LEARNING FREQUENCY DOMAIN CODES FOR SEMANTIC VISION

Anonymous authors
Paper under double-blind review

ABSTRACT

Visually semantic concepts such as objects and categories provide a natural foundation for structured reasoning, yet models like convolutional neural networks (CNNs) and transformers routinely extract and aggregate features using homogeneous stacks of spatial layers. These entangle feature extraction and reasoning, rendering decision-making processes opaque and difficult to interpret. Psychovisual processing provides a way to mimic how the brain encodes and interprets visual information that produces higher abstractions from low-level processing. In this paper, we propose Semantic Visual Coding (SVC), a learnt frequency domain representation that introduces explicit psychovisual abstraction into CNNs. Inspired by psychovisual codes from the 1990s, SVC learns band-limited filters that encode task-relevant semantics as distinct regions of the discrete Fourier Transform (DFT). These converge towards sparse (data-driven) coronal patterns, suggesting a natural representation scheme for high-level features. We also introduce PsychoNet, a framework that adapts CNNs to make them psychovisually aware by combining traditional low-level feature extraction with frequency domain abstraction and reasoning via SVC. Salience analyses show that PsychoNet's spatial layers extract highly interpretable object parts and morphological features, unlike blob-like regions produced by standard CNNs. Through tracing gradient flow, we find SVC likely leverages these parts to form abstract representations of semantic features of image categories, highlighting frequency domain abstraction as a compelling direction for interpretable model reasoning and semantic-based decision making.

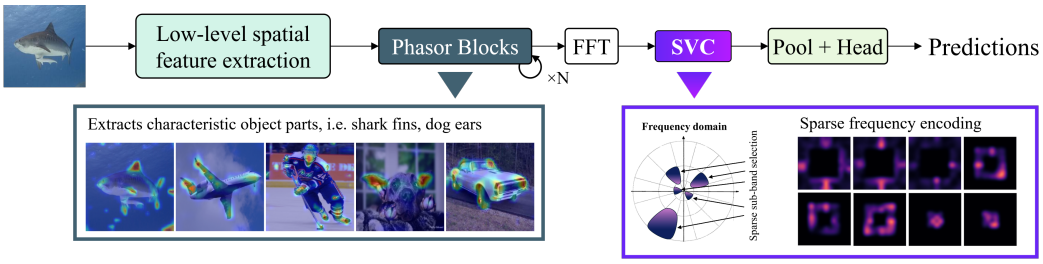

Figure 1: The brain encodes and interprets visual information using **psychovisual processing**, which separates feature extraction from higher cognition using intermediate abstractions. **PsychoNet** introduces similar pipelines to CNNs. Early spatial layers extract low-level features, similar to early cortical processing, and **Phasor Blocks** localise key characteristic object parts. Subsequently, these parts are encoded in the frequency domain by **Semantic Visual Coding (SVC)** into sparse frequency sub-bands. We believe that this is a naturally emergent representation for semantic information, similar to psychovisual abstractions. FFT denotes the Fast Fourier Transform.

1 INTRODUCTION

Ever since the ImageNet challenge popularised deep learning for computer vision (Krizhevsky et al., 2017; Deng et al., 2009), architectural advances have focused on the design of spatial domain feature extractors, from convolution layers (He et al., 2016; Xie et al., 2017; Huang et al., 2017; Liu

et al., 2022) to more recent token mixers based on attention mechanisms (Dosovitskiy et al., 2021; Tolstikhin et al., 2021; Gao et al., 2021; Rao et al., 2023). Although these models achieve impressive performance, the way they reason in the later layers is often opaque and difficult to interpret. Psychovisual processing—the way human vision encodes and interprets visual information—separates feature extraction from higher cognition using intermediate abstractions, like objects, relations, and categories, providing a natural basis for reasoning (Quiroga et al., 2005; Kriegeskorte et al., 2008; Quiroga, 2012; Le et al., 2024). In contrast, current architectures refine and aggregate image features with homogeneous stacks of spatial layers, entangling feature extraction and reasoning. In this work, we propose a frequency domain representation module for high-level semantic information called Semantic Visual Coding (SVC). We use it to bridge feature extraction and decision-making network stages, enabling a psychovisual-like processing pipeline. Concretely, they are implemented with data-driven band-limited frequency filters, inspired by psychovisual coding schemes from the 1990s that targeted perceptually salient frequencies found by human vision studies (Saadane et al., 1994; Jean-Pierre Guédon et al., 1995; Saadane et al., 1998). SVC extends this idea to high-level representations by allowing networks to discover by learning the sparse frequency subsets most relevant to a given task. These both capture task-specific semantic information and provide potentially interpretable insights into model reasoning, indicating a naturally emergent scheme for high-level abstraction.

Additionally, we develop a deep learning framework called PsychoNet that adapts conventional convolutional neural networks (CNNs) to use SVC. PsychoNet establishes a coherent dual-domain pipeline (Figure 1): initial low-level image features are extracted in the spatial domain and augmented by learning complex-valued representations, then SVC constructs abstractions in the frequency domain that support final decision making. In comparison, previous work alternates between the two domains (Rao et al., 2023; Li et al., 2020a) or treats them as separate signals (Lee-Thorp et al., 2021). To the best of our knowledge, this work also presents the first data-driven exploration of the frequency domain for high-level representation learning in vision, whereas prior studies focus mainly on lower level feature learning or parameterising spatial models (Chi et al., 2020; Rippel et al., 2015; Rao et al., 2023). To summarise, the key contributions of this work are:

- Inspired by psychovisual abstraction, we introduce SVC, a deep-learning based module that automatically learns frequency domain representations of high-level semantic image information. These emerge as sparse selections of coronal frequency sub-bands in the discrete Fourier Transform (DFT).
- Our PsychoNet (Figure 1) is a framework that integrates SVC into CNNs. We show it maintains or improves the performance of common and state-of-the-art CNNs across various classification tasks while enabling psychovisual-like processing: early spatial layers capture low-level features, while SVC performs abstraction and reasoning in the frequency domain.
- Through salience analysis, we find clear evidence of separation between feature extraction in PsychoNet's spatial layers, which capture interpretable object parts, and abstraction in SVC, which encodes these features for high-level processing and reasoning. This mirrors human psychovisual processing and highlights a promising pathway towards interpretable model reasoning.

2 Background

In this section, we review related prior computer vision works on methods with biological motivations, as well as those that use the frequency domain. Additional details/background about psychovisual coding, the Fourier Transform and complex-valued neural networks are provided in Appendix A.

Biologically Inspired Vision. Biologically inspired approaches in computer vision predominantly focus on modelling early vision stages. In particular, much attention has been given to receptive fields (RF) - regions of visual stimuli that elicit strong neural responses in the visual cortex. Mammalian RFs are known to act as directional differential operators, closely resembling traditional image processing functions like wavelets and Gabor filters (Olshausen & Field, 1996; Hubel & Wiesel, 1962; Ringach, 2002). These parallels motivated their use in approximating low-level human vi-

sion, serving as effective feature extractors for basic visual structures like edges and shapes. In deep learning, these functions have been used to build neural networks that mimic cortical pathways (Liu et al., 2023), and early-layer CNN kernels also perform similar directional operations (Krizhevsky et al., 2017; Rippel et al., 2015). Beyond RFs, cortical responses have also been modelled from a frequency domain perspective. Saadane et al. (1998) performed psychovisual experiments which determined thresholds for frequency sensitivities of the human visual cortex. Their results were used to design image quantizers, called 'psychovisual codes' that encoded frequency bands corresponding to perceptually salient features (Figure 2 (left)). In this work, we extend psychovisual coding to high-level features and show that it produces naturally emergent (data-driven) semantic representations. By combining this frequency domain abstraction with (standard or modern) convolutional layers for initial low-level feature extraction, we achieve a first-of-its-kind psychovisual-like reasoning pipeline.

Frequency Domain Learning. Frequency analysis has long been a staple in traditional image processing. Unlike the spatial domain, which is highly localised and expresses features in contiguous pixel neighbourhoods, the frequency domain is more conducive to global representations (see Appendix A.2). Formulated in this space, image processing functions like ridgelets (Candés & Donoho, 1999), curvelets (Starck et al., 2002) and contourlets (Do & Vetterli, 2005) have appealing sparse representations. In fact, they bear a strong resemblance to psychovisual codes since they target specific selections of sub-bands, corresponding to features from different spatial scales. Although these functions have been incorporated into neural networks before, they are only effective on small problems due to their handcrafted nature (Liu et al., 2021).

Accordingly, in deep learning, the frequency domain has largely been used to re-parameterize spatial domain models, particularly convolution operations via the Convolution Theorem (Gonzalez & Woods, 2014). This enables improved performance and efficiency (Rao et al., 2023; Li et al., 2020a; Chi et al., 2020; Guan et al., 2021), analysis of model properties and behaviours (Rippel et al., 2015; Grabinski et al., 2023), and even exploration of theoretical links to neural operators (Kabri et al., 2023). In this work, we move beyond previous groundings in spatial models by constructing a frequency domain-first representation for semantic information. Since studies have shown that global context is crucial for high-level features (Rao et al., 2023; Dosovitskiy et al., 2021), the frequency domain is likely a more natural setting in which to represent and process semantic information. We employ learnable band-limited frequency filters, a data-driven analogue to hand-crafted visual codes, to extract task-relevant semantic information. Perhaps the closest work to ours is the use of frequency filters for feature modulation in domain generalisation, amplifying frequencies with desirable features and suppressing those without (Lin et al., 2023). We extend these ideas to representation learning, applying frequency filters to select task-specific features that serve as the basis for semantic abstraction.

3 METHOD

Semantic Visual Coding. A $N \times N$ digital image, or a spatial feature map derived from it by a neural network, can be viewed as a 2D discrete signal $x[m, n],\ m, n \in 0, ..., N$. This can be represented in the frequency domain as a linear combination of complex-valued sinusoids via the 2D DFT(Cooley et al., 1969):

$$X[u, v] = \frac{1}{N^2} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x[m, n] e^{-2\pi i \left(\frac{um+vn}{N}\right)} \tag{1}$$

where i denotes the imaginary unit. These weights are the (frequency) spectrum of the image and is a complex-valued space known as the frequency domain, which can be computed efficiently using the Fast Fourier Transform (FFT) (Cooley et al., 1969). Psychovisual coding (Saadane et al., 1998) partitions this space into radial sub-bands (2 (left)), and assigns each a threshold corresponding to sensitivity to human vision. These thresholds decide the level of granularity when quantizing images, so that perceptually important features are preserved while others are coarsely represented or discarded.

