamba framework. The use of multiple scanning directions and bi-directional approaches can result in significant redundancy and increased computational demands, reducing Mamba's linear complexity advantages. Efficient computation strategies are necessary to improve performance, without excessive resource consumption. Originally designed for causal sequential data, Mamba's selective scanning struggles with non-causal visual data, indicating a need for further refinement to adapt Mamba for visual data processing. Gradient vanishing and exploding are persistent challenges in deep learning, which is exacerbated as datasets grow larger. The Mamba architecture encounters stability issues that need addressing to bolster its robustness and reliability.

Achieving trustworthiness is an ongoing concern. Understanding Mamba's effectiveness in visual tasks poses a significant challenge, requiring a deeper theoretical and empirical grasp of its mechanisms compared to models such as RNNs, CNNs, and ViTs. Improved interpretability will enable more effective application and optimization across diverse visual tasks. Mamba's hidden states tend to accumulate domain-specific information, which can hinder generalization. Its reliance on 1D scanning strategies may introduce biases specific to certain domains, and current techniques often fall short in ensuring domain-agnostic processing. Enhancing Mamba's ability to generalize and its robustness, especially in adversarial contexts, presents a critical challenge.

5.2. Future Directions

Mamba represents an exciting and emerging direction with numerous avenues for exploration. Here, we highlight several promising directions:

Innovative Scanning Mechanisms: To harness the full potential of visual Mamba, new scanning schemes are needed. These schemes should effectively address the non-causal nature of visual data and capture spatial information across multiple dimensions. Developing more sophisticated scanning mechanisms will be crucial for improving Mamba's performance in visual tasks.

Hybrid Architectures: Combining Mamba with other architectures like transformers could mitigate some of its inherent limitations. Hybrid models that integrate Mamba with self-attention mechanisms or CNNs may leverage the strengths of each approach. However, careful design is necessary to ensure these hybrid models do not conflict with their sequence modeling methods and can effectively capture comprehensive and detailed information.

Large-Scale Models and Data Efficiency: As large models become the norm, scaling Mamba while maintaining its computational efficiency is essential. Developing large-scale Mamba models that retain their advantages in sequence modeling could lead to powerful visual foundation models. Additionally, improving data efficiency and enabling optimal performance without reliance on extensive datasets will broaden Mamba's applicability in various tasks.

Integration with Other Methodologies: Mamba can be integrated with other methodologies, such as multi-modal information processing, diffusion models, domain generalization, and visual-language models. Exploring how Mamba can synergize with these methods will expand its utility and effectiveness in complex tasks across multiple domains.

Computation Efficiency: Enhancing the computational efficiency of Mamba models, especially for vision tasks, is a promising research direction. Developing hardware-aware algorithms tailored for visual Mamba models can reduce computational overheads, while maintaining or improving performance, making them more practical for real-world applications.

Author Contributions: Conceptualization, H.Z.; Writing—original draft preparation, H.Z., Y.Z., D.W. and Z.Y.; Visualization, L.Z. and T.C.; Writing—review and editing, Z.W. and Z.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Rosenblatt, F. *The Perceptron, a Perceiving and Recognizing Automaton Project Para*; Cornell Aeronautical Laboratory: Buffalo, NY, USA, 1957.
2. Rosenblatt, F.; Jones, B.; Smith, T.; Brown, C.; Green, M.; Wilson, A.; Taylor, J.; White, P.; King, R.; Johnson, L. *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*; Spartan Books: Washington, DC, USA, 1962; Volume 55.
3. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
4. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 84–90. [CrossRef]
5. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
6. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv* **2014**, arXiv:1409.0473.

