DEEP PSYCHOVISUAL IMAGE REPRESENTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Psychovisual models suggest human vision decouples low-level feature extraction from higher cognition by first forming intermediate abstractions. In contrast, deep learning-based vision models routinely extract and aggregate features using homogeneous stacks of spatial layers, rendering their decision-making processes opaque. In this paper, we propose Deep Visual Coding (DVC), a learned frequency-domain representation inspired by 1990s image codes that quantised perceptually salient frequencies, which together with complex-valued image representations produces psychovisual-style abstractions. This approach enables the first psychovisual-based deep learning framework, utilizing data-driven spectral filters that learn to encode task-relevant semantic structures within distinct frequency sub-bands. Salience analyses reveal that our psychovisual models extract highly interpretable object parts compared to the amorphous regions produced by regular Convolutional Neural Networks (CNNs). Furthermore, we find that our models are less depth dependent than CNNs for model scaling, since our complex-valued representations and learned abstractions subsume the role of deep spatial layers. Together, these findings demonstrate that psychovisual coding provides a promising path toward more efficient and transparent vision models.

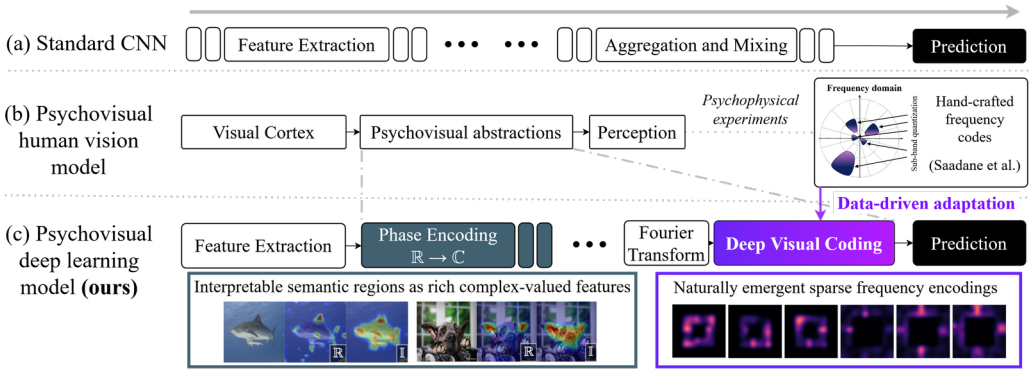

Figure 1: (a) Standard CNNs use deep stacks of homogeneous spatial layers. (b) **Psychovisual** models suggest human vision uses explicit intermediate abstractions. (c) Our psychovisual deep learning framework first produces rich complex-valued representations of interpretable semantic regions. They are then encoded in the frequency domain by **Deep Visual Coding (DVC)**, a data-driven adaptation of hand-crafted psychovisual coding that introduces psychovisual-like abstractions into deep learning models.

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). These models refine and aggregate image features through deep, homogeneous stacks of spatial layers, where feature processing emerges implicitly

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

through training (Figure 1 (a)). As a result, intermediate representations are typically produced ad hoc with little explicit structure or organisation. Psychovisual processing (Figure 1 (b)), the way human vision encodes and interprets visual information, separates feature extraction from higher cognition using intermediate abstractions, such as 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 the 1990s, psychovisual coding schemes were developed that targeted the perceptually salient frequencies found by human vision studies (Saadane et al., 1994; Guedon et al., 1995; Saadane et al., 1998) (see Appendix B.1 for a brief introduction). We hypothesize that these perceptually salient frequency selections may implicitly capture the high-level semantic abstractions utilized by human cognition.

In this work, we develop a data-driven adaptation of psychovisual coding that introduces abstraction-like representations into deep learning frameworks (Figure 1 (c)). To the best of our knowledge, our work represents the first data-driven exploration of the frequency domain for high-level representation learning in vision, whereas previous studies primarily focused on lower-level feature learning or parameterizing spatial models (Chi et al., 2020; Rippel et al., 2015; Yadav & Jerripothula, 2023). Furthermore, our approach processes feature abstractions almost entirely in the frequency domain without the need for repeated spatial-to-spectral transformations. Our key contributions are:

- We propose **Deep Visual Coding (DVC)**, a first-of-its-kind data-driven adaptation of psychovisual coding schemes designed to introduce human-vision-like abstractions into deep learning models. It employs learnable, band-limited frequency filters that learn representations of high-level semantic image information, emerging as sparse selections of coronal frequency sub-bands in the Discrete Fourier Transform (DFT).
- We introduce **Phasor Block** which has the ability to learn complex-valued features as phase information from real-valued coloured signals to build complex-valued image features that are well suited to the Fourier domain and the proposed DVC.
- We assemble DVC and Phasor Blocks into a cohesive psychovisual deep learning framework. We demonstrate that this pipeline enables interpretable psychovisual-like processing, maintaining or improving performance across multiple classification benchmarks.
- We provide salience analysis that demonstrate evidence of semantic decision making that reveal intermediate spatial layers consistently focus on meaningful object parts. These features are then encoded by DVC and used for final output prediction, mirroring the use of abstractions to support reasoning in a psychovisual model of human vision.

2 BACKGROUND

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 (RFs) - 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 vision, 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 Convolutional Neural Network (CNN) kernels also perform similar directional operations (Krizhevsky et al., 2017; Rippel et al., 2015). However, networks that directly incorporate these functions, for example as fixed filter banks or bases for convolution kernels, have generally been restricted to small-scale problems and do not scale well to modern vision tasks (Liu et al., 2021). Beyond RFs, human vision has also been studied from a frequency domain perspective in previous image coding literature. In particular, we build on psychovisual coding (introduced below) to model higher-level visual abstractions in a form that scales to modern tasks.

Frequency Domain Learning. Deep learning computer vision models have largely focused on processing in the spatial domain, which expresses localized relationships through contiguous pixel neighborhoods. The frequency (Fourier) domain, conversely, projects signals onto orthogonal basis functions, inherently capturing global representations (see Appendix B.2). Formulated in this space, traditional 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 strong resemblances to psychovisual codes since they target specific selections of sub-bands, corresponding to features from different spatial scales.

In deep learning, the frequency domain has primarily been used to exploit the Convolution Theorem (Gonzalez & Woods, 2008), whereby spatial circular convolution becomes simple element-wise multiplication in the frequency domain. Many works leveraging this property are performance-driven: Li et al. (2020a); Chi et al. (2020); Guan et al. (2021) use frequency domain filters to accelerate CNNs and incorporate global context, while Rao et al. (2023); Lee-Thorp et al. (2021); Huang et al. (2023) employ global frequency filters as lightweight and effective token mixers for transformer-style models. Other studies use frequency domain re-parameterizations of CNNs to analyse model properties and behaviours (Rippel et al., 2015; Grabinski et al., 2023; Kabri et al., 2023), such as optimal kernel structures. However, while there is extensive work exploiting the frequency domain for computational efficiency and model analysis, it has remained largely unexplored as an explicit representation space in its own right, particularly for high-level semantic features.

Psychovisual Coding. Research conducted in the 1990s by Dominique Barba and colleagues used psychophysical experiments to characterise the frequency sensitivities of the human visual cortex (Saadane et al., 1994; Senane et al., 1995; Saadane et al., 1998). These findings informed *psychovisual coding*, an image quantization and compression scheme designed to be perceptually lossless. This approach exploits the human visual system's varying sensitivity to spatial frequencies by decomposing the frequency domain into a series of coronal sub-bands (Figure 2 (left)), each assigned quantization thresholds based on measured sensitivities. This process enables frequencies corresponding to perceptually salient image features to be preserved in full, while others are removed or heavily compressed without affecting perceived image quality (see Appendix B.1).

While psychovisual coding was developed for image compression, the psychophysical experiments behind it establish that a sparse selection of frequency sub-bands retains the information sufficient for an observer to understand an image, even when most of the spectrum is discarded. What we take from psychovisual coding is its mechanism: rather than hand-crafting such a selection on raw pixels for image compression, we ask whether an analogous selection can be learned from data, applied after spatial feature extraction, and optimised for recognition. The frequency domain also has established connections to receptive fields and early visual processing [1-4], which is a further reason to explore whether it can also support the higher-level representations that follow. In this work, we provide evidence that a psychovisual coding-style mechanism, learned in a data-driven setting, can serve as a functional proxy for introducing psychovisual-like abstraction into vision models, without directly modelling any neural mechanism. To the best of our knowledge, this connection has not yet been explored in deep learning, unlike receptive fields.