We introduce *Semantic Visual Coding* (Figure 2 (right)) which aims to generalize this coding principle beyond low-level vision and adapt it to high-level features in deep network layers. In this setting,

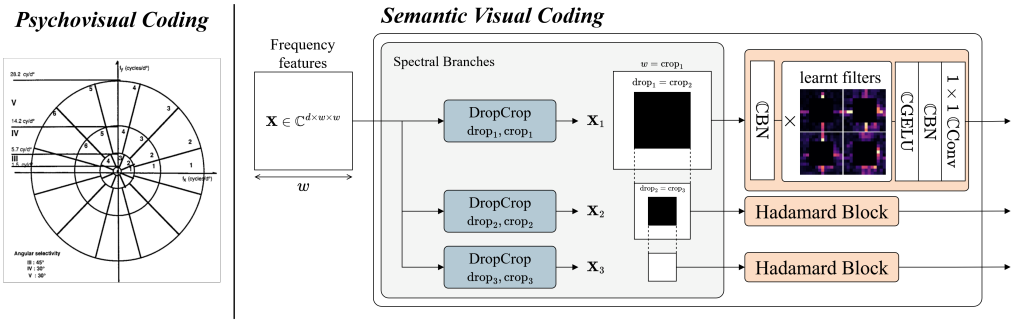

Figure 2: **(Left) Hand-crafted psychovisual coding** from Saadane et al. (1998), which quantizes perceptually salient radial frequencies determined by human vision experiments. **(Right) Our Semantic Visual Coding module,** a data-driven adaptation of psychovisual coding. It uses (1) *Spectral Branches* for radial spectral decomposition (2) *Hadamard Blocks* to apply learnt element-wise filters and channel mixing. $\mathbb{C}$Conv/BN/GELU denote complex-valued convolution, batch norm and GELU operations - see Appendix X.

the selection of frequencies should no longer be fixed by handcrafted thresholds, but instead learnt directly from data to encode task-relevant semantic information. Semantic Visual Coding has the following formulation:

Let $\boldsymbol{X} \in \mathbb{C}^{d \times w \times w}$ be frequency domain input features, where d is the number of channels and $w \times w$ the spatial size.

1. We apply *Spectral Branches* which replicate the radial frequency partitioning in psychovisual codes. These divide $\boldsymbol{X}$ into disjoint rectangular sub-bands $\boldsymbol{X}_1, \boldsymbol{X}_2, ...$ using *DropCrop* blocks, which set a lower frequency boundary (drop_i) by zeroing central frequencies and an upper boundary (crop_i) by cropping $\boldsymbol{X}$ to size $d \times \text{crop}_i \times \text{crop}_i$.
2. For each sub-band $\boldsymbol{X_i}$, *Hadamard Blocks* apply a set of learnt filters $\boldsymbol{W_i} \in \mathbb{C}^{d \times \text{crop}_i \times \text{crop}_i}$ via element-wise (Hadamard) multiplication. Additionally, we also apply Softmax across the channels of $\boldsymbol{W_i}$ to amplify important frequency selections and suppress unimportant ones, emulating the quantization in psychovisual coding.
3. Hadamard Blocks further apply complex 1×1 convolution block to mix together different information extracted by each channel/filter, yielding our final representations.

In practice, for all models we apply Spectral Branches at a spatial resolution of $w = 14$ and use three sub-bands with [crop_i, drop_i] values of [14, 8], [8, 4] and [4, 1] respectively. More details can be found in Appendix B.

PsychoNet. The PsychoNet framework adapts standard spatial CNNs to use Semantic Visual Coding. This setting enables experimentation to assess if our codes produce meaningful high-level semantic representations that support interpretable reasoning, as well as practical performance evaluation against standard baselines. In our experiments, we apply PsychoNet to ResNet (He et al., 2016) and ConvNeXt (Liu et al., 2022) architectures, and provide full architectural configurations in Appendix B.

The architecture of the PsychoNet framework comprises:

1. A number of low-level feature extraction layers are retained from the base CNN, for example, the first two resolution stages in the case of ResNet-50 and ResNet-101.
2. The remaining spatial layers are replaced with Phasor Blocks, described further below. Compared to the original CNNs layers, Phasor Blocks typically use higher spatial resolution and only downsample down to 14×14 instead of 7×7. Though this increases FLOPs (Appendix C), we found there is insufficient granularity at 7×7 to clearly separate low and high frequencies after FFT.

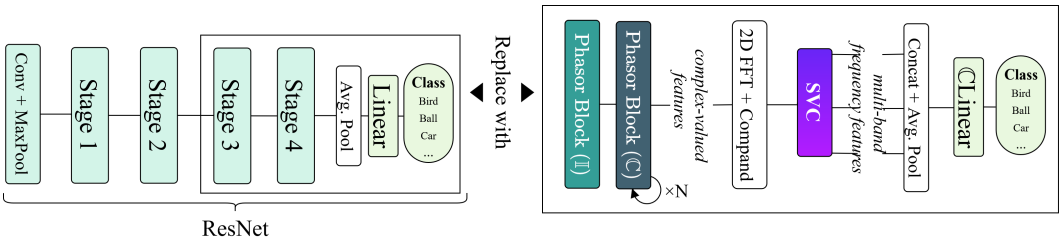

Figure 3: Example: Converting ResNet-50/101 image classification models with PsychoNet.

3. 2D FFT is applied to convert features from the spatial to frequency domain. As with most visual features, the magnitude of $\boldsymbol{X}$'s DC (0 frequency) and low frequency features typically dominate over those of high frequency ones, so we use a simple companding operation to reduce this imbalance (Appendix B).
4. SVC is applied, and its outputs from each frequency band aggregated. These are then used for output prediction directly in the frequency domain using a complex-valued linear layer.

The following section outlines the motivation and formulation of Phasor Blocks, with additional architectural details provided in Appendix B

Phasor Blocks. Filtering in the frequency domain is powerful as it captures information encoded as both magnitude and phase. However, PsychoNet has real-valued input (natural images) and real features incur conjugate symmetry of the Fourier Transform (FT), which renders half of the frequency domain redundant. While this constraint matters little when only using the frequency domain for convolution (Rao et al., 2023; Li et al., 2020a), it limits learnt filtering from fully exploiting complex representations. As such, we introduce Phasor Blocks (Figure 4) to augment real-valued spatial features with complementary complex-valued ones, breaking symmetry to improve the specificity of the subbands learnt. In practice, we largely just derive new imaginary components from existing real features, and only employ a minimal number of expensive complex operations (e.g. complex convolutions). We believe this simplification to be sufficient since generating enough complex features to break symmetry is a considerably easier task than supporting rich complex spatial computations, as in standard complex-valued networks (Trabelsi et al., 2018).

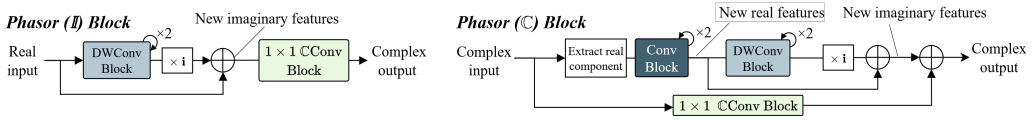

Figure 4: **Phasor Blocks architectures**. Phasor $(\mathbb{I})$ blocks generate complementary imaginary features for real-valued input. Phasor $(\mathbb{C})$ blocks generate additional complex features for complex-valued input based on its real component. Implementations of normal convolution (Conv), depthwise convolution (DWConv) Liu et al. (2022) and complex-valued convolution ($\mathbb{C}$Conv) blocks for each of our models are presented in Figure A.2

Phasor $(\mathbb{I})$ blocks generate an initial set of imaginary components using depthwise convolution blocks, comprising pairs of depthwise and 1×1 (pointwise) convolution layers. This configuration decouples spatial and channel mixing, which is intended to encourage cross-channel interactions without interfering with spatial relationships. In natural complex signals, the real and imaginary components carry complementary information for the same spatial location (Gonzalez & Woods, 2014; Lee et al., 2022), so it is likely important that our generated imaginary features do not significantly introduce new spatial information. A 1×1 complex convolution block then mixes the real and imaginary features. Subsequently, Phasor $(\mathbb{C})$ blocks further refine the complex representations. The top branch generates new real and imaginary features, while the bottom channel-mixes the original features and combines them with the new ones. For ConvNeXt, we use 'DWConv' blocks in place of Phasor $(\mathbb{C})$'s 'Conv' blocks to more closely match its original depthwise convolution layers. Appendix B.2 presents further details about Phasor Block configurations.

4 EXPERIMENTS AND RESULTS

We evaluate PsychoNet on ResNet and ConvNeXt models across multiple classification datasets. Key results are highlighted in Table 1, while full results are reported in Tables A.8 and A.9 in the appendix. Our experiments span small to large-scale benchmarks: CIFAR-10/100 (Krizhevsky, 2009) both contain ∼50K low-resolutions images, while ImageNet-100 is a moderate sized subset (∼130K images) of ImageNet (Deng et al., 2009). Finally, we also use the standard large ImageNet-1K subset containing ∼1.2 million training and ∼50K validation images. Full dataset, training and hardware details are presented in Appendix C.

Model	Param. (M)	# Layers	CIFAR-10	CIFAR-100	IN100	IN1K
ResNet152	60.10	156	93.17	77.51	83.60	79.59
Psycho-L	61.28	93 $\downarrow$ *40.4% less*	94.95	79.64	84.82	79.85
ResNet270	89.60	276	76.51	50.87	83.80	80.01
Psycho-H	88.61	93 $\downarrow$ *66.3% less*	94.68	79.89	85.00	80.45
ConvNeXt-S	50.22	113	94.09	76.96	86.98	80.78
PsychoDW	49.51	106 $\downarrow$ *6.2% less*	95.46	79.67	86.76	80.59

Table 1: Classification results (% top-1 accuracies) for PsychoNet on the largest two ResNet sizes we try and ConvNeXt-S, for CIFAR-10, CIFAR-100, ImageNet-100 (IN100) and ImageNet-1K (IN1K). Each pair of rows (separated by horizontal lines) compares a baseline CNN and the PsychoNet based on it. Full classification results for all models are presented in Appendix C.