7. Parikh, A.P.; Täckström, O.; Das, D.; Uszkoreit, J. A decomposable attention model for natural language inference. *arXiv* **2016**, arXiv:1606.01933.
8. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
9. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
10. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
11. Gu, A.; Dao, T. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv* **2023**, arXiv:2312.00752.
12. Lieber, O.; Lenz, B.; Bata, H.; Cohen, G.; Osin, J.; Dalmedigos, I.; Safahi, E.; Meirom, S.; Belinkov, Y.; Shalev-Shwartz, S.; et al. Jamba: A Hybrid Transformer-Mamba Language Model. *arXiv* **2024**, arXiv:2403.19887.
13. Pióro, M.; Ciebiera, K.; Król, K.; Ludziejewski, J.; Jaszczur, S. Moe-mamba: Efficient selective state space models with mixture of experts. *arXiv* **2024**, arXiv:2401.04081.
14. Anthony, Q.; Tokpanov, Y.; Glorioso, P.; Millidge, B. BlackMamba: Mixture of Experts for State-Space Models. *arXiv* **2024**, arXiv:2402.01771.
15. Fu, D.Y.; Dao, T.; Saab, K.K.; Thomas, A.W.; Rudra, A.; Ré, C. Hungry hungry hippos: Towards language modeling with state space models. *arXiv* **2022**, arXiv:2212.14052.
16. Hendrycks, D.; Gimpel, K. Gaussian error linear units (gelus). *arXiv* **2016**, arXiv:1606.08415.
17. Ramachandran, P.; Zoph, B.; Le, Q.V. Swish: A Self-Gated Activation Function. *arXiv* **2017**, arXiv:1710.05941.
18. Ba, J.L.; Kiros, J.R.; Hinton, G.E. Layer normalization. *arXiv* **2016**, arXiv:1607.06450.
19. Sun, Y.; Dong, L.; Huang, S.; Ma, S.; Xia, Y.; Xue, J.; Wang, J.; Wei, F. Retentive network: A Successor to Transformer for Large Language Models. *arXiv* **2023**, arXiv:2307.08621.
20. Katharopoulos, A.; Vyas, A.; Pappas, N.; Fleuret, F. Transformers are rnns: Fast autoregressive transformers with linear attention. In Proceedings of the International Conference on Machine Learning, PMLR, Virtual Event, 13–18 July 2020; pp. 5156–5165.
21. Poli, M.; Massaroli, S.; Nguyen, E.; Fu, D.Y.; Dao, T.; Baccus, S.; Bengio, Y.; Ermon, S.; Ré, C. Hyena hierarchy: Towards larger convolutional language models. In Proceedings of the International Conference on Machine Learning, PMLR, Honolulu, HI, USA, 23–29 July 2023; pp. 28043–28078.
22. Romero, D.W.; Kuzina, A.; Bekkers, E.J.; Tomczak, J.M.; Hoogendoorn, M. Ckconv: Continuous kernel convolution for sequential data. *arXiv* **2021**, arXiv:2102.02611.
23. Zhai, S.; Talbott, W.; Srivastava, N.; Huang, C.; Goh, H.; Zhang, R.; Susskind, J. An attention free transformer. *arXiv* **2021**, arXiv:2105.14103.
