TRAVELING WAVES INTEGRATE SPATIAL INFORMATION INTO SPECTRAL REPRESENTATIONS

Mozes Jacobs[1] **Roberto C. Budzinski**[2] **Lyle Muller**[2] **Demba Ba**[1] **T. Anderson Keller**[1]
[1]The Kempner Institute for the Study of Natural and Artificial Intelligence, Harvard University
[2]Western University, Department of Mathematics, London, Ontario, Canada

ABSTRACT

Traveling waves are widely observed in the brain, but their precise computational function remains unclear. One prominent hypothesis is that they enable the transfer and integration of spatial information across neural populations. However, few computational models have explored how traveling waves might be harnessed to perform such integrative processing. Drawing inspiration from the famous "*Can one hear the shape of a drum?*" problem – which highlights how spectral modes encode geometric information – we introduce a set of convolutional recurrent neural networks that learn to produce traveling waves in their hidden states in response to visual stimuli. By applying a spectral decomposition to these wave-like activations, we obtain a powerful new representational space that outperforms equivalently local feed-forward networks on tasks requiring global spatial context. In particular, we observe that traveling waves effectively expand the receptive field of locally connected neurons, supporting long-range encoding and communication of information. We demonstrate that models equipped with this mechanism and spectral readouts solve visual semantic segmentation tasks demanding global integration, where local feed-forward models fail. As a first step toward traveling-wave-based representations in artificial networks, our findings suggest potential efficiency benefits and offer a new framework for connecting to biological recordings of neural activity.

1 INTRODUCTION

The propagation of traveling waves of neural activity has been measured on the surface of the brain from the earliest neural recordings (Adrian & Matthews, 1934). Driven by these observations, many theoretical arguments have been put forth to explain the functional roles of these dynamics. A non-exhaustive list of proposed functions includes that they are relevant to predictive coding (Alamia & VanRullen, 2019), the representation of symmetries (Keller et al., 2024), the consolidation of long term memories (Muller et al., 2018), or the encoding or motion (Heitmann & Ermentrout, 2020).

Most relevant to this study, one hypothesized role is that traveling waves serve as a mechanism for integration and transfer of information over long distances. For example, Besserve et al. (2015) use direction-specific causal information transfer metrics to demonstrate that traveling waves in the gamma frequency band are correlated with information transfer between different cortical regions; while Bhattacharya et al. (2022) show that waves change direction during information retrieval and processing. Despite these promising observations however, it remains challenging to investigate these

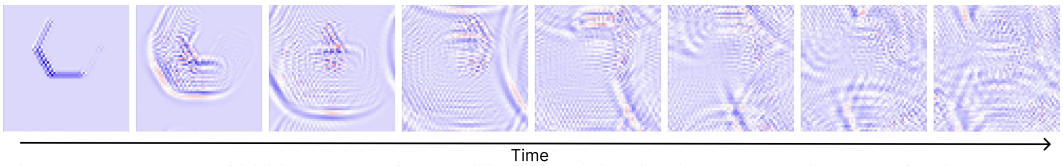

Figure 1: Sequence of hidden states of an oscillator model trained to segment images of polygons. We see that the shape of the hexagon is visible throughout the wave dynamics, and that waves propagate differently within the polygon due to reflections induced by the differing natural frequencies. Full videos at: `https://github.com/anonymous123-user/Wave_Representations`

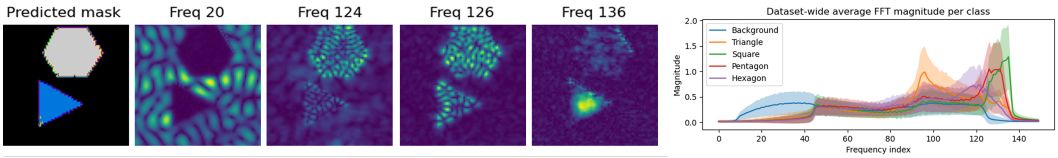

Figure 2: (Left) Plot of predicted semantic segmentation and select set of frequency bins for each pixel of a give test image. (Right) The full frequency spectrum for each shape in the dataset, averaged over all pixels containing that class label in the dataset. We see that different shapes have qualitatively different frequency curves, allowing for $> 99\%$ pixel-wise classification accuracy on a test set.

ideas computationally due to a lack of artificial neural network models which exhibit traveling wave dynamics. In modern artificial neural networks, information is transmitted over spatial distances or between tokens either via extremely deep convolutional neural networks, bottleneck/pooling layers, or all-to-all connectivity as-in Transformers (Vaswani et al., 2023). Each of these approaches comes with its own computational complexity and expressivity limitations, and it is therefore of great interest to explore alternative methods to integrate disparate information in neural systems.