ResNet was chosen as it is a simple and very well-known baseline, which additionally relies heavily on increasing layer depth for scaling model size. Psycho-S/B/L/H models are based on ResNet-50/101/152/270 respectively. Since we hypothesise that SVC should handle high-level processing, we stop increasing Phasor Blocks depth after Psycho-B/ResNet-101 to see if it can replace the role of late spatial layers (the width of existing layers are increased to compensate for parameter size.) We further develop PsychoDW, based on ConvNeXt-S, as an example of a state-of-the-art CNN model, to determine if PsychoNet is compatible with modern architectures. Full model configurations are presented in Appendix B. Overall, PsychoNet slightly improves the performance of each baseline model, except on ImageNet-100 and ImageNet-1K for ConvNeXt-S, where it is slightly worse. We also found that PsychoNet scaling is considerably less dependent on layer depth for large ResNet models: Psycho-L and Psycho-H use $\sim 1.7\times$ and $\sim 3\times$ less layers than their ResNet baselines respectively (Figure 5 (a), Table 1).

Although PsychoNet models use considerably more FLOPs than their respective CNN baselines (Table A.11), they achieve clear psychovisual separation of low and high-level processing, as detailed later in this section. The increased computation is attributed to (1) Phasor Blocks requiring higher-resolution features than the CNN layers, to support 14×14 SVC filters and (2) complex-valued operations (complex convolution etc.) being poorly optimised in deep learning frameworks.

Filter learning. Figure 6 visualises SVC filters learnt by PsychoNet, showing the top spatial principal components as an approximation of the most important frequency features. We find that filters across every sub-band learn very sparse selections of frequencies. However, this requires sufficiently large training corpora - the ImageNet-100-trained filters are noticeably noisier than for ImageNet-1K, and those for CIFAR-10/100 (Figure A.4) even more still. This suggests that these patterns correspond to a data-driven representation naturally emergent from visual information. Additionally, we ran ablation experiments to evaluate the effects of Phasor Blocks and Spectral Branching (Appendix D). Removing Phasor Blocks (and replacing them with ResNet-style residual bottleneck blocks (ResBlocks)) removes complex-valued spatial features and introduces conjugate symmetry to the frequency domain. This yields symmetric filter features that are far less expressive. Likewise, removing the spectral decomposition of Spectral Branches (we use one global filter instead of three band-limited ones) also reduces filter sparsity. This is likely as exposure to the entire frequency domain makes it harder for filters to specialise to specific sub-bands.

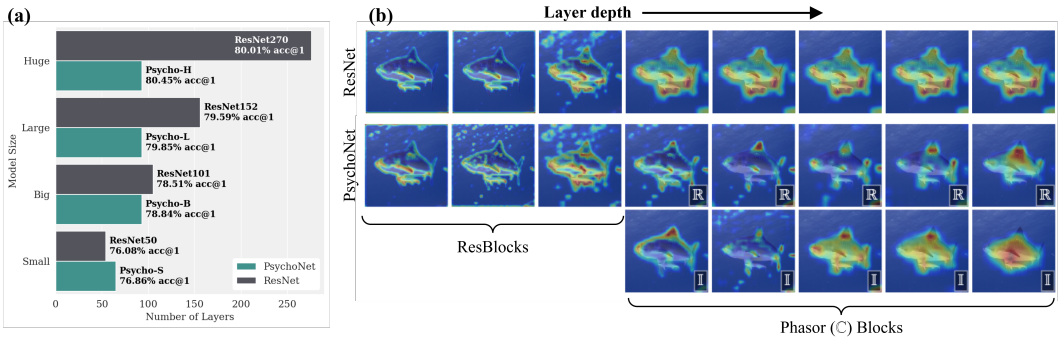

Figure 5: **(a)** Comparison of model depth when scaling ResNet vs. PsychoNet. **(b)** Comparison between activation maps (via KPCA-CAM) of Psycho-B and ResNet-101 for a range of layer depths. Real and imaginary components are denoted by $\mathbb{R}$ and $\mathbb{I}$.

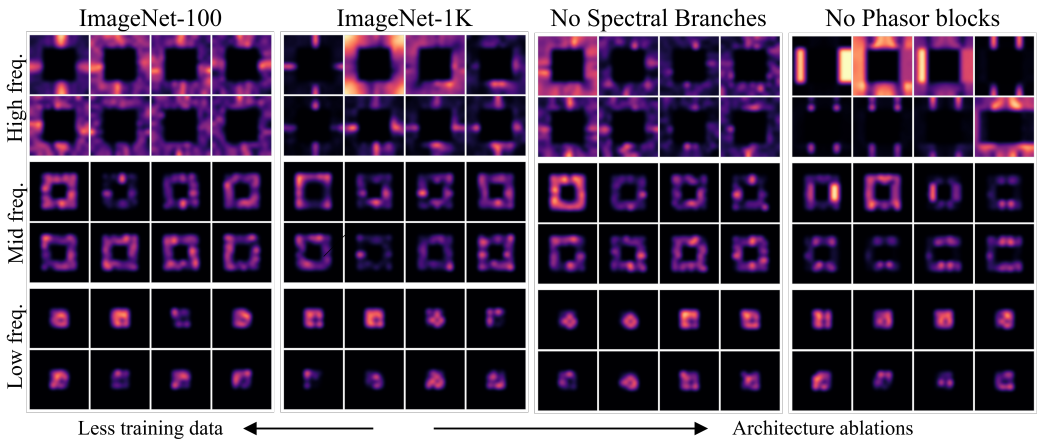

Figure 6: SVC filters learnt by Psycho-B trained on ImageNet-100 and ImageNet-1K, as well as for two ablation models on ImageNet-1K. Bilinear smoothing has been applied. 'High/mid/low freq.' refer to the [14, 8], [8, 4] and [4, 1] frequency sub-bands created by Spectral Branches. 'No Spectral Branches' removes Spectral Branches and uses a single Hadamard Block with global filters - we extract sub-bands only for the visualisation. 'No Phasor Blocks' replaces all Phasor Blocks with ResBlocks.

Representation Analysis. As PsychoNet is far less depth-dependent than ResNet at larger scales, SVC likely already performs the high-level processing that additional spatial layers would otherwise introduce. Accordingly, we visualise layer activations using KPCA-CAM (Karmani et al., 2024) to compare spatial processing between the two models (Figure 5 (b)). This approach generates salience maps by projecting activations onto the first principal component of their kernel PCA. ResNet's early layers target low-level features (edges), but later salience regions quickly grow to cover the entire subject. From these, it is not particularly clear which parts of the shark each layer is focusing on, suggesting they perform a wide range of diffuse operations. In contrast, early-mid level Phasor Blocks clearly fixate on morphological features of the shark, such as its snout, fins and tail. Figure 7 shows further examples of Phasor Blocks localising key characteristics of different object categories, such as dog ears, elephant tusks and car wheels. Since KPCA-CAM only uses activations of the layer being visualised and is not influenced by model predictions (e.g. via backpropagation in gradient-based CAMs), the primary function of Phasor Blocks must be to extract these semantic object parts. Interestingly, it appears that the imaginary components of Phasor Block activations capture more global features than the real components (i.e. a dog's face vs. its ears), suggesting a rich utilisation of complex-valued representations.

For initial exploration of SVC's encoding mechanisms, we use HiResCAM (Draelos & Carin, 2020) for gradient-based activation visualisation. It produces salience maps by element-wise multiplying

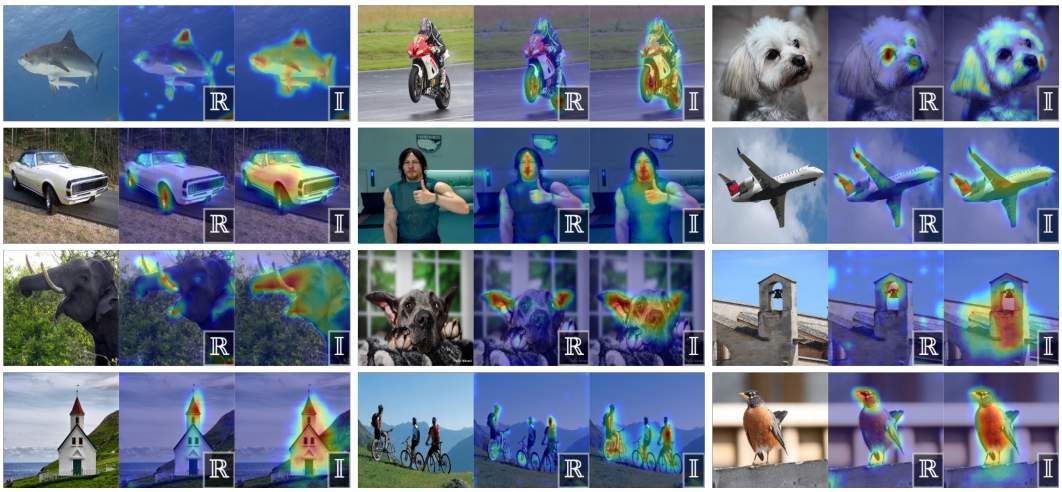

Figure 7: Assorted activation maps (via KPCA-CAM) for mid-level Phasor Blocks of Psycho-B. Real and imaginary components are denoted by $\mathbb{R}$ and $\mathbb{I}$.