3 METHOD

Deep Visual Coding (DVC). Shown in Figure 2 (right), DVC is a learnable frequency filtering module modelled after the coronal sub-band design of psychovisual coding, intended to realise the psychovisual abstraction step at the core of our approach.

Let $\boldsymbol{x} \in \mathbb{C}^{d \times w \times w}$ be a spatial feature map produced by a neural network, where d is the number of channels and $w \times w$ the spatial size. Its frequency spectrum $\boldsymbol{X} \in \mathbb{C}^{d \times w \times w}$ can be obtained via applying the 2D DFT (Cooley et al., 1969) on the last two dimensions. The *Spectral Branches* module replicates the radial frequency partitioning in psychovisual codes, decomposing $\boldsymbol{X}$ into disjoint sub-bands $\boldsymbol{X}_1, \boldsymbol{X}_2, ...$ As the DFT samples the spectrum on a rectangular grid, we use nested square sub-bands for efficient sub-band extraction via simple tensor slicing. Each sub-band is produced via applying *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 $d \times \text{crop}_i \times \text{crop}_i$. For each sub-band $\boldsymbol{X}_i$, we then apply a set of learnable filters $\boldsymbol{W_i} \in \mathbb{C}^{d \times \text{crop}_i \times \text{crop}_i}$ via:

$$\boldsymbol{X}_{i_{\text{filtered}}} = \text{Softmax}\left(\boldsymbol{X}_i \odot \boldsymbol{W}_i\right) \tag{1}$$

where $\odot$ denotes the Hadamard (element-wise) product. Channel-wise softmax is applied here to amplify important frequency selections and suppress unimportant ones - emulating the quantization step of psychovisual coding. This filtering step is encapsulated in a *Hadamard Block* which additionally applies normalization and channel-mixing via a 1×1 convolution layer.

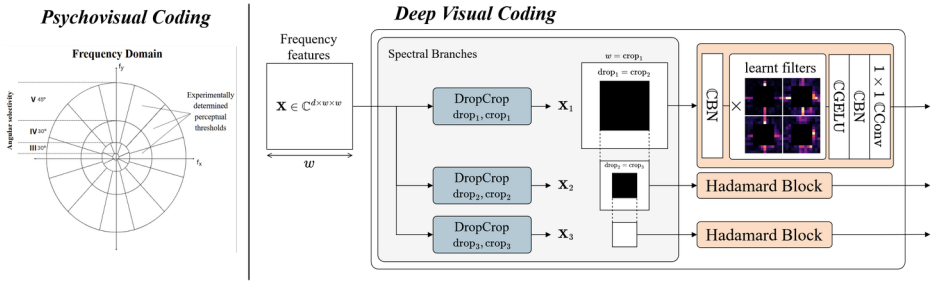

Figure 2: **(Left) Hand-crafted psychovisual coding** from Saadane et al. (1998), which quantizes perceptually salient radial frequencies determined by human vision experiments. **(Right) Our DVC module,** a data-driven adaptation of psychovisual coding. It uses (1) *Spectral Branches* for radial spectral decomposition (2) *Hadamard Blocks* to apply learnt element-wise filters and channel mixing. ℂConv/BN/GELU: complex-valued convolution, batch norm and GELU (Appendix D.2).

Phasor Blocks. To prepare spatial features for DVC, we also introduce Phasor Blocks (Figure 3 (top)) to first convert real-valued image features into rich complex-valued representations. As demonstrated in our experiments (Section 4.1), this transformation produces activations that target interpretable semantic regions, such as characteristic object parts, which provide the part-level features that DVC subsequently encodes into psychovisual abstractions. An additional practical motivation for this design is to avoid conjugate symmetry. Real-valued features, such as natural images, incur the conjugate symmetry of the Fourier Transform (FT), which renders half of the frequency domain redundant and limits learnt filtering from fully exploiting complex representations. By generating complementary imaginary components, Phasor Blocks break conjugate symmetry and improve the specificity of learned sub-bands.