In this paper, we aim to make progress towards understanding the causal role of wave dynamics in the transfer of information by filling this modeling gap, and improving our understanding how information may be represented by traveling waves in neural systems. Specifically, in this paper, we aim to shed light on the answer to two primary questions: I) How precisely can global information be communicated and integrated between neurons in locally constrained architectures? II) In what format is this transmitted information represented such that it can be read out for task-relevant processing?

To answer these questions, in the following, we introduce a set of convolutional recurrent neural network (conv-RNN) architectures that have inherently limited receptive field sizes in the initial forward pass, but are required to solve a global-information processing (semantic segmentation) task through recurrent processing over time. We show that such models learn to leverage wave dynamics to integrate information over space (Figure 1), thereby solving global tasks while each neuron only has access to local information at each timestep. Specifically, we leverage a spectral decomposition of each neuron's recurrent neural activity as our primary neural representation during training (Figure 2), and show that this is a viable method for extracting global information integrated through traveling waves, while individual static snapshots of neural activity are insufficient.

2 Motivation

To build intuition for how traveling waves may integrate information over space, we take inspiration from the famous mathematical question '*Can one hear the shape of a drum?*' posed by Mark Kac (1966). Simply put, this question asks whether the boundary conditions of an idealized drum head are uniquely identified by the frequencies at which the drum head will vibrate. Intuitively, it is straightforward to understand that when one strikes a drum head, this initial disturbance will propagate outwards as a traveling wave until it reaches the fixed boundary conditions where it will reflect with a phase shift. This reflected wave will thus have *collected information* about the boundary, and serves to bring it back towards the center. As the reflected wave returns towards the center, it will eventually collide with other reflected waves from the other edges of the shape, and combine in a superposition of wavefronts, such that eventually the displacement at each point of the drum will inevitably be a superposition of all the traveling wave components which carry information about the distant edges of the shape to the interior. Another way to think about Mark Kac's original question then, which is more directly related to the neural systems we study in this work, is how much information can one actually gather about the global environment from the sequence of vibrations at any single point on the 'drum's surface'? At the time of Mark Kac's question, it was known that the area of a drum-head could be deduced from its eigenspectrum uniquely; and indeed, it took more than 25 years for researchers to find counter examples of drum heads that cannot be distinguished by their eigenspectra (Gordon et al., 1992) – implying that the amount of unique geometric information in spectral representations is significant. In the following set of experiments, we will use this intuition as a starting point to begin building models which propagate and combine information over space through traveling waves, proposing to read this information out through a spectral decomposition of the hidden state activity, much like 'hearing the shape' of visual objects.

3 METHODS

In all experiments, the task is semantic segmentation, where each pixel of the original image must be classified as either background, or one of the classes from the dataset, and models are trained to minimize a pixel-wise cross-entropy loss. Crucially, locally restricted models all make use of shallow convolutional encoders (with 3×3 kernels), convolutional recurrent connections (if recurrent), and pixel-local decoders, ensuring that the spatial receptive field of each neuron in a single feed-forward pass is limited to be significancy less than the inherent length scale of features necessary to identify class labels in each dataset. The full code and video visualizations for the results in this paper are available at: `https://github.com/anonymous123-user/Wave_Representations`

3.1 DATASETS

To validate the core idea that traveling waves can be used to transmit information over large spatial distances, and that information can be decoded via a spectral decomposition, we use three datasets:

Polygons First, we consider a simple dataset composed of polygons on black backgrounds, where the classes are given by the number of sides of the polygons. The examples are synthetic 75×75 pixel grayscale images with 1 to 2 polygons, each with 3 to 6 edges (yielding 5 total classes with the background) roughly circumscribed within circles with radii of 15 to 20 pixels. On this dataset, the angle of the corners of the shape are sufficient to correctly classify those patches, but this information must then be transferred to the center of the shape for correct semantic segmentation of the interior.