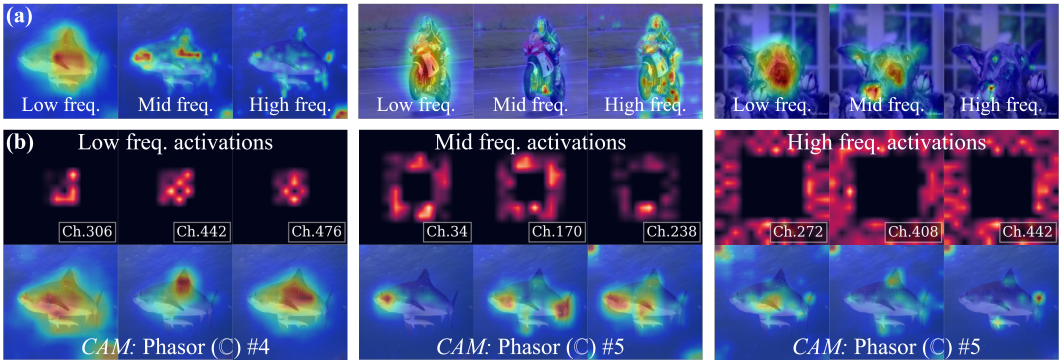

Figure 8: Psycho-B Phasor Block salience maps (via HiResCAM) conditioned on gradients **(a)** from individual Spectral Branch sub-bands and **(b)** from individual frequency domain feature channels.

layer activations with gradients backpropogated from model predictions, so in classification the salience regions have a high contribution to the class prediction. We extend this approach to isolate regions used by specific parts of SVC by first masking (setting to zero) gradients from the other components, enabling exploration of how SVC encodes Phasor Blocks features. First, we examine each of the three sub-bands created by PsychoNet's Spectral Branches. After masking gradients of Hadamard Blocks for all but one of the sub-bands, Phasor Blocks' salience regions reveal that SVC distributes object parts by scale. Figure 8 (a) shows that the low-frequency sub-band focuses on subjects broadly, while mid-high frequencies isolate more specific parts of different sizes. We also isolate activations from individual Hadamard Block channels, showing that within each band, channels specialise to distinct object parts and correspond to distinct sparse frequency selections (Figure 8 (b)).

Overall, these result suggest that SVC learns a semantic intermediate representations that encodes selections of object parts. Given that SVC is placed immediately before the decision making (classification) layers of PsychoNet, it is likely selecting those most relevant to the task. In doing so, SVC functions as an abstraction bridging part extraction in Phasor Blocks and higher-level reasoning, mirroring the role of intermediate abstractions in psychovisual processing.

Limitations and Future Work. The key limitation of our work is that though we show SVC organises and encodes selections of object components, it is not clear how these representations are used for reasoning, which remains an important direction for future work. Similarly, we should also explore addressing the high FLOP usage of PsychoNet, perhaps by exploring optimisations for

complex-valued operations (e.g. employing Cauchy-Riemann identities (Ahlfors, 1979)) or more sophisticated formulations of Phasor Blocks.

Additionally, it would also be informative to explore applying PsychoNet to broader task types. It would be insightful to explore image-to-image tasks, which may allow SVC to utilize wider frequency ranges than classification, as well as domains with natural frequency domain data, such as magnetic resonance imaging (Chandra et al., 2021). Finally, it is also known that aliasing can afflict standard CNN architectures (Grabinski et al., 2022); future work should assess its impact on our frequency-domain representations and whether mitigation can improve results.

5 Conclusion

In this work, we introduced Semantic Visual Coding (SVC), the first high-level vision representation learnt in the frequency domain that produces sparse, data-driven coronal selections of discrete Fourier space. Our PsychoNet framework integrating SVCs show that it can maintain performance across multiple classification datasets, but is less depth-dependent, suggesting that SVC improves high-level processing previously done by deep spatial layers. In contrast to the entangled computation of conventional CNNs, we find that PsychoNet separates processing stages: Phasor Blocks extract semantically meaningful object parts, while SVCs encode and organise these parts into sparse, frequency domain representations used to make classification decisions that can be visualized. This pipeline mimics intermediate abstractions used by the brain to separate feature extraction from higher cognition. While further work is required to understand the reasoning mechanisms of SVCs, it is clear that frequency domain abstraction is a promising direction for interpretable human-like model reasoning.

Acknowledgements

In accordance to ICLR 2026 guidelines, we acknowledge the use of large language models (LLMs) in preparing this manuscript. Their role was limited to assisting with editing and polishing writing.

Additional acknowledgements will be added after deanonymization.

Ethics Statement

All authors have reviewed the ICLR 2026 code of ethics and verified to the best of our knowledge that the work in our paper conforms with it. In particular, this work introduces a new theoretical framework, so it is unlikely to cause direct harm or negative impacts to society. Additionally, we only use open datasets such as ImageNet, so privacy concerns do not arise, though we acknowledge that these datasets contain known biases that may influence model behaviour.

Reproducibility Statement

We are committed to ensuring the reproducibility of our results. All experiments in this work were conducted on publicly available datasets, which have been appropriately cited. Detailed training recipes and hardware details are presented in Appendix C and Appendix D. Detailed model configurations are presented in Appendix B. After deanonymization, we will also release our code repository including training scripts, model weights and instructions to reproduce all of our results.

REFERENCES

Lars V. Ahlfors. *Complex Analysis*. McGraw-Hill, New York, 3rd edition, 1979. ISBN 978-0070006577.

Irwan Bello, W. Fedus, Xianzhi Du, E. D. Cubuk, A. Srinivas, Tsung-Yi Lin, Jonathon Shlens, and Barret Zoph. Revisiting ResNets: Improved Training and Scaling Strategies. *Neural Information Processing Systems*, 2021.

E. J. Candés and D. L. Donoho. Ridgelets: a key to higher-dimensional intermittency? *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 357(1760):2495–2509, 1999. doi: http://dx.doi.org/10.1098/rsta.1999.0444. URL http://dx.doi.org/10.1098/rsta.1999.0444.

Shekhar S Chandra, Marlon Bran Lorenzana, Xinwen Liu, Siyu Liu, Steffen Bollmann, and Stuart Crozier. Deep learning in magnetic resonance image reconstruction. *Journal of Medical Imaging and Radiation Oncology*, 65(5):564–577, 2021. doi: https://doi.org/10.1111/1754-9485.13276. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/1754-9485.13276.

Lu Chi, Borui Jiang, and Yadong Mu. Fast Fourier Convolution. In *Advances in Neural Information Processing Systems*, volume 33, pp. 4479–4488. Curran Associates, Inc., 2020. URL https://papers.nips.cc/paper_files/paper/2020/hash/2fd5d41ec6cfab47e32164d5624269b1-Abstract.html.

Elizabeth Cole, Joseph Cheng, John Pauly, and Shreyas Vasanawala. Analysis of deep complex-valued convolutional neural networks for mri reconstruction and phase-focused applications. *Magnetic Resonance in Medicine*, 86(2):1093–1109, 2021. doi: https://doi.org/10.1002/mrm.28733. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/mrm.28733.

J. W Cooley, P. Lewis, and P. Welch. The finite Fourier transform. *Audio and Electroacoustics, IEEE Transactions on*, 17(2):77–85, June 1969. ISSN 0018-9278.

Muneer Dedmari, Sailesh Conjeti, Santiago Estrada, Phillip Ehses, Tony Stocker, and Martin Reuter. Complex fully convolutional neural networks for mr image reconstruction. In *Machine Learning for Medical Image Reconstruction : first International Workshop, MLMIR 2018*, volume 1. Springer, 2018.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, June 2009. doi: 10.1109/cvpr.2009.5206848.

M.N. Do and M. Vetterli. The contourlet transform: an efficient directional multiresolution image representation. *IEEE Transactions on Image Processing*, 14(12):2091–2106, December 2005. ISSN 1941-0042. doi: 10.1109/TIP.2005.859376. URL https://ieeexplore.ieee.org/document/1532309.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Rachel Lea Draelos and Lawrence Carin. Use HiResCAM instead of Grad-CAM for faithful explanations of convolutional neural networks. *arXiv preprint arXiv:2011.08891*, November 2020. URL https://arxiv.org/abs/2011.08891.

Peng Gao, Jiasen Lu, Hongsheng Li, Roozbeh Mottaghi, and Aniruddha Kembhavi Kembhavi. Container: Context Aggregation Network. *Neural Information Processing Systems*, 2021.

Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing 3rd Edition*. Prentice Hall, January 2014.

Julia Grabinski, Steffen Jung, Janis Keuper, and Margret Keuper. Frequencylowcut pooling - plug and play against catastrophic overfitting. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIV*, pp. 36–57, Berlin, Heidelberg, 2022. Springer-Verlag. ISBN 978-3-031-19780-2. doi: 10.1007/978-3-031-19781-9_3. URL https://doi.org/10.1007/978-3-031-19781-9_3.

Julia Grabinski, Janis Keuper, and Margret Keuper. As large as it gets: Learning infinitely large filters via neural implicit functions in the fourier domain. *ArXiv*, abs/2307.10001, 2023. URL https://api.semanticscholar.org/CorpusID:259982481.

Bochen Guan, Jinnian Zhang, William A. Sethares, Richard Kijowski, and Fang Liu. Spectral Domain Convolutional Neural Network. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2795–2799, June 2021. doi: 10.1109/ICASSP39728.2021.9413409. ISSN: 2379-190X.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160(1):106–154.2, January 1962. ISSN 0022-3751. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1359523/.

Jean-Pierre Guédon, Jean-Pierre Guédon, Dominique Barba, Dominique Barba, Nicole Bürger, and Nicole Burger. Psychovisual image coding via an exact discrete Radon transform. *Other Conferences*, 2501:562–572, April 1995. doi: 10.1117/12.206765.

Samira Kabri, Tim Roith, Daniel Tenbrinck, and Martin Burger. Resolution-invariant image classification based on fourier neural operators. In *Scale Space and Variational Methods in Computer Vision: 9th International Conference, SSVM 2023, Santa Margherita Di Pula, Italy, May 21–25, 2023, Proceedings*, pp. 236–249, Berlin, Heidelberg, 2023. Springer-Verlag. ISBN 978-3-031-31974-7. doi: 10.1007/978-3-031-31975-4_18. URL https://doi.org/10.1007/978-3-031-31975-4_18.

Sachin Karmani, Thanushon Sivakaran, Gaurav Prasad, Mehmet Ali, Wenbo Yang, and Sheyang Tang. KPCA-CAM: Visual Explainability of Deep Computer Vision Models Using Kernel PCA. *IEEE International Workshop on Multimedia Signal Processing*, 2024. doi: 10.1109/mmsp61759.2024.10743968.