In practice, imaginary components are generated from existing real features using lightweight depthwise convolution based blocks which decouple spatial and channel mixing to encourage cross-channel interaction without altering spatial structure. In natural complex signals, real and imaginary components convey complementary information at the same spatial location (Gonzalez & Woods, 2008; Lee et al., 2022), making it important that the generated imaginary features do not introduce substantial new spatial information. Two Phasor Block configurations are used in this work: Phasor $(\mathbb{I})$ performs the initial real-to-complex conversion, while Phasor $(\mathbb{C})$ further refines complex-valued features (Figure 3 (top), Appendix D.3).

Psychovisual Framework. To experimentally evaluate DVC and Phasor Blocks in Section 4, we combine them into a cohesive deep learning framework designed to explicitly mimic psychovisual processing. We emphasize that this framework is not intended as a definitive architecture for achieving state-of-the-art performance, but rather as a proof-of-concept for psychovisually-grounded deep learning. We refer to models utilising this pipeline collectively as PsychoNet.

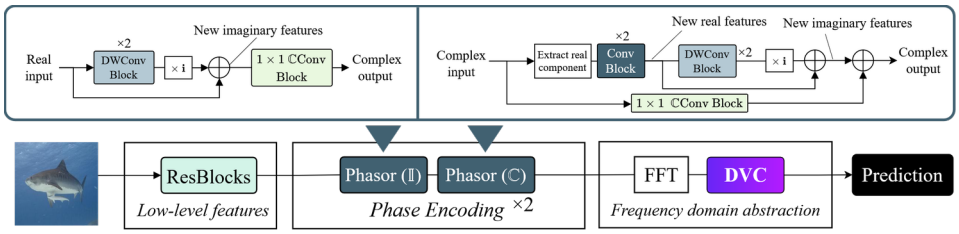

Figure 3: Overall design of the PsychoNet framework (bottom) and Phasor Block architectures (top). ResNet-style Residual Blocks (ResBlocks) first extract low-level features, then are converted into complex-valued representations by Phasor Blocks and encoded in the frequency domain by DVC.

As illustrated in Figure 3, PsychoNet operates as a coherent dual-domain pipeline comprising the following stages:

1. Standard ResNet-style Residual Blocks (ResBlocks) perform initial spatial feature extraction.

2. Phasor Blocks convert real-valued spatial features into complex-valued representations.
3. 2D Fast Fourier Transform (FFT) is applied to convert the features from the spatial to the frequency domain. As the magnitude of the DC (zero frequency) and low-frequency components typically dominate visual data, a simple companding operation is also applied to balance the spectrum (see Appendix D).
4. DVC is applied, and its outputs from each frequency band aggregated. For classification, these are further pooled spatially and used for output prediction directly in the frequency domain using a complex-valued linear layer.

Mirroring psychovisual processing, stages 1–2 roughly parallel early cortical feature extraction, while stages 3–4 realise the abstraction step that decouples those features from the high-level representations used for decision-making.

4 EXPERIMENTS

We evaluate our proposed psychovisual framework using natural image classification as a straightforward, scalable, and well-understood testbed to investigate whether it learns the abstraction-like representations our framing predicts. In the following subsections, we first present visualisations exploring the representations learnt by the core components of PsychoNet. We then use various classification benchmarks to explore its scaling properties and demonstrate it achieves reasonable practical performance, before finally discussing ablation studies.

Implementation details. Our PsychoNet models retain the first two spatial stages of a standard ResNet backbone and replace the remaining deeper stages with our Phasor Blocks and DVC module. We evaluate these models across small to large-scale datasets, ranging from the low-resolution CIFAR-10/100 (Krizhevsky, 2009) (∼50K images) to the moderate-sized ImageNet-100 (∼130K images) and the standard large-scale ImageNet-1K (∼1.2 million images) (Deng et al., 2009). We trained both our own and comparison models from scratch using identical recipes; on ImageNet-1K we train for 90 epochs, use an AdamW optimizer with a cosine scheduler and standard augmentations. Full details are provided in Appendices C and D.

4.1 REPRESENTATION ANALYSIS

In this section, we analyse the internal representations learned by the primary components of our framework. We first investigate what our DVC module and Phasor Blocks learn individually, followed by an exploration of their interaction to reveal how DVC encodes spatial features extracted by Phasor Blocks. For all visualisations, unless stated otherwise, we utilize a model based on Psycho-B. This model retains the first two ResBlock stages of a ResNet backbone but replaces the remaining stages with Phasor Blocks and an DVC module that match the original depth and parameter count.