Tetrominoes As a second dataset with slightly increased complexity that has been employed in prior segmentation work (Miyato et al., 2024), we employ own re-implementation of the Tetrominoes dataset (Kabra et al., 2019), composed of 'Tetris' like blocks of varying shapes and colors arranged on a black background. This dataset increases the difficulty due to the increased number of objects per image, and the increased complexity of the shapes themselves. Specifically, this dataset contains 6 distinct classes and the objects are 14-28 pixels long. Each image can have 1 to 5 shapes.

MNIST Finally, we use the MNIST dataset (LeCun, 1998) but increase the spatial dimensions to 56×56 through interpolation. The pixels are binarized at a threshold of 0.5, and are assigned the associated class label of the digit in the image or 'background'. This task is significantly harder than those above since the shapes now differ between instances (i.e. each hand-wirtten 3 is unique) and thus the model must learn these sets of invariances when processing the spectral representations.

3.2 MODELS

For an input image $\mathbf{x} \in \mathbb{R}^{C \times H \times W}$, the following networks output $\mathbf{y} \in \mathbb{R}^{N \times H \times W}$, a set of N class logits for each pixel, which are then passed through a softmax to compute the predicted class labels.

Local Feed-Forward CNN Baseline The CNN baselines are composed of a number (L) of convolutional layers, each with (3×3) convolutional kernels, and 16 channels. Again, this baseline is intended to demonstrate the inability of neurons with restricted local receptive fields to perform tasks which require global information. Explicitly, with $\mathbf{w}^l$ denoting layer l's convolutional kernel:

$$\mathbf{y} = \sigma(\mathbf{w}^L \star \sigma(\mathbf{w}^{L-1} \star \dots \sigma(\mathbf{w}^1 \star \mathbf{x} + \mathbf{b}^1) \dots + \mathbf{b}^{L-1}) + \mathbf{b}^L) \tag{1}$$

Locally Coupled Oscillatory RNN As a model which most closely follows the theory of 'hearing the shape of a drum', we implement a recurrent neural network parameterized as a network of locally coupled oscillators. In prior work, this model has been referred to as the Neural Wave Machine (Keller & Welling, 2023), based on the coRNN (Rusch & Mishra, 2021), and is known to be biased towards the production of traveling waves. One difference in this work is that we make the natural frequencies of each oscillator γ, and the damping term α, a function of the input image, computed through a shallow 3-layer CNN model. This crucially allows the way in which waves propagate over the hidden state (and therefore the frequencies) to be dictated by the input image. The initial state of the model is also set by the shallow 4-layer CNN encoder $f_\theta(\mathbf{x})$, with (3×3) kernels. Explicitly:

$$\mathbf{h}_0 = f_\theta(\mathbf{x}), \qquad \ddot{\mathbf{h}} = \sigma\left(\mathbf{w}_h \star \mathbf{h}\right) - \gamma_\theta(\mathbf{x})\mathbf{h} - \alpha_\theta(\mathbf{x})\dot{\mathbf{h}} \tag{2}$$

We initialize the recurrent convolutional kernel $\mathbf{w}_h$ to the finite difference approximation of the Laplacian operator to bias towards wave propagation. Finally, we numerically integrate the second order ODE above using euler integration with a timestep size of 0.1 for a fixed amount of time.