Nikolaus Kriegeskorte, Marieke Mur, Douglas A. Ruff, Roozbeh Kiani, Jerzy Bodurka, Hossein Esteky, Keiji Tanaka, and Peter A. Bandettini. Matching categorical object representations in inferior temporal cortex of man and monkey. *Neuron*, 2008. doi: 10.1016/j.neuron.2008.10.043.

Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. URL https://api.semanticscholar.org/CorpusID:18268744.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet classification with deep convolutional neural networks. *Communications of The ACM*, 60(6):84–90, May 2017. doi: 10.1145/3065386.

Lynn Le, Paolo Papale, K. Seeliger, Antonio Lozano, Thirza Dado, Feng Wang, Pieter R. Roelfsema, M. van Gerven, Yağmur Güçlütürk, and Umut Güçlü. Monkeysee: Space-time-resolved reconstructions of natural images from macaque multi-unit activity. *Neural Information Processing Systems*, 2024.

ChiYan Lee, Hideyuki Hasegawa, and Shangce Gao. Complex-Valued Neural Networks: A Comprehensive Survey. *IEEE/CAA Journal of Automatica Sinica*, 9(8):1406–1426, August 2022. doi: 10.1109/jas.2022.105743.

J. Lee-Thorp, J. Ainslie, Ilya Eckstein, and Santiago Ontañón. FNet: Mixing Tokens with Fourier Transforms. *North American Chapter of the Association for Computational Linguistics*, 2021. doi: 10.18653/v1/2022.naacl-main.319.

Shaohua Li, Kaiping Xue, Bin Zhu, Chenkai Ding, Xindi Gao, David Wei, and Tao Wan. Falcon: A fourier transform based approach for fast and secure convolutional neural network predictions. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8702–8711, 2020a. doi: 10.1109/CVPR42600.2020.00873.

Wenhan Li, Wenqing Xie, and Zhifang Wang. Complex-valued densely connected convolutional networks. In Jianchao Zeng, Weipeng Jing, Xianhua Song, and Zeguang Lu (eds.), *Data Science*, pp. 299–309, Singapore, 2020b. Springer Singapore. ISBN 978-981-15-7981-3.

Shiqi Lin, Zhizheng Zhang, Zhipeng Huang, Yan Lu, Cuiling Lan, Peng Chu, Quanzeng You, Jiang Wang, Zicheng Liu, Amey Parulkar, Viraj Navkal, and Zhibo Chen. Deep Frequency Filtering for Domain Generalization . In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11797–11807, Los Alamitos, CA, USA, June 2023. IEEE Computer Society. doi: 10.1109/CVPR52729.2023.01135. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52729.2023.01135.

Mengkun Liu, Licheng Jiao, Xu Liu, Lingling Li, Fang Liu, and Shuyuan Yang. C-CNN: Contourlet Convolutional Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(6):2636–2649, June 2021. ISSN 2162-2388. doi: 10.1109/TNNLS.2020.3007412. URL https://ieeexplore.ieee.org/document/9145825. Conference Name: IEEE Transactions on Neural Networks and Learning Systems.

Mengkun Liu, Licheng Jiao, Xu Liu, Lingling Li, Fang Liu, Shuyuan Yang, and Xiangrong Zhang. Bio-Inspired Multi-scale Contourlet Attention Networks. *IEEE transactions on multimedia*, pp. 1–16, January 2023. doi: 10.1109/tmm.2023.3304448.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11966–11976, 2022. doi: 10.1109/CVPR52688.2022.01167.

Bruno A. Olshausen and David J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607, June 1996. ISSN 1476-4687. doi: 10.1038/381607a0. URL https://www.nature.com/articles/381607a0.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

Rodrigo Quian Quiroga. Concept cells: the building blocks of declarative memory functions. *Nature Reviews Neuroscience*, 2012. doi: 10.1038/nrn3251.

Rodrigo Quian Quiroga, Leila Reddy, Gabriel Kreiman, Christof Koch, and Itzhak Fried. Invariant visual representation by single neurons in the human brain. *Nature*, 2005. doi: 10.1038/nature03687.

Yongming Rao, Wenliang Zhao, Zheng Zhu, Jie Zhou, and Jiwen Lu. Gfnet: Global filter networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9): 10960–10973, 2023. doi: 10.1109/TPAMI.2023.3263824.

Dario L. Ringach. Spatial structure and symmetry of simple-cell receptive fields in macaque primary visual cortex. *Journal of Neurophysiology*, 88(1):455–463, July 2002. doi: 10.1152/jn.2002.88.1.455.

Oren Rippel, Jasper Snoek, and Ryan P. Adams. Spectral Representations for Convolutional Neural Networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, pp. 2449–2457, Cambridge, MA, USA, 2015. MIT Press.

Abdelhakim Saadane, H. Senane, and Dominique Barba. Design of psychovisual quantizers for a visual subband image coding. *Other Conferences*, 2308:1446–1453, September 1994. doi: 10.1117/12.185903.

Abdelhakim Saadane, Hakim Senane, and Dominique Barba. Visual Coding. *Journal of Visual Communication and Image Representation*, 9(4):381–391, December 1998. doi: 10.1006/jvci.1998.0393.

Simone Scardapane, Steven Van Vaerenbergh, Amir Hussain, and Aurelio Uncini. Complex-valued Neural Networks with Non-parametric Activation Functions, February 2018. URL http://arxiv.org/abs/1802.08026. arXiv:1802.08026 [cs].

Jean-Luc Starck, E.J. Candes, and D.L. Donoho. The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, 11(6):670–684, June 2002. ISSN 1941-0042. doi: 10.1109/TIP.2002.1014998. Conference Name: IEEE Transactions on Image Processing.

I. Tolstikhin, N. Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. MLP-Mixer: An all-MLP Architecture for Vision. In *Neural Information Processing Systems*, 2021.

Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, Joao Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Christopher J. Pal. Deep Complex Networks. In *International Conference on Learning Representations*, February 2018. URL https://openreview.net/forum?id=H1T2hmZAb.

Bhavya Vasudeva, Puneesh Deora, Saumik Bhattacharya, and Pyari Mohan Pradhan. Compressed sensing mri reconstruction with co-vegan: Complex-valued generative adversarial network. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1779–1788, 2022. doi: 10.1109/WACV51458.2022.00184.

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5987–5995, 2017. doi: 10.1109/CVPR.2017.634.

Saurabh Yadav and Koteswar Rao Jerripothula. FCCNs: Fully Complex-valued Convolutional Networks using Complex-valued Color Model and Loss Function. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10655–10664, October 2023. doi: 10.1109/ICCV51070.2023.00981. URL https://ieeexplore.ieee.org/document/10377516. ISSN: 2380-7504.

Appendices

In the following we present appendices to our work, structured as follows: Appendix A presents additional background material. Appendix B provides detailed information about the architectural configurations of all models used in our experiments. Appendix C presents results, dataset information and training recipes for all of our classification experiments. Appendix D provides full results and details for PsychoNet architectural ablation studies.

A Background

In this section we present additional background and details about psychovisual coding, the Fourier Transform and complex-valued networks.

A.1 Psychovisual Coding

The psychovisual coding scheme from Saadane et al. (1998) was originally developed for image quantization and compression that is perceptually lossless to humans. It first decomposes the frequency domain into a number of coronal sub-bands, as shown in Figure A.1 (enlarged version of Figure 2 (left)). Then, authors conducted experiments with human observers measuring their sensitive to quantization noise across different sub-bands. Based on these results, they derived sub-band-specific quantization thresholds and step-sizes to try encode only perceptually salient image features. Our SVC is a data-driven adaptation of this approach, using band-limited frequency filters learn sparse frequency selections using supervisory signals from a classification task.

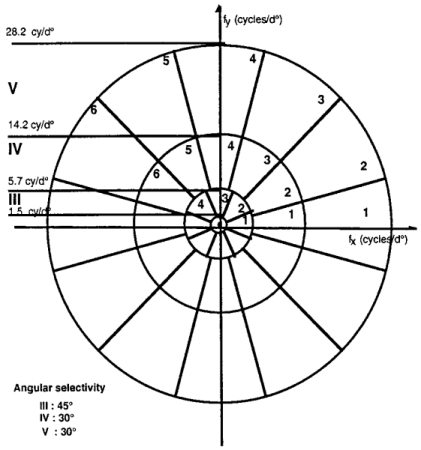

Figure A.1: Coronal frequency sub-bands used in psychovisual coding from Saadane et al. (1998).

A.2 The Fourier Transform and the Frequency Domain

In Section 3 we only describe the 2D DFT as digital images are discrete 2D signals in the spatial domain, while the standard FT operates on continuous signals. The DFT is derived by first viewing a discrete signal as the product of a continuous signal and a sequence of unit impulses (sampling), applying the FT to yield a continuous function in the frequency domain, then sampling it again to discretize it. Detailed derivations of both the FT and DFT may be found in most image processing texts, such as Gonzalez & Woods (2014). There are also inverse transforms, namely the Inverse Fourier Transform (IFT) and Inverse Discrete Fourier Transform (IDFT), for transforming frequency domain signals back into the spatial domain. While we do not use them in PsychoNet, they reflect the duality between the spatial and frequency domains - any operation in one domain has a counterpart in the other. The most famous example of this relationship is the Convolution Theorem.

Let $x[u, v], y[u, v], u, v \in 0, ..., N-1$ be two discrete $N \times N$ spatial signals. The circular convolution of these two signals is defined as:

$$x[u, v] * y[u, v] = \frac{1}{N^2} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x[m, n] y[((u-m))_N, ((v-n))_N] \tag{2}$$

where $(.)_N$ denotes modulo N. The Convolution Theorem (Gonzalez & Woods, 2014) then states that:

$$\mathcal{F}[x * y] = \mathcal{F}[x] \odot \mathcal{F}[y] \text{ or equivalently } x * y = \mathcal{F}^{-1}[\mathcal{F}[x] \odot \mathcal{F}[y]] \tag{3}$$

where $\mathcal{F}[.]$ and $\mathcal{F}^{-1}[.]$ denote the DFT and IDFT, and $\odot$ the Hadamard product. Hence, circular convolution in the spatial domain is equivalent to applying the Hadamard product in the frequency domain. As such, the frequency domain is highly conducive to global representations, since each element of an image's frequency spectra presents a unique global view of the image, analogous to convolving it with a directional striped kernel.