DVC filter learning. Figure 4 visualises filters learnt by DVC, showing the top spatial principal components as an approximation of the most important frequency features. Remarkably we find that with sufficient training corpora and the full PsychoNet framework, DVC learns structured and sparse selections of frequencies that are highly reminiscent of the original hand-crafted psychovisual coding scheme. We further investigated how the amount of training data and architectural components affect the quality of these learnt patterns, finding that sparsity requires sufficiently large training corpora: ImageNet-100-trained filters are noticeably noisier than those for ImageNet-1K, and filters trained on CIFAR-10/100 (Figure A.3) are noisier still. Additionally, we also find that both Spectral Branches (providing sub-band decomposition) and Phasor Blocks (providing complex-valued features) are important; removing either significantly reduces filter sparsity and expressiveness. Together, these results suggest a naturally emergent psychovisual-style abstraction in the frequency domain, given sufficient data and complex-valued sub-bands. Our further analysis below of their interaction with Phasor Blocks reveals these sparse patterns correspond to structured encodings of meaningful object parts.

Phasor Block salience. We employ KPCA-CAM (Karmani et al., 2024) for salience mapping to visualise the features extracted by Phasor Blocks. Figure 5 shows that Phasor Blocks particularly

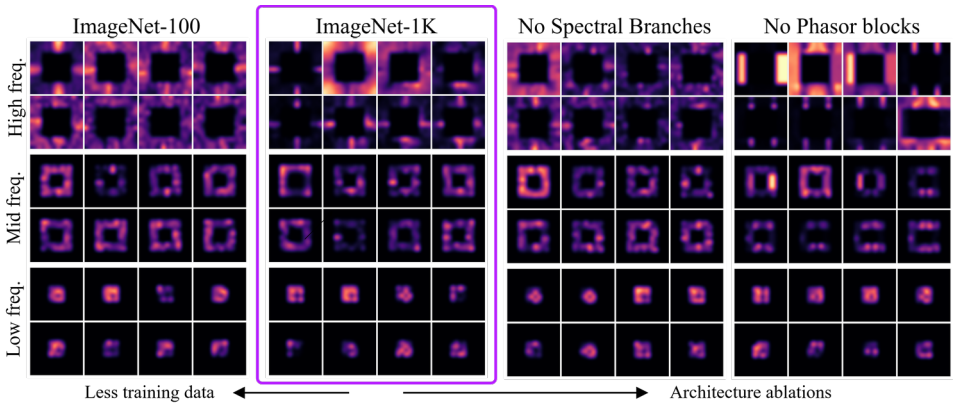

Figure 4: DVC filters learnt by Psycho-B trained on ImageNet-100 and ImageNet-1K, as well as for two ablation models on ImageNet-1K. Bilinearly smoothed. 'High/mid/low freq.' refer to the [14, 8], [8, 4] and [4, 1] frequency sub-bands created by Spectral Branches. 'No Spectral Branches' removes Spectral 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.

specialise towards localising characteristic morphological object parts of different classes, such as dog ears, elephant tusks and car wheels. Since KPCA-CAM only uses activations of the visualised layer and is uninfluenced by model predictions (e.g. via backpropagation in gradient-based CAMs), these results indicate that Phasor Blocks specialise to extract meaningful semantic object parts. This organisation is likely shaped by the presence of DVC downstream, which we show below encodes these parts into higher-level abstractions that support interpretable semantic reasoning. Additionally, it appears that the imaginary components of Phasor Block activations capture more global features than the real components (e.g. a dog's face vs. its ears), suggesting a structured and rich utilisation of the complex domain. An initial clustering visualisation of Phasor Block activations is also presented in Figure A.7, which finds that observable clustering emerges in both components and becomes increasingly pronounced at deeper layers. These findings further demonstrate that this complex-valued representation is iteratively refined through each block.