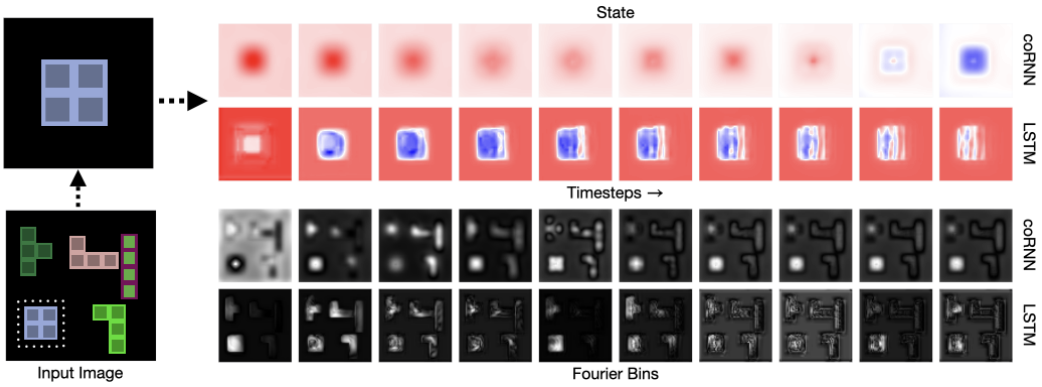

Figure 3: Visualization of a subset (outlined in dashed white line) of the LSTM and coRNN hidden state evolution after training (top) for a given image (left). We see waves propagate over the timesteps throughout the shape. On the bottom we plot the associated pixel-wise Fourier transform of the hidden state dynamics and observe individual shapes appear to pop out in different frequency bins.

Convolutional LSTM We implement a convolutional variant of the LSTM to see how a recurrent network without a traveling wave inductive bias might solve the same tasks. In this model, all otherwise dense connections which are used to update the cell state and compute gate activations are now replaced with local convolutions over the spatial dimensions of the hidden state. Explicitly:

$$\mathbf{x}_0 = f_\theta(\mathbf{x}), \quad \mathbf{i}_t = \sigma\big(\mathbf{w}_i \star \mathbf{x}_{t-1} + \mathbf{b}_i\big), \quad \mathbf{f}_t = \sigma\big(\mathbf{w}_f \star \mathbf{x}_{t-1} + \mathbf{b}_f\big), \quad \tilde{\mathbf{C}}_t = \tanh\big(\mathbf{w}_c \star \mathbf{x}_{t-1} + \mathbf{b}_c\big), \tag{3}$$

$$\mathbf{C}_t = \mathbf{f}_t \odot \mathbf{C}_{t-1} \; + \; \mathbf{i}_t \odot \tilde{\mathbf{C}}_t, \quad \mathbf{o}_t = \sigma\big(\mathbf{w}_o \star \mathbf{x}_{t-1} + \mathbf{b}_o\big), \quad \mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{C}_t), \quad \mathbf{x}_t = \mathbf{w}_{o2} \star \sigma(\mathbf{w}_{o1} \star \mathbf{h}_t). \tag{4}$$

As in the coRNN, we use two hidden channels, a 4-layer CNN for f_θ, and two 3-layer CNNs to initialize $\mathbf{h}_0$ and $\mathbf{C}_0$. We do no special initialization, yet as we show in Figure 3, these models still learn to produce waves in their hidden states to solve the task.

Recurrent Readout For the recurrent models, we must decide how to read out the class label from the sequence of hidden states. The simplest option is to feed the hidden state at the final timestep to a pixel-wise 'readout' network for classification (*Last* in Table 1). Alternatively, we can feed a function of the hidden states over time into the readout network. The options for such a function include taking the maximum or mean hidden state values over time (*Max* and *Mean* respectively) and computing the Fourier coefficient amplitudes of the time series (*FFT*). In all cases, we parameterize the readout module as a 4-layer MLP.

Non-local U-Net Baseline As a competitive non-local effective upper bound on the local models' performance, we implement a simple U-Net model (Ronneberger et al., 2015) to perform classification. This model contains 4 layers and up to 1024 channels in the bottleneck. Crucially, because of the spatial bottleneck, the receptive fields of pixels in the output layer of this model cover the entire image, allowing for simple solutions to the semantic segmentation task.

4 Results

Polygons As an initial proof of concept, in Figure 1 we show the hidden state evolution for the Locally Coupled Oscillatory RNN model on a single example of the polygons dataset. We see that the model initializes the hidden state such that waves are propagated from the edges of the object in all directions. From naive inspection of sequence, we can see that waves appear to propagate differently within the object, seemingly changing the spectral representation of each point on the interior of the hexagon. In Figure 2, we plot the spectral representation of a given test image, and the frequency representations for each object class averaged over all pixels in the validation set which are labeled as that class. We see that there is a clear distinction between the shapes that the model appears to pick up on which allows for the model to identify all polygons with $> 99\%$ accuracy on a held out test set.