In practice, the DFT and IDFT are computed using the Fast Fourier Transform and Inverse Fast Fourier Transform respectively (Cooley et al., 1969). Note that if x and y were multi-channel features instead, i.e. of dimension $d \times N \times N$ for d channels like the input features and learnt filters of our Hadamard Blocks, then the frequency domain Hadamard product is equivalent to circular *depthwise* convolution in the spatial domain. Unlike the Conv2D operation of CNNs, this does not mix channels, which is why both of our Hadamard Blocks and the Global Filter block from Rao et al. (2023) include explicit channel-mixing via 1×1 convolution layers.

A.3 Complex-valued Neural Networks

Most work for complex-valued neural networks involve developing components of these networks to work in the complex domain, such as activation functions (Scardapane et al., 2018). Most complex-valued CNNs use the network blocks introduced by Trabelsi et al. (2018). The distributive property of convolution allows convolution between a complex input $\boldsymbol{h} = \boldsymbol{a} + i\boldsymbol{b}$ and a complex kernel $\boldsymbol{W} = \boldsymbol{W_R} + i\boldsymbol{W_I}$ to be decomposed into four real-valued component wise convolutions:

$$\boldsymbol{W} * \boldsymbol{h} = (\boldsymbol{W_R} * \boldsymbol{a} - \boldsymbol{W_I} * \boldsymbol{b}) + i\,(\boldsymbol{W_I} * \boldsymbol{a} + \boldsymbol{W_R} * \boldsymbol{b}) \tag{4}$$

Consequently, complex-valued convolution layers are usually more computationally and memory intensive (additionally stores imaginary features) than real-valued ones. Trabelsi et al. (2018) also developed complex normalization methods and activation functions. Complex-valued modules in PsychoNet use the complex-valued convolution ($\mathbb{C}$Conv) and batch-normalization ($\mathbb{C}$BN) layers from Trabelsi et al. (2018), and a naïve adaptation of the GELU activation function ($\mathbb{C}$GELU) which just applies the original function to real and imaginary channels separately.

When applying complex-valued networks to real-valued images, most works use a small initial module to convert the input into complex-valued features. However, such approaches have yielded only minor improvements in the past over directly using real-valued networks (Trabelsi et al., 2018; Li et al., 2020b). Accordingly, recent complex-valued networks predominantly focus on domains with naturally complex data, such as magnetic resonance imaging, radar and audio signal processing (Dedmari et al., 2018; Vasudeva et al., 2022; Cole et al., 2021; Lee et al., 2022; Trabelsi et al., 2018). To try bridge this gap, a complex-valued colour space by reinterpreting the cylindrical coordinates of the HSV colour model as 2D magnitude and phase was developed Yadav & Jerripothula (2023). They applied this to standard complex-valued CNNs, improving results on common image classification tasks, but retained the high complexity of complex-valued networks. On the other hand, PsychoNet primarily uses real-valued modules that learn to generate *complementary* complex-valued features to given real features, as described in Section 3.

B MODEL CONFIGURATIONS

Here we provide full details of the architectural configurations of all of our models. For all tables, we use the ResNet approach of counting the number of model layers as the number of convolutional and linear layers; each element-wise filter block in Hadamard Blocks are also counted as one layer.

B.1 CNN BASELINES

The ResNet50, 101 and 152 models we use are from He et al. (2016) and are implemented in most common deep learning frameworks (we use the one from `PyTorch` (Paszke et al., 2017)). For ResNet270, we follow the block configurations in Bello et al. (2021), but do not implement any of the newer blocks/layers they also introduce, so it purely just adds more residual bottleneck blocks (ResBlocks) to ResNet152 for fair scaling. Table A.1 compares the sizes of the four ResNet models as well as their block configurations, grouped by feature resolution (which are 56×56, 28×28, 14×14 and 7×7).

Table A.1: ResNet block configurations.

Model	Parameters (M)	# Layers	# Blocks
ResNet50	25.56	54	`[3-4-6-3]`
ResNet101	44.55	105	`[3-4-23-3]`
ResNet152	60.19	156	`[3-8-36-3]`
ResNet270	89.60	276	`[4-29-53-3]`

For ConvNeXt-S, we follow the original implemention in Liu et al. (2022).

B.2 PSYCHONET

Detailed architectural configurations of each PsychoNet model are presented in Tables X through Y. For Phasor Blocks, we list each layer using the '**resolution**: layer configuration' format. The ResNet-based PsychoNet models use the same initial input embedding layer as ResNet (7×7 Conv2D and maxpooling) is used, while PsychoDW uses the same 4×4 patch embeddings as ConvNeXt-S. Figure A.2 presents further details, particular of the configurations of various convolutional blocks, for the Phasor Blocks shown in Figure 4. Interestingly, we found that using the initial layers of ResNet-50, instead of ConvNeXt-S, in our ConvNeXt-S based PsychoDW actually yielded better results (approx. $\uparrow 0.5\%$ top-1 accuracy on ImageNet-1K), so we chose to use it for the model. However, we do change all ConvBlocks in Phasor $(\mathbb{C})$ (see Figure A.2) to depthwise convolution blocks to maintain general faithfulness to the ConvNeXt model.

Finally, the companding operation we apply after taking the 2D FFT (in Figure 3) simply zeros the DC component and applies the element-wise function:

$$x \in \mathbb{C}, \quad \text{Compand} : x \to |x|^{\frac{1}{1.25}} \cdot \exp(i\angle x) \tag{5}$$

where $|x|$ denotes the magnitude of x and $\angle x$ its phase. Since the exponent applied to the magnitude is $\in (0, 1)$, this function compresses frequencies of large magnitude (i.e. frequencies very close to the DC component), and expands the magnitude of those further from it.

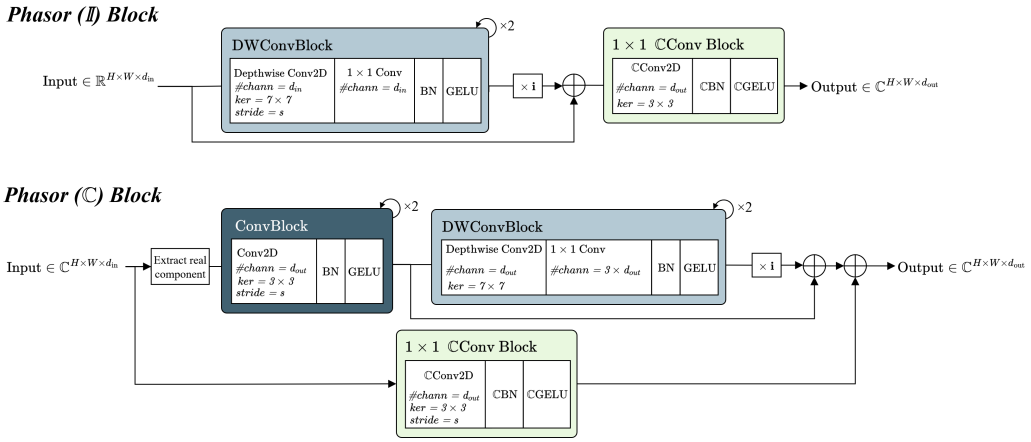

Figure A.2: Further architecture details for the Phasor Blocks presented in Figure 4. For ConvNeXt-based PsychoNet, we replace the two ConvBlocks at the start of Phasor $(\mathbb{C})$ blocks with two DWConvBlocks with the same number of channels. ℂConv/BN/GELU denote complex-valued convolution, batch norm and GELU operations - see Appendix A.3. The following PsychoNet architecture tables specify the values of d_{in}, $d_{\text{out}}$and stride (s) for all of their Phasor Blocks.

Table A.2: Detailed architecture of Psycho-S.

Psycho-S - based on ResNet-50	
Parameters (M)	25.35
# Layers (overall)	65
# Layers (complex)	9
Blocks	
Input layer	Conv2D(7×7, d_{in}=3, d_{out}=64, stride=2), MaxPool(3×3, stride=2)
Initial CNN layers	First 7 ResBlocks from ResNet-50 (first two resolution stages).
Phasor Blocks	$\mathbf{14 \times 14}$: $(\mathbb{I})$ [d_{in}=128, d_{out}=256, stride=2] $\mathbf{14 \times 14}$: $(\mathbb{C})$ [d_{in}=256, d_{out}=256] $\mathbf{14 \times 14}$: $(\mathbb{C})$ [d_{in}=256, d_{out}=384] $\mathbf{14 \times 14}$: $(\mathbb{C})$ [d_{in}=384, d_{out}=512] $\mathbf{14 \times 14}$: $(\mathbb{C})$ [d_{in}=512, d_{out}=512]
Spectral filters	Sub-bands ([crop, drop]): [14, 8], [8, 4], [4, 1], d_filter = 512
Output layer	Average pool, ComplexLinear(d_{in}=1536, d_{out}=1000), Softmax

Table A.3: Detailed architecture of Psycho-B.

Psycho-B architecture - based on ResNet-101	
Parameters (M)	42.01
# Layers (overall)	93
# Layers (complex)	13
Blocks	
Input layer	Conv2D(7×7, d_{in}=3, d_{out}=64, stride=2), MaxPool(3×3, stride=2)
Initial CNN layers	First 7 ResBlocks from ResNet-101 (first two resolution stages).
Phasor Blocks	$\mathbf{28 \times 28}$: ($\mathbb{I}$) [d_{in}=128, d_{out}=256] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=256, d_{out}=256] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=256, d_{out}=256] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=256, d_{out}=384] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=384, d_{out}=384, stride=2] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=384, d_{out}=384] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=384, d_{out}=512] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512]
Spectral filters	Sub-bands ([crop, drop]): [14, 8], [8, 4], [4, 1], d_filter = 512
Output layer	Average pool, ComplexLinear(d_{in}=1536, d_{out}=1000), Softmax

Table A.4: Detailed architecture of Psycho-L.