24. Peng, B.; Alcaide, E.; Anthony, Q.; Albalak, A.; Arcadinho, S.; Cao, H.; Cheng, X.; Chung, M.; Grella, M.; GV, K.K.; et al. Rwkv: Reinventing rnns for the transformer era. *arXiv* **2023**, arXiv:2305.13048.
25. Tallec, C.; Ollivier, Y. Can recurrent neural networks warp time? *arXiv* **2018**, arXiv:1804.11188.
26. Zhu, L.; Liao, B.; Zhang, Q.; Wang, X.; Liu, W.; Wang, X. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv* **2024**, arXiv:2401.09417.
27. Liu, Y.; Tian, Y.; Zhao, Y.; Yu, H.; Xie, L.; Wang, Y.; Ye, Q.; Liu, Y. Vmamba: Visual state space model. *arXiv* **2024**, arXiv:2401.10166.
28. Yang, C.; Chen, Z.; Espinosa, M.; Ericsson, L.; Wang, Z.; Liu, J.; Crowley, E.J. PlainMamba: Improving Non-Hierarchical Mamba in Visual Recognition. *arXiv* **2024**, arXiv:2403.17695.
29. Huang, T.; Pei, X.; You, S.; Wang, F.; Qian, C.; Xu, C. LocalMamba: Visual State Space Model with Windowed Selective Scan. *arXiv* **2024**, arXiv:2403.09338.
30. Pei, X.; Huang, T.; Xu, C. EfficientVMamba: Atrous Selective Scan for Light Weight Visual Mamba. *arXiv* **2024**, arXiv:2403.09977.
31. Hu, V.T.; Baumann, S.A.; Gui, M.; Grebenkova, O.; Ma, P.; Fischer, J.; Ommer, B. Zigma: Zigzag mamba diffusion model. *arXiv* **2024**, arXiv:2403.13802.
32. Shi, Y.; Xia, B.; Jin, X.; Wang, X.; Zhao, T.; Xia, X.; Xiao, X.; Yang, W. VmambaIR: Visual State Space Model for Image Restoration. *arXiv* **2024**, arXiv:2403.11423.
33. Li, K.; Li, X.; Wang, Y.; He, Y.; Wang, Y.; Wang, L.; Qiao, Y. Videomamba: State space model for efficient video understanding. *arXiv* **2024**, arXiv:2403.06977.
34. Zhang, Z.; Liu, A.; Reid, I.; Hartley, R.; Zhuang, B.; Tang, H. Motion mamba: Efficient and long sequence motion generation with hierarchical and bidirectional selective ssm. *arXiv* **2024**, arXiv:2403.07487.
35. Yang, Y.; Xing, Z.; Zhu, L. Vivim: A video vision mamba for medical video object segmentation. *arXiv* **2024**, arXiv:2401.14168.
36. Chen, K.; Chen, B.; Liu, C.; Li, W.; Zou, Z.; Shi, Z. Rsmamba: Remote sensing image classification with state space model. *arXiv* **2024**, arXiv:2403.19654.
37. Behrouz, A.; Santacatterina, M.; Zabih, R. MambaMixer: Efficient Selective State Space Models with Dual Token and Channel Selection. *arXiv* **2024**, arXiv:2403.19888.
38. Li, S.; Singh, H.; Grover, A. Mamba-ND: Selective State Space Modeling for Multi-Dimensional Data. *arXiv* **2024**, arXiv:2402.05892.
39. Patro, B.N.; Agneeswaran, V.S. SiMBA: Simplified Mamba-Based Architecture for Vision and Multivariate Time series. *arXiv* **2024**, arXiv:2403.15360.