Tetrominoes & MNIST As feed-forward models with different receptive field sizes and local recurrent models, in Table 1 we include the aggregated results of 300 total models. We observe that CNNs with 2 and 4 layers perform poorly on both datasets, with performance improving as the receptive field (RF) increases, as expected. This improvement is most pronounced in 16-layer

		MNIST			Tetrominoes		
Model	Arch.	FG-Acc	FG-IoU	Loss	FG-Acc	FG-IoU	Loss
CNN	2	0.14 ± 0.06	0.10 ± 0.04	0.34 ± 0.13	0.24 ± 0.08	0.14 ± 0.05	0.27 ± 0.13
	4	0.19 ± 0.11	0.13 ± 0.07	0.35 ± 0.18	0.31 ± 0.16	0.20 ± 0.11	0.28 ± 0.19
	8	0.42 ± 0.15	0.30 ± 0.11	0.25 ± 0.16	$\mathbf{0.74 \pm 0.26}$	$\mathbf{0.64 \pm 0.22}$	$\mathbf{0.11 \pm 0.19}$
	16	$\mathbf{0.57 \pm 0.39}$	$\mathbf{0.50 \pm 0.35}$	$\mathbf{0.27 \pm 0.29}$	0.40 ± 0.51	0.40 ± 0.51	0.39 ± 0.33
	32	0.27 ± 0.43	0.25 ± 0.41	0.50 ± 0.31	0.33 ± 0.47	0.32 ± 0.47	0.41 ± 0.31
LSTM	Max	0.42 ± 0.05	0.31 ± 0.04	0.21 ± 0.02	0.62 ± 0.30	0.52 ± 0.31	0.16 ± 0.18
	Mean	0.42 ± 0.09	0.31 ± 0.07	0.21 ± 0.03	0.64 ± 0.25	0.53 ± 0.23	0.15 ± 0.18
	Last	0.32 ± 0.23	0.24 ± 0.18	0.35 ± 0.24	0.59 ± 0.41	0.52 ± 0.38	0.24 ± 0.28
	FFT	$\mathbf{0.73 \pm 0.27}$	$\mathbf{0.66 \pm 0.25}$	$\mathbf{0.14 \pm 0.20}$	$\mathbf{0.95 \pm 0.04}$	$\mathbf{0.91 \pm 0.07}$	$\mathbf{0.03 \pm 0.02}$
coRNN	Max	0.48 ± 0.07	0.37 ± 0.07	0.19 ± 0.02	0.88 ± 0.11	0.81 ± 0.15	0.05 ± 0.03
	Mean	0.46 ± 0.08	0.35 ± 0.07	0.20 ± 0.03	0.92 ± 0.07	0.87 ± 0.11	0.04 ± 0.02
	Last	0.50 ± 0.08	0.39 ± 0.07	0.18 ± 0.03	0.94 ± 0.06	0.90 ± 0.09	0.03 ± 0.02
	FFT	$\mathbf{0.76 \pm 0.05}$	$\mathbf{0.65 \pm 0.07}$	$\mathbf{0.10 \pm 0.02}$	$\mathbf{0.98 \pm 0.01}$	$\mathbf{0.97 \pm 0.01}$	$\mathbf{0.01 \pm 0.00}$
U-Net		0.99 ± 0.00	0.99 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00

Table 1: **Locally constrained recurrent models with timeseries readouts can semantically segment images at the pixel level, a task requiring global information.** Models with the lowest foreground loss are in bold. Arch (architecture) refers to number of CNN layers, and type of readout for LSTM and coRNN. Results are from 10 random seeds, displayed as $mean \pm std$.

CNNs on MNIST and in 8-layer CNNs on Tetrominoes (effective RF sizes of 33 and 17, respectively). However, mean performance declines with 32 layers on MNIST and with 16 and 32 layers on Tetrominoes, though the variance remains high. As deeper networks contain more parameters, they pose a more challenging optimization problem, leading to inconsistent convergence. Neverthless, peak performance continues to improve as the number of layers increases, even in the 16- and 32-layer models, despite more failed training runs. As expected, U-Net exhibits the best performance.