Psycho-L architecture - based on ResNet-152	
Parameters (M)	61.28
# Layers (overall)	93
# Layers (complex)	13
Blocks	
Input layer	Conv2D(7×7, d_{in}=3, d_{out}=64, stride=2), MaxPool(3×3, stride=2)
Initial CNN layers	First 7 ResBlocks from ResNet-152.
Phasor Blocks	$\mathbf{28 \times 28}$: ($\mathbb{I}$) [d_{in}=128, d_{out}=256] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=256, d_{out}=512] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512, stride=2] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512]
Spectral filters	Sub-bands ([crop, drop]): [14, 8], [8, 4], [4, 1], d_filter = 512
Output layer	Average pool, ComplexLinear(d_{in}=1536, d_{out}=1000), Softmax

Table A.5: Detailed architecture of Psycho-H.

Psycho-H architecture - based on ResNet-270	
Parameters (M)	88.61
# Layers (overall)	93
# Layers (complex)	13
Blocks	
Input layer	Conv2D(7×7, d_{in}=3, d_{out}=64, stride=2), MaxPool(3×3, stride=2)
Initial CNN layers	First 7 ResBlocks from ResNet-270.
Phasor Blocks	$\mathbf{28 \times 28}$: ($\mathbb{I}$) [d_{in}=128, d_{out}=256] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=256, d_{out}=512] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512, stride=2] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=640] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=640, d_{out}=1024]
Spectral filters	Sub-bands ([crop, drop]): [14, 8], [8, 4], [4, 1], d_filter = 1024
Output layer	Average pool, ComplexLinear(d_{in}=3072, d_{out}=1000), Softmax

Table A.6: Detailed architecture of PsychoDW.

PsychoDW architecture - based on ConvNeXt-S	
Parameters (M)	49.512
# Layers (overall)	109
# Layers (complex)	13
Blocks	
Input layer	Conv2D(7×7, d_{in}=3, d_{out}=64, stride=2), MaxPool(3×3, stride=2)
Initial CNN layers	First 7 ResBlocks from ResNet-50.
Phasor Blocks	$\mathbf{28 \times 28}$: ($\mathbb{I}$) [d_{in}=128, d_{out}=256] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=256, d_{out}=256] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=256, d_{out}=256] $\mathbf{28 \times 28}$: ($\mathbb{C}$) [d_{in}=256, d_{out}=512] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512, stride=2] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=1024] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=1024, d_{out}=1024] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=1024, d_{out}=1024] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=1024, d_{out}=1024]
Spectral filters	Sub-bands ([crop, drop]): [14, 8], [8, 4], [4, 1], d_filter = 1024
Output layer	Average pool, ComplexLinear(d_{in}=3072, d_{out}=1000), Softmax

C CLASSIFICATION EXPERIMENTS

In this section we present detailed results, dataset details and training recipes for all classification experiments conducted.

C.1 IMAGENET-1K

We use the standard large ImageNet-1K subset from (Deng et al., 2009) containing ~1.2 million training and ~50000 images for validation/testing. Table A.7 presents the training recipe used for ImageNet experiments.

Table A.7: ImageNet training recipe

Setting	Value
Image size	224×224
Epochs	90
Batch size (overall, not per GPU)	1024
Loss	Cross entropy
Optimizer	AdamW ($\beta_1 = 0.9, \beta_2 = 0.999$)
Scheduler	cosine
Initial learning rate (LR)	$5 \cdot 10^{-4}$
Warmup	warmup LR $= 10^{-6}$, 5 epochs
Learning rate decay	min. LR $= 10^{-5}$, 12 epochs
Augmentation	resize, crop, interpolate, horizontal flip, RandAugment, MixUp, CutMix, label smoothing
GPU	2× NVIDIA H100: Psycho-B, ResNet101, all 'Big' sized ablation models 2× AMD MI300X: Psycho-S, ResNet50 4× AMD MI300X: All other models

Table A.7 presents all ImageNet-1K experiment results. PsychoNet moderately improves top-1 accuracy for all ResNet baselines (↑ 0.82%, 0.41%, 0.26% and 0.44% vs. ResNet50 to 270), and incurs a small decrease for ConvNeXt-S (↓ 0.19). Figure A.3 compares SVC filters learnt by different ResNet-based PsychoNet sizes, showing that with larger model size, the filters become increasingly structured and sparser, with clearer frequency selectivity and reduced noise. Figure A.4 compares SVC filters learnt by Psycho-B on ImageNet-1K to the smaller resolution/size datasets in Appendix C.2. It is evident that increasing image resolution and dataset size both yield much sparser filters. These results suggest that the sparse patterns correspond to a data-driven representation naturally emergent from visual information.

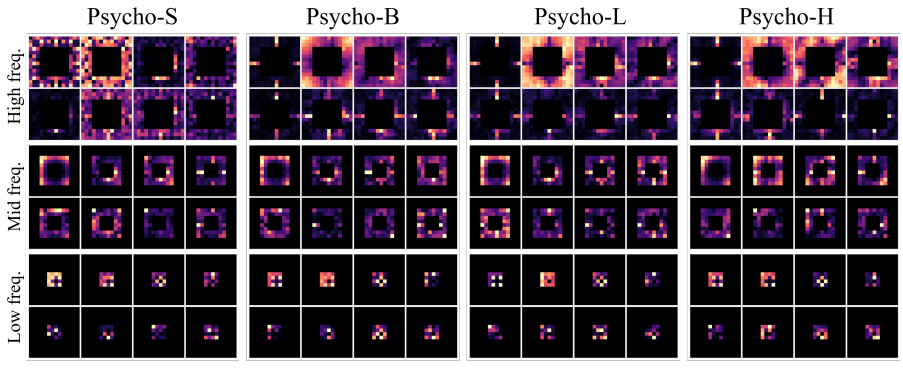

Figure A.3: Top principal components of SVC filters learnt by different sized ResNet-based PsychoNet models on ImageNet-1K. 'High/mid/low freq.' refer to the [14, 8], [8, 4] and [4, 1] frequency sub-bands created by Spectral Branches.

Table A.8: ImageNet-1K classification results. Each pair of rows (separated by horizontal lines) compares a baseline CNN and the PsychoNet based on it. FLOPs were measured using a single 224×224 input.

Model	Top-1 Acc. (%)	Top-5 Acc. (%)	Layers	Params (M)	FLOPs (G)	GPU
ResNet50	76.044	92.992	54	25.56	8.18	2× MI300X
Psycho-S	76.864	93.386	65	25.35	12.31	2× MI300X
ResNet101	78.428	94.220	105	44.55	15.60	2× H100
Psycho-B	78.846	94.600	93	42.01	30.13	2× H100
ResNet152	79.586	94.684	156	60.19	23.03	4× MI300X
Psycho-L	79.848	95.056	93	61.28	54.47	4× MI300X
ResNet270	80.012	95.088	276	89.60	40.50	4× MI300X
Psycho-H	80.454	95.290	93	88.61	64.12	4× MI300X
ConvNeXt-S	80.780	95.488	113	50.22	17.36	2× MI300X
Psycho-DW	80.590	95.384	106	49.51	27.42	2× MI300X

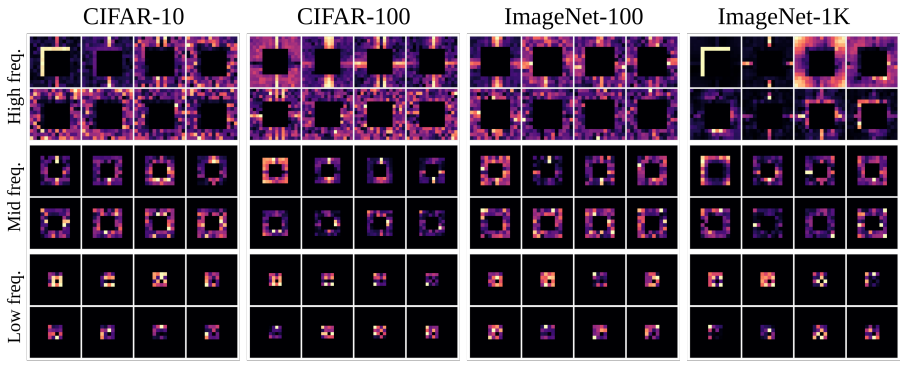

Figure A.4: Top principal components of SVC filters learnt by Psycho-B on different resolution and size datasets. 'High/mid/low freq.' refer to the [14, 8], [8, 4] and [4, 1] frequency sub-bands created by Spectral Branches.

C.2 Smaller classification datasets

Table A.9 presents experiment results for the CIFAR-10, CIFAR-100 and ImageNet-100 classification experiments.

CIFAR-10 is a small scale dataset comprising 50000 natural images for training and 10000 images for testing across 10 classes, at a resolution of 32×32 (Krizhevsky, 2009). For compatibility with this lower resolution (the ImageNet models have 224×224 input resolution), we reduce initial downsampling steps from our models. For ResNet and ResNet-based PsychoNet models, we removed the first maxpooling layer and set stride=1 for the first two ResBlocks that originally had stride=2. For ConvNeXt-S and PsychoDW, we replace the initial 4×4 patch embedding layer with a standard 3×3 Conv2D layer, and set stride=1 for the second downsampling layer. Table A.10 presents the training recipe for the CIFAR-10 experiments. Overall, all of our PsychoNet models outperformed their respective CNN baselines.

CIFAR-100 contains the same images and train-test split as CIFAR-10, but with labels reorganised into 100 classes instead of 10. We use the same model configurations and training recipe as CIFAR-10, but increase the number of epochs to 90 since the greater number of classes results in a harder classification problem. Table A.10 presents the training recipe for the CIFAR-10 experiments. Overall, all of our PsychoNet models outperformed their respective CNN baselines.

Module	Parameters (M)	# Layers	CIFAR-10	CIFAR-100	ImageNet-100
ResNet50	25.56	54	94.14	78.10	80.90
Psycho-S	25.35	65	95.08	78.97	82.50
ResNet101	44.55	105	93.64	79.13	81.90
Psycho-B	42.01	95	94.99	79.49	83.60
ResNet152	60.10	156	93.17	77.51	83.60
Psycho-L	61.28	93	94.95	79.64	84.82
ResNet270	89.60	276	76.51	50.87	83.80
Psycho-H	88.61	93	94.68	79.89	85.00
ConvNeXt-S	50.22	113	94.09	76.96	86.98
PsychoDW	49.51	106	95.46	79.67	86.76

Table A.9: Classification results (% top-1 accuracies) for CIFAR-10, CIFAR-100 and ImageNet-100. Each pair of rows (separated by horizontal lines) compares a baseline CNN and the PsychoNet based on it.