40. Li, S.; Zhu, T.; Duan, F.; Chen, L.; Ning, H.; Wan, Y. HARMamba: Efficient Wearable Sensor Human Activity Recognition Based on Bidirectional Selective SSM. *arXiv* **2024**, arXiv:2403.20183.
41. Cheng, C.; Wang, H.; Sun, H. Activating Wider Areas in Image Super-Resolution. *arXiv* **2024**, arXiv:2403.08330.
42. Qiao, Y.; Yu, Z.; Guo, L.; Chen, S.; Zhao, Z.; Sun, M.; Wu, Q.; Liu, J. VL-Mamba: Exploring State Space Models for Multimodal Learning. *arXiv* **2024**, arXiv:2403.13600.
43. Chen, G.; Huang, Y.; Xu, J.; Pei, B.; Chen, Z.; Li, Z.; Wang, J.; Li, K.; Lu, T.; Wang, L. Video mamba suite: State space model as a versatile alternative for video understanding. *arXiv* **2024**, arXiv:2403.09626.
44. Liu, J.; Yu, R.; Wang, Y.; Zheng, Y.; Deng, T.; Ye, W.; Wang, H. Point mamba: A novel point cloud backbone based on state space model with octree-based ordering strategy. *arXiv* **2024**, arXiv:2403.06467.
45. Wang, J.; Chen, J.; Chen, D.; Wu, J. Large Window-based Mamba UNet for Medical Image Segmentation: Beyond Convolution and Self-attention. *arXiv* **2024**, arXiv:2403.07332.
46. Zhang, Y.; Yan, W.; Yan, K.; Lam, C.P.; Qiu, Y.; Zheng, P.; Tang, R.S.Y.; Cheng, S.S. Motion-Guided Dual-Camera Tracker for Low-Cost Skill Evaluation of Gastric Endoscopy. *arXiv* **2024**, arXiv:2403.05146.
47. Tang, Y.; Dong, P.; Tang, Z.; Chu, X.; Liang, J. VMRNN: Integrating Vision Mamba and LSTM for Efficient and Accurate Spatiotemporal Forecasting. *arXiv* **2024**, arXiv:2403.16536.
48. Chen, C.S.; Chen, G.Y.; Zhou, D.; Jiang, D.; Chen, D.S. Res-VMamba: Fine-Grained Food Category Visual Classification Using Selective State Space Models with Deep Residual Learning. *arXiv* **2024**, arXiv:2402.15761.
49. Wan, Z.; Wang, Y.; Yong, S.; Zhang, P.; Stepputtis, S.; Sycara, K.; Xie, Y. Sigma: Siamese Mamba Network for Multi-Modal Semantic Segmentation. *arXiv* **2024**, arXiv:2404.04256.
50. Yang, Y.; Ma, C.; Yao, J.; Zhong, Z.; Zhang, Y.; Wang, Y. ReMamber: Referring Image Segmentation with Mamba Twister. *arXiv* **2024**, arXiv:2403.17839.
51. Wang, Z.; Zheng, J.Q.; Zhang, Y.; Cui, G.; Li, L. Mamba-unet: Unet-like pure visual mamba for medical image segmentation. *arXiv* **2024**, arXiv:2402.05079.
52. Ma, C.; Wang, Z. Semi-Mamba-UNet: Pixel-Level Contrastive and Pixel-Level Cross-Supervised Visual Mamba-based UNet for Semi-Supervised Medical Image Segmentation. *arXiv* **2024**, arXiv:2402.07245.
53. Wang, Z.; Zheng, J.Q.; Ma, C.; Guo, T. VMambaMorph: A Visual Mamba-based Framework with Cross-Scan Module for Deformable 3D Image Registration. *arXiv* **2024**, arXiv:2404.05105.
54. Chen, H.; Song, J.; Han, C.; Xia, J.; Yokoya, N. ChangeMamba: Remote Sensing Change Detection with Spatio-Temporal State Space Model. *arXiv* **2024**, arXiv:2404.03425.