Among recurrent models, those with the FFT perform best, with the coRNN outperforming baselines. Recurrent models that rely solely on the last hidden state for predictions achieve the weakest results. We note that it should also be possible to learn an arbitrary linear map over time as another powerful readout mechanism in-place of the FFT, but we leave this to future work. The coRNN models exhibit the lowest variance, indicating greater training stability. Figure 3 visualizes the recurrent states and fourier bins for a sample image. Interestingly, the LSTM learns to generate wave dynamics despite lacking an explicit inductive bias for doing so.

5 Discussion

In the above, we have presented arguments both theoretical and empirical supporting the idea that traveling waves may serve to integrate spatial information through the time dimension in otherwise locally constrained architectures, achieving performance comparable with globally connected counterparts. Furthermore, we have demonstrated empirically that this wave-encoded information is most directly accessible through spectral decompositions of the hidden state, while it is generally less accessible through standard readout methods using the final hidden state of RNNs. Finally, we showed that even if models are not biased towards wave dynamics initially, such as the Conv-LSTM, they will still learn to propagate waves in order to transfer information effectively through space, thereby implicating waves and wave-based representations as an optimal solution. We hope that this work draws increased attention to the idea that wave-based and spectral representations may carry global task-relevant information in both biological and artificial systems, thereby encouraging their increased study.

In future work, we believe that this research direction could provide a pathway to an alternative method for computing representational alignment between biological and artificial neural networks. By incorporating local recurrence and spectral decompositions, the representations of these models naturally exist in both the spatial and frequency domains for a given input, thereby making comparison with recording methodologies such as EEG and MEG very natural. Additionally, we speculate that this work may have implications for how something like all-to-all attention in Transformers (one of the key bottlenecks to scaling) may be alleviated in a biologically plausible manner. More precisely, if attention could be framed as wave-propagation and read out in frequency space, it may be possible to achieve similar performance with significantly reduced computation cost.

REFERENCES

E. D. Adrian and B. H. C. Matthews. The interpretation of potential waves in the cortex. *The Journal of Physiology*, 81(4):440–471, 1934. doi: https://doi.org/10.1113/jphysiol.1934.sp003147. URL https://physoc.onlinelibrary.wiley.com/doi/abs/10.1113/jphysiol.1934.sp003147.

Andrea Alamia and Rufin VanRullen. Alpha oscillations and traveling waves: Signatures of predictive coding? *PLOS Biology*, 17(10):1–26, 10 2019. doi: 10.1371/journal.pbio.3000487. URL https://doi.org/10.1371/journal.pbio.3000487.

Michel Besserve, Scott C. Lowe, Nikos K. Logothetis, Bernhard Schölkopf, and Stefano Panzeri. Shifts of gamma phase across primary visual cortical sites reflect dynamic stimulus-modulated information transfer. *PLOS Biology*, 13(9):1–29, 09 2015. doi: 10.1371/journal.pbio.1002257. URL https://doi.org/10.1371/journal.pbio.1002257.

Sayak Bhattacharya, Scott L. Brincat, Mikael Lundqvist, and Earl K. Miller. Traveling waves in the prefrontal cortex during working memory. *PLOS Computational Biology*, 18(1):1–22, 01 2022. doi: 10.1371/journal.pcbi.1009827. URL https://doi.org/10.1371/journal.pcbi.1009827.

Anand Gopalakrishnan, Aleksandar Stanić, Jürgen Schmidhuber, and Michael Curtis Mozer. Recurrent Complex-Weighted Autoencoders for Unsupervised Object Discovery, October 2024. URL http://arxiv.org/abs/2405.17283. arXiv:2405.17283.

Carolyn Gordon, David L. Webb, and Scott Wolpert. One cannot hear the shape of a drum. *Bulletin of the American Mathematical Society*, 27:134–138, 1992.

Stewart Heitmann and G. Bard Ermentrout. Direction-selective motion discrimination by traveling waves in visual cortex. *PLOS Computational Biology*, 16(9):1–20, 09 2020. doi: 10.1371/journal.pcbi.1008164. URL https://doi.org/10.1371/journal.pcbi.1008164.