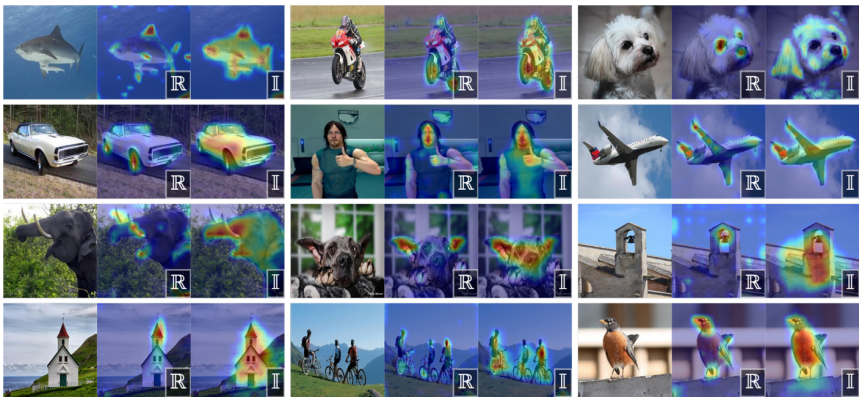

Figure 5: Assorted activation maps (via KPCA-CAM) for mid-level Phasor Blocks of Psycho-B. Real and imaginary components are denoted by $\mathbb{R}$ and $\mathbb{I}$.

DVC and Phasor Block interaction. We use a modified version of HiResCAM (Draelos & Carin, 2020) to explore how DVC uses the semantic features produced by Phasor Blocks. Normal HiResCAM produces salience maps by element-wise multiplying layer activations with gradients backpropogated from model predictions, so the salience regions have a high contribution to the final classification prediction. We extend this approach to isolate regions used by specific branches and filters of DVC by selectively masking (setting to zero) gradient contributions from the other components. 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 sub-band, Phasor Blocks' salience regions reveal that DVC distributes object parts by scale. Figure 6 (a) shows that the low-frequency sub-band focuses on subjects broadly, while mid-high frequencies isolate more specific parts of different sizes. This aligns with frequency domain theory, in which low frequencies capture coarse spatial structure and higher frequencies finer detail and edges, supporting the view that DVC performs structured filtering in the frequency domain. 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 6 (b)).

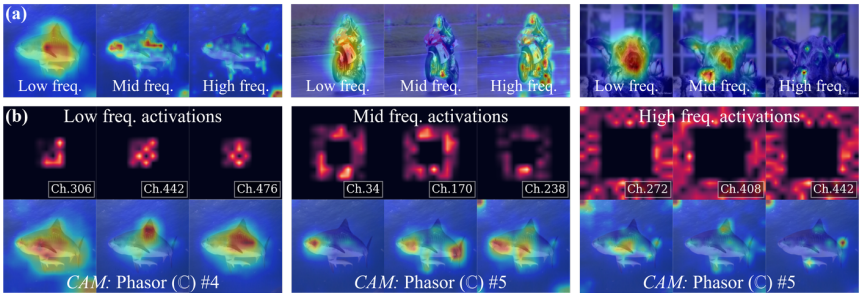

Figure 6: Psycho-B Phasor Block salience maps (via HiResCAM) conditioned on gradients **(a)** from individual Spectral Branch sub-bands and **(b)** from individual frequency domain feature channels.

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

4.2 Quantitative Evaluation

Unlike ResNet architectures which rely on increasing depth to scale representational capacity (He et al., 2016), we hypothesize that because PsychoNet's high-level processing is handled by frequency domain modules, it should remain relatively depth-independent. To test this, we designed Psycho-S/B/L/H to match four ResNet parameter sizes while sharing the same early spatial layers; however, rather than adding depth, we scale representational capacity by increasing the channel width of existing Phasor Blocks and DVC filters. As the base PsychoNet models only downsample features to 14×14 resolution at the smallest in order to maintain spectral fidelity for DVC, they use higher FLOPs than ResNet which downsamples further to 7×7. As such, we also created three efficient PsychoNet variants, Psycho-Eff-S/B/L, which do downsample to 7×7 and extract frequency features in the 7 to 14 sub-band from an earlier layer. These have similar FLOPs with ResNet at the cost of a small performance compromise compared to the normal PsychoNet models. However, as complex-valued operations currently lack the hardware and framework optimisations available to real-valued ones, they still run slower in practice than ResNet (Appendix C.3). Table 