Table A.10: CIFAR-10 training recipe

Setting	Value
Image size	32×32
Epochs	35
Batch size	64
Loss	Cross entropy
Optimizer	AdamW ($\beta_1 = 0.9, \beta_2 = 0.999$)
Scheduler	OneCycle
Learning rate (LR)	10^{-3}
Augmentation	crop, horizontal flip
GPU	$1\times$ NVIDIA A100: Psycho-S/B, ResNet-50/101 $1\times$ NVIDIA H100: All other models

ImageNet-100 is a subset of the ImageNet dataset (Deng et al., 2009) that contains examples for 100 classes. It contains 130100 images for training and 5100 images for testing, at the original resolution of 224×224. The model architectures remain the same as the ImageNet experiments, but with the output linear layer modified to predict 100 logits. We use the same training recipe as ImageNet-1K (Table A.7), but reduce the batch size to 128. Psycho-S/B and ResNet50/101 were trained on $1\times$ NVIDIA A100, while all over models used $1\times$ AMD MI300X. Overall, the ResNet-based PsychoNet models outperformed their respective baselines, but PsychoDW fell slightly short of ConvNeXt-S.

D ABLATION STUDIES

We design four ablation model configurations to assess the impact of Phasor Blocks and Spectral Branches on classification performance and visual code quality.

The postfix SP (Single Phasor) indicates that we remove all Phasor Blocks ($\mathbb{C}$)s and make up for the resultant layer and parameter deficit by adding additional ResBlocks. SP_MB (Big/Large) were created by applying this modification to Psycho-B/L respectively, and in-depth architectural details of them are presented in Tables A.12 and A.13. The Single Branch (SB) models replace Spectral Branches with a single Hadamard Block with full band filters and no prior DropCrop operations. MP_SB (Big/Large) and SP_SB (Big/Large) were created by applying this modification to Psycho-B/L and SP_MB (Big/Large) respectively. Table A.11 and Figure A.5 present quantitative and qualitative results from this study.

Spectral Branches appreciably improve classification accuracy (0.45-0.58%), except for between Big size SP_MB and SP_SB. In Figure A.5, we visualize the first two spectral bands of the MB models, as well as the corresponding bands isolated from the full-band filters of the SB models. The former are sparse and highlight distinct frequencies, while the latter exhibit a similar structure but with significant noise. This suggests that the explicit spectral decomposition of Spectral Branches is important for generating clear visual codes. Multiple Phasor Blocks slightly improve classification accuracy (0.24-0.252%) for MB models and have little effect on SB ones. However, Figure A.5 shows that they drastically reduce noise and improve clarity of the SB filters, and moderately so for the MB ones. Finally, we also try removing the Phasor ($\mathbb{I}$) block from the Big size SP_MB model, yielding a model without any Phasor Blocks (no_phasor). This further reduces accuracy slightly, and results in the filters exhibiting conjugate symmetry as shown in Figure 6.

Note that as per He et al. (2016), ResBlocks each comprises 1×1, 3×3 and 1×1 kernel size Conv2D layers. In the below architecture tables, we denote their respective output channel sizes with d_{in}, d_{bot} and d_{out} respectively ('bot' is short for bottleneck, as these layers follow a channel bottleneck configuration). We also write 'stride=2' if a ResBlock performs $2\times$ spatial downsampling, since it is achieved by setting stride=2 in the 3×3 Conv2D layer.

Table A.11: **Ablation study results.** We compare all combinations of MB/SB and MP/SP model configurations, for Big and Large model sizes, using ImageNet top-1 accuracy.

Model	**M**ultiple Phasor **B**locks	**M**ultiple (Spectral) **B**ranches	Top-1 Acc. (%) (Psycho-B base)	Top-1 Acc. (%) (Psycho-L base)
MP_MB (Psycho-B/L)	✓	✓	78.846	79.848
MP_SB	✓	✗	78.394	79.268
SP_MB	✗	✓	78.600	79.596
SP_SB	✗	✗	78.548	79.124
no_phasor (SP_SB w/o Phasor ($\mathbb{I}$))	✗	✗	78.44	

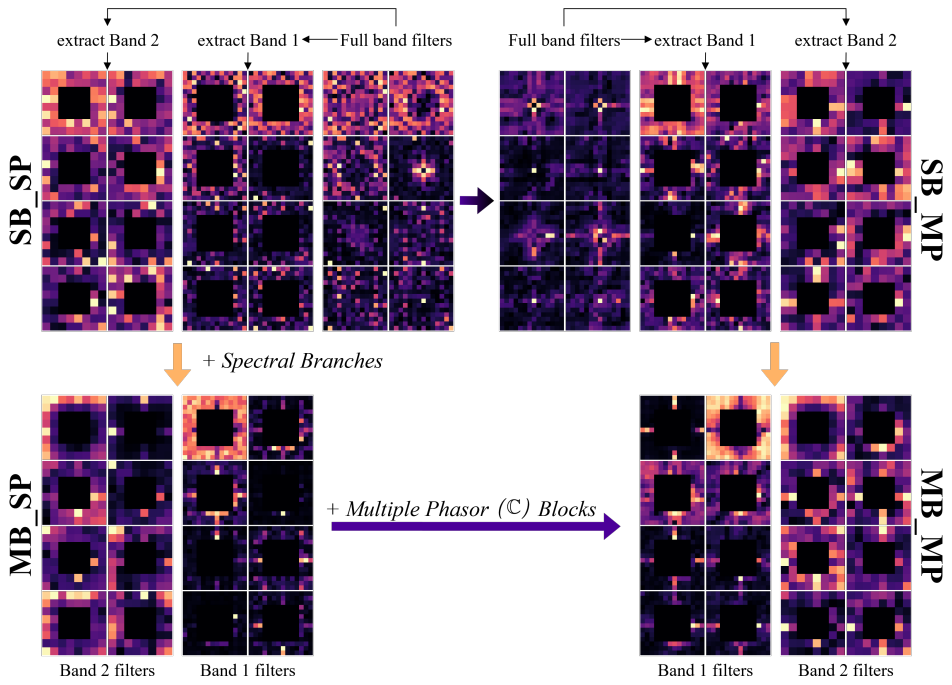

Figure A.5: Most significant channel-wise principal components of learnt spectral filters from Large size ablation models. We show the first two sub-bands of filters for MB models, and the full band filter for SB models.

Table A.12: Detailed architecture of the SP_MB (Big) ablation model.

SP_MB (Big) architecture	
Parameters (M)	42.263
# Layers (overall)	91
# Layers (complex)	2
Blocks	
Input layer	Conv2D(7×7, d_{in}=3, d_{out}=64, stride=2), MaxPool(3×3, stride=2)
ResBlocks	$\mathbf{56 \times 56}$: [d_{in}=64, d_{bot}=256, d_{out}=256] $\mathbf{56 \times 56}$: [d_{in}=256, d_{bot}=64, d_{out}=256] $\times$ 2 $\mathbf{28 \times 28}$: [d_{in}=256, d_{bot}=128, d_{out}=512, stride=2] $\mathbf{28 \times 28}$: [d_{in}=512, d_{bot}=128, d_{out}=512] $\times$ 7 $\mathbf{28 \times 28}$: [d_{in}=512, d_{bot}=256, d_{out}=1024] $\mathbf{28 \times 28}$: [d_{in}=1024, d_{bot}=256, d_{out}=1024] $\times$ 4 $\mathbf{14 \times 14}$: [d_{in}=1024, d_{bot}=256, d_{out}=1024, stride=2] $\mathbf{14 \times 14}$: [d_{in}=1024, d_{bot}=384, d_{out}=1536] $\mathbf{14 \times 14}$: [d_{in}=1536, d_{bot}=384, d_{out}=1536] $\times$ 6 $\mathbf{14 \times 14}$: [d_{in}=1536, d_{bot}=128, d_{out}=512]
Phasor Blocks	$\mathbf{14 \times 14}$: ($\mathbb{I}$) [d_{in}=512, d_{out}=512, stride=1]
SVC filters	Sub-bands ([crop, drop]): [14, 8], [8, 4], [4, 1], d_{filter} 512
Output layer	Average pool, ComplexLinear(d_{in}=1536, d_{out}=1000), Softmax

Table A.13: Detailed architecture of the SP_MB (Large) ablation model.

SP_MB (Large) architecture	
Parameters (M)	60.42
# Layers (overall)	90
# Layers (complex)	2
Blocks	
Input layer	Conv2D(7×7, d_{in}=3, d_{out}=64, stride=2), MaxPool(3×3, stride=2)
ResBlocks	$\mathbf{56 \times 56}$: $[d_{\text{in}}$=64, d_{bot}=256, d_{out}=256$]$ $\mathbf{56 \times 56}$: $[d_{\text{in}}$=256, d_{bot}=64, d_{out}=256$] \times 2$ $\mathbf{28 \times 28}$: $[d_{\text{in}}$=256, d_{bot}=128, d_{out}=512, stride=2$]$ $\mathbf{28 \times 28}$: $[d_{\text{in}}$=512, d_{bot}=128, d_{out}=512$] \times 5$ $\mathbf{28 \times 28}$: $[d_{\text{in}}$=512, d_{bot}=256, d_{out}=1024$]$ $\mathbf{28 \times 28}$: $[d_{\text{in}}$=1024, d_{bot}=256, d_{out}=1024$] \times 6$ $\mathbf{14 \times 14}$: $[d_{\text{in}}$=1024, d_{bot}=512, d_{out}=2048, stride=2$]$ $\mathbf{14 \times 14}$: $[d_{\text{in}}$=2048, d_{bot}=512, d_{out}=2048$] \times 7$ $\mathbf{14 \times 14}$: $[d_{\text{in}}$=2048, d_{bot}=128, d_{out}=512$]$
Phasor Blocks	$\mathbf{14 \times 14}$: $(\mathbb{I})$ $[d_{\text{in}}$=512, d_{out}=512, stride=1$]$
SVC filters	Sub-bands ([crop, drop]): [14, 8], [8, 4], [4, 1], d_{filter} 512
Output layer	Average pool, ComplexLinear(d_{in}=1536, d_{out}=1000), Softmax