Rishabh Kabra, Chris Burgess, Loic Matthey, Raphael Lopez Kaufman, Klaus Greff, Malcolm Reynolds, and Alexander Lerchner. Multi-object datasets. https://github.com/deepmind/multi-object-datasets/, 2019.

Mark Kac. Can One Hear the Shape of a Drum? *The American Mathematical Monthly*, 73(4):1–23, 1966. ISSN 0002-9890. doi: 10.2307/2313748. URL https://www.jstor.org/stable/2313748. Publisher: [Taylor & Francis, Ltd., Mathematical Association of America].

T. Anderson Keller and Max Welling. Neural wave machines: Learning spatiotemporally structured representations with locally coupled oscillatory recurrent neural networks. In *Proceedings of the 40th International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR, 2023.

T. Anderson Keller, Lyle Muller, Terrence J. Sejnowski, and Max Welling. A spacetime perspective on dynamical computation in neural information processing systems, 2024. URL https://arxiv.org/abs/2409.13669.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, January 2017. URL http://arxiv.org/abs/1412.6980. arXiv:1412.6980 [cs].

Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

Luisa H. B. Liboni, Roberto C. Budzinski, Alexandra N. Busch, Sindy Löwe, Thomas A. Keller, Max Welling, and Lyle E. Muller. Image segmentation with traveling waves in an exactly solvable recurrent neural network, November 2023. URL http://arxiv.org/abs/2311.16943. arXiv:2311.16943.

Sindy Löwe, Phillip Lippe, Maja Rudolph, and Max Welling. Complex-Valued Autoencoders for Object Discovery, November 2022. URL http://arxiv.org/abs/2204.02075. arXiv:2204.02075.

Sindy Löwe, Phillip Lippe, Francesco Locatello, and Max Welling. Rotating features for object discovery. *Advances in Neural Information Processing Systems*, 36, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/bb36593e5e438aac5dd07907e757e087-Abstract-Conference.html.

Takeru Miyato, Sindy Löwe, Andreas Geiger, and Max Welling. Artificial Kuramoto Oscillatory Neurons, October 2024. URL http://arxiv.org/abs/2410.13821. arXiv:2410.13821.

Lyle Muller, Frédéric Chavane, John Reynolds, and Terrence J. Sejnowski. Cortical travelling waves: mechanisms and computational principles. *Nature Reviews Neuroscience*, 19(5):255–268, 2018. doi: 10.1038/nrn.2018.20. URL https://doi.org/10.1038/nrn.2018.20.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation, May 2015. URL http://arxiv.org/abs/1505.04597. arXiv:1505.04597 [cs].

T. Konstantin Rusch and Siddhartha Mishra. Coupled oscillatory recurrent neural network (cornn): An accurate and (gradient) stable architecture for learning long time dependencies. In *International Conference on Learning Representations*, 2021.

Aleksandar Stanić, Anand Gopalakrishnan, Kazuki Irie, and Jürgen Schmidhuber. Contrastive training of complex-valued autoencoders for object discovery. *Advances in Neural Information Processing Systems*, 36, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/2439ec22091b9d6cfbebf3284b40116e-Abstract-Conference.html.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. URL https://arxiv.org/abs/1706.03762.

A Appendix

A.1 Related Work

Our work shares a strong connection with synchrony-based object-centric learning methods. One of the pioneering efforts in this area was the development of the complex autoencoder (Löwe et al., 2022). Since then, several advancements have been made, such as rotating features (Löwe et al., 2024), contrastive training (Stanić et al., 2024), and the recurrent complex autoencoder (Gopalakrishnan et al., 2024). The rotating features approach expanded the capabilities to multiple feature dimensions, while the latter two approaches marked significant progress in tackling more intricate datasets.

Recently, the Artificial Kuramoto Oscillatory Neurons (AKOrN) architecture was introduced, leveraging a network structure based on multi-dimensional Kuramoto oscillators to perform image segmentation and puzzle-solving (Miyato et al., 2024). Unlike our method, the waves in the AKOrN model are not used directly as a representation themselves, but instead are neglected through the use of the 'last hidden state' readout method that we discuss in this paper.

In another relevant development, a complex-valued recurrent neural network designed to generate traveling waves was proposed for image segmentation, with object binding information encoded in the phase of these waves (Liboni et al., 2023).