55. Wu, R.; Liu, Y.; Liang, P.; Chang, Q. H-vmunet: High-order Vision Mamba UNet for Medical Image Segmentation. *arXiv* **2024**, arXiv:2403.13642.
56. Huang, J.; Yang, L.; Wang, F.; Wu, Y.; Nan, Y.; Aviles-Rivero, A.I.; Schönlieb, C.B.; Zhang, D.; Yang, G. MambaMIR: An Arbitrary-Masked Mamba for Joint Medical Image Reconstruction and Uncertainty Estimation. *arXiv* **2024**, arXiv:2402.18451.
57. Guo, H.; Li, J.; Dai, T.; Ouyang, Z.; Ren, X.; Xia, S.T. MambaIR: A Simple Baseline for Image Restoration with State-Space Model. *arXiv* **2024**, arXiv:2402.15648.
58. Shahab Sepehri, M.; Fabian, Z.; Soltanolkotabi, M. Serpent: Scalable and Efficient Image Restoration via Multi-scale Structured State Space Models. *arXiv* **2024**, arXiv:2403.17902.
59. Sanjid, K.S.; Hossain, M.T.; Junayed, M.S.S.; Uddin, D.M.M. Integrating Mamba Sequence Model and Hierarchical Upsampling Network for Accurate Semantic Segmentation of Multiple Sclerosis Legion. *arXiv* **2024**, arXiv:2403.17432.
60. Tang, H.; Cheng, L.; Huang, G.; Tan, Z.; Lu, J.; Wu, K. Rotate to Scan: UNet-like Mamba with Triplet SSM Module for Medical Image Segmentation. *arXiv* **2024**, arXiv:2403.17701.
61. Liu, J.; Yang, H.; Zhou, H.Y.; Xi, Y.; Yu, L.; Yu, Y.; Liang, Y.; Shi, G.; Zhang, S.; Zheng, H.; et al. Swin-umamba: Mamba-based unet with imagenet-based pretraining. *arXiv* **2024**, arXiv:2402.03302.
62. Wu, R.; Liu, Y.; Liang, P.; Chang, Q. UltraLight VM-UNet: Parallel Vision Mamba Significantly Reduces Parameters for Skin Lesion Segmentation. *arXiv* **2024**, arXiv:2403.20035.
63. Ruan, J.; Xiang, S. Vm-unet: Vision mamba unet for medical image segmentation. *arXiv* **2024**, arXiv:2402.02491.
64. Zhang, M.; Yu, Y.; Gu, L.; Lin, T.; Tao, X. VM-UNET-V2 Rethinking Vision Mamba UNet for Medical Image Segmentation. *arXiv* **2024**, arXiv:2403.09157.
65. Yue, Y.; Li, Z. MedMamba: Vision Mamba for Medical Image Classification. *arXiv* **2024**, arXiv:2403.03849.
66. Chen, T.; Tan, Z.; Gong, T.; Chu, Q.; Wu, Y.; Liu, B.; Ye, J.; Yu, N. MiM-ISTD: Mamba-in-Mamba for Efficient Infrared Small Target Detection. *arXiv* **2024**, arXiv:2403.02148.
67. Ma, X.; Zhang, X.; Pun, M.O. RS3Mamba: Visual State Space Model for Remote Sensing Images Semantic Segmentation. *arXiv* **2024**, arXiv:2404.02457.
68. Zhen, Z.; Hu, Y.; Feng, Z. FreqMamba: Viewing Mamba from a Frequency Perspective for Image Deraining. *arXiv* **2024**, arXiv:2404.09476.
69. Zhao, S.; Chen, H.; Zhang, X.; Xiao, P.; Bai, L.; Ouyang, W. RS-Mamba for Large Remote Sensing Image Dense Prediction. *arXiv* **2024**, arXiv:2404.02668.
70. Gong, H.; Kang, L.; Wang, Y.; Wan, X.; Li, H. nnmamba: 3d biomedical image segmentation, classification and landmark detection with state space model. *arXiv* **2024**, arXiv:2402.03526.