While our newly proposed mechanism shares ties with the "binding by synchrony" concept, it has a crucial distinction. Our model's object-centricity does not rely on precise zero-lag synchrony—where oscillators within an object are perfectly in phase—but rather on traveling waves of activity that can be interpreted as phase-lag synchrony. The "binding operation" then involves a transformation of the time signal into frequency space, accessible by examining the Fourier coefficients. We believe this connection is vital, as it enables us to leverage the extensive research in early neuroscience on "binding by synchrony" while simultaneously forming novel predictions on how such phenomena might manifest in natural neural systems.

A.2 Experimental Details

This section provides details on the training and evaluation procedures for the models presented in this paper. The full code for reproducing results and visualizations from the main text is available at: `https://github.com/anonymous123-user/Wave_Representations`.

For dataset partitioning, we use 51,000 images for training, 9,000 for validation, and 10,000 for testing in MNIST. The Tetrominoes dataset consists of 10,000 images for training, 1,000 for validation, and 1,000 for testing.

In Table 1, we report IOU and pixel-wise accuracy exclusively for the foreground classes. The loss, measured as cross-entropy loss, is calculated for all the classes.

A.2.1 Recurrent and CNN Models

Each model is trained for 300 epochs on the MNIST and Tetrominoes datasets. We evaluate the validation loss at the end of every epoch and retain the model with the lowest validation loss throughout training. The training process employs the Adam optimizer (Kingma & Ba, 2017) with a learning rate of 0.001 and a batch size of 64.

Each model is trained using multiple random seeds. For MNIST and Tetrominoes, we train each model using 10 different random seeds. For example, the coRNN with a FFT readout is trained on MNIST 10 times, each with a different random seed. After training, we evaluate each model individually and present the aggregated results, including the mean and standard deviation, in Table 1. In total, we train 150 models on MNIST and 150 models on Tetrominoes, leading to a total of 300 models.

We train the Conv-LSTMs for 20 timesteps. The coRNN model is trained for 100 timesteps on the MNIST and Tetrominoes datasets, while for the polygons dataset, the coRNN runs for 500 timesteps. In the FFT readout, we use the real component of the discrete Fourier transform. This results in 50 bins for the coRNN on MNIST and Tetrominoes, and 250 bins for the polygons dataset. The LSTM model outputs 10 Fourier bins for MNIST and Tetrominoes. Additionally, all recurrent convolutions in this paper are performed with circular padding to avoid boundary effects.

The Conv-LSTMs and coRNNs use nearly identical CNN encoders. Both use the same 4-layer CNN to process the input x (to initialize the input x_0 for the LSTM and to initialize the hidden state h_0 for the coRNN). The LSTM uses two 3-layer CNNs to intialize the hidden and cell states. The coRNN uses two identical (to the LSTM) 3-layer CNNs to initialize the natural frequencies γ and damping term α, except the coRNN 3-layer CNN's include an extra $ReLU$ function at the end of the CNN to support waves.

All convolutional models are trained with 16 channels. To ensure a fair comparison with the coRNN, a linear layer operating channelwise outputs 100 channels. The readout MLP used for MNIST and Tetrominoes consists of four layers. Its input size is given by the number of Fourier bins multiplied by 2, followed by two hidden layers of 256 neurons each, with ReLU activation between layers, and a final output layer producing logits for classification.

A.2.2 U-Net

We train each U-Net model on the MNIST and Tetrominoes datasets for a minimum of 25 epochs and a maximum of 50 epochs. After completing 25 epochs, we utilize a stopping criterion: if the model does not achieve a validation loss within 0.001 of its best value in any 10-epoch window, training is halted and the model with the best validation loss is saved. This approach allows training to potentially conclude as early as 25 epochs. We employ a learning rate of 0.001 using the Adam optimizer (Kingma & Ba, 2017). We train each model three times, using a different random seed for each iteration. After training, we evaluate each model individually and present the aggregated results, including the mean and standard deviation, in Table 1. We use a batch size of 64. We utilize this alternative training criteria because, as shown in Table 1, the U-Net easily solves MNIST and Tetrominoes and does not need to train as long, and the goal is to provide an upper bound. The U-Net uses 1024 channels