71. Xing, Z.; Ye, T.; Yang, Y.; Liu, G.; Zhu, L. Segmamba: Long-range sequential modeling mamba for 3d medical image segmentation. *arXiv* **2024**, arXiv:2401.13560.
72. Yang, J.X.; Zhou, J.; Wang, J.; Tian, H.; Liew, A.W.C. Hsimamba: Hyperpsectral imaging efficient feature learning with bidirectional state space for classification. *arXiv* **2024**, arXiv:2404.00272.
73. He, X.; Cao, K.; Yan, K.; Li, R.; Xie, C.; Zhang, J.; Zhou, M. Pan-Mamba: Effective pan-sharpening with State Space Model. *arXiv* **2024**, arXiv:2402.12192.
74. Xu, Z.; Lin, Y.; Han, H.; Yang, S.; Li, R.; Zhang, Y.; Li, X. MambaTalk: Efficient Holistic Gesture Synthesis with Selective State Space Models. *arXiv* **2024**, arXiv:2403.09471.
75. Zhu, Q.; Cai, Y.; Fang, Y.; Yang, Y.; Chen, C.; Fan, L.; Nguyen, A. Samba: Semantic Segmentation of Remotely Sensed Images with State Space Model. *arXiv* **2024**, arXiv:2404.01705.
76. Wang, Z.; Ma, C. Weak-Mamba-UNet: Visual Mamba Makes CNN and ViT Work Better for Scribble-based Medical Image Segmentation. *arXiv* **2024**, arXiv:2402.10887.
77. Hao, J.; He, L.; Hung, K.F. T-Mamba: Frequency-Enhanced Gated Long-Range Dependency for Tooth 3D CBCT Segmentation. *arXiv* **2024**, arXiv:2404.01065.
78. Xie, J.; Liao, R.; Zhang, Z.; Yi, S.; Zhu, Y.; Luo, G. ProMamba: Prompt-Mamba for polyp segmentation. *arXiv* **2024**, arXiv:2403.13660.
79. Zubić, N.; Gehrig, M.; Scaramuzza, D. State Space Models for Event Cameras. *arXiv* **2024**, arXiv:2402.15584.
80. Islam, M.M.; Bertasius, G. Long movie clip classification with state-space video models. In Proceedings of the European Conference on Computer Vision. Springer, Glasgow, UK, 23–28 August 2022; pp. 87–104.
81. Zheng, Z.; Zhang, J. FD-Vision Mamba for Endoscopic Exposure Correction. *arXiv* **2024**, arXiv:2402.06378.
82. Yang, G.; Du, K.; Yang, Z.; Du, Y.; Zheng, Y.; Wang, S. CMViM: Contrastive Masked Vim Autoencoder for 3D Multi-modal Representation Learning for AD classification. *arXiv* **2024**, arXiv:2403.16520.
83. Li, Y.; Yang, W.; Fei, B. 3DMambaComplete: Exploring Structured State Space Model for Point Cloud Completion. *arXiv* **2024**, arXiv:2404.07106.
84. Ma, J.; Li, F.; Wang, B. U-mamba: Enhancing long-range dependency for biomedical image segmentation. *arXiv* **2024**, arXiv:2401.04722.
85. Zheng, Z.; Wu, C. U-shaped Vision Mamba for Single Image Dehazing. *arXiv* **2024**, arXiv:2402.04139.
86. Liao, W.; Zhu, Y.; Wang, X.; Pan, C.; Wang, Y.; Ma, L. Lightm-unet: Mamba assists in lightweight unet for medical image segmentation. *arXiv* **2024**, arXiv:2403.05246.
87. Guo, T.; Wang, Y.; Meng, C. Mambamorph: A mamba-based backbone with contrastive feature learning for deformable mr-ct registration. *arXiv* **2024**, arXiv:2401.13934.
88. Fei, Z.; Fan, M.; Yu, C.; Huang, J. Scalable Diffusion Models with State Space Backbone. *arXiv* **2024**, arXiv:2402.05608.
89. Oshima, Y.; Taniguchi, S.; Suzuki, M.; Matsuo, Y. SSM Meets Video Diffusion Models: Efficient Video Generation with Structured State Spaces. *arXiv* **2024**, arXiv:2403.07711.
90. Fu, L.; Li, X.; Cai, X.; Wang, Y.; Wang, X.; Shen, Y.; Yao, Y. MD-Dose: A Diffusion Model based on the Mamba for Radiotherapy Dose Prediction. *arXiv* **2024**, arXiv:2403.08479.
91. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 10–17 October 2021; pp. 10012–10022.
92. Shi, X.; Chen, Z.; Wang, H.; Yeung, D.Y.; Wong, W.K.; Woo, W.c. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 802–810.
93. Li, W.; Hong, X.; Fan, X. SpikeMba: Multi-Modal Spiking Saliency Mamba for Temporal Video Grounding. *arXiv* **2024**, arXiv:2404.01174.
94. Radosavovic, I.; Kosaraju, R.P.; Girshick, R.; He, K.; Dollár, P. Designing network design spaces. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10428–10436.
95. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
96. Zhang, P.; Dai, X.; Yang, J.; Xiao, B.; Yuan, L.; Zhang, L.; Gao, J. Multi-scale vision longformer: A new vision transformer for high-resolution image encoding. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 2998–3008.
97. Yang, J.; Li, C.; Zhang, P.; Dai, X.; Xiao, B.; Yuan, L.; Gao, J. Focal self-attention for local-global interactions in vision transformers. *arXiv* **2021**, arXiv:2107.00641.
98. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
99. He, T.; Zhang, Z.; Zhang, H.; Zhang, Z.; Xie, J.; Li, M. Bag of tricks for image classification with convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 558–567.
100. Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; He, K. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1492–1500.

101. Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; Jégou, H. Training data-efficient image transformers & distillation through attention. In Proceedings of the International conference on machine learning, PMLR, Virtual, 18–24 July 2021; pp. 10347–10357.
102. Chen, Z.; Duan, Y.; Wang, W.; He, J.; Lu, T.; Dai, J.; Qiao, Y. Vision transformer adapter for dense predictions. *arXiv* **2022**, arXiv:2205.08534.
103. Wang, W.; Xie, E.; Li, X.; Fan, D.P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 568–578.
104. Liu, Z.; Mao, H.; Wu, C.Y.; Feichtenhofer, C.; Darrell, T.; Xie, S. A convnet for the 2020s. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 11976–11986.
105. Touvron, H.; Cord, M.; Jégou, H. Deit iii: Revenge of the vit. In *Computer Vision—ECCV 2022: 17th European Conference, Tel Aviv, Israel, 23–27 October 2022, Proceedings, Part XXIV*; Springer: Cham, Switzerland, 2022; pp. 516–533.
106. Zhao, H.; Zhang, M.; Zhao, W.; Ding, P.; Huang, S.; Wang, D. Cobra: Extending Mamba to Multi-Modal Large Language Model for Efficient Inference. *arXiv* **2024**, arXiv:2403.14520.
107. Gao, H.; Dang, D. Aggregating Local and Global Features via Selective State Spaces Model for Efficient Image Deblurring. *arXiv* **2024**, arXiv:2403.20106.
108. Zhou, Q.; Yang, W.; Fei, B.; Xu, J.; Zhang, R.; Liu, K.; Luo, Y.; He, Y. 3DMambaIPF: A State Space Model for Iterative Point Cloud Filtering via Differentiable Rendering. *arXiv* **2024**, arXiv:2404.05522.
109. Zhang, T.; Li, X.; Yuan, H.; Ji, S.; Yan, S. Point Could Mamba: Point Cloud Learning via State Space Model. *arXiv* **2024**, arXiv:2403.00762.
110. Liang, D.; Zhou, X.; Wang, X.; Zhu, X.; Xu, W.; Zou, Z.; Ye, X.; Bai, X. PointMamba: A Simple State Space Model for Point Cloud Analysis. *arXiv* **2024**, arXiv:2402.10739.
111. Shen, Q.; Yi, X.; Wu, Z.; Zhou, P.; Zhang, H.; Yan, S.; Wang, X. Gamba: Marry Gaussian Splatting with Mamba for single view 3D reconstruction. *arXiv* **2024**, arXiv:2403.18795.
112. Seeram, E. *Digital Radiography: Physical Principles and Quality Control*; Springer: Singapore, 2019.
113. Lui, R.N.; Wong, S.H.; Sánchez-Luna, S.A.; Pellino, G.; Bollipo, S.; Wong, M.Y.; Chiu, P.W.; Sung, J.J. Overview of guidance for endoscopy during the coronavirus disease 2019 pandemic. *J. Gastroenterol. Hepatol.* **2020**, *35*, 749–759. [CrossRef]
114. Withers, P.J.; Bouman, C.; Carmignato, S.; Cnudde, V.; Grimaldi, D.; Hagen, C.K.; Maire, E.; Manley, M.; Du Plessis, A.; Stock, S.R. X-ray computed tomography. *Nat. Rev. Methods Prim.* **2021**, *1*, 18. [CrossRef]
115. Christensen-Jeffries, K.; Couture, O.; Dayton, P.A.; Eldar, Y.C.; Hynynen, K.; Kiessling, F.; O'Reilly, M.; Pinton, G.F.; Schmitz, G.; Tang, M.X.; et al. Super-resolution ultrasound imaging. *Ultrasound Med. Biol.* **2020**, *46*, 865–891. [CrossRef]
116. Tiwari, A.; Srivastava, S.; Pant, M. Brain tumor segmentation and classification from magnetic resonance images: Review of selected methods from 2014 to 2019. *Pattern Recognit. Lett.* **2020**, *131*, 244–260. [CrossRef]
117. Ye, Z.; Chen, T. P-Mamba: Marrying Perona Malik Diffusion with Mamba for Efficient Pediatric Echocardiographic Left Ventricular Segmentation. *arXiv* **2024**, arXiv:2402.08506.
118. Yang, S.; Wang, Y.; Chen, H. MambaMIL: Enhancing Long Sequence Modeling with Sequence Reordering in Computational Pathology. *arXiv* **2024**, arXiv:2403.06800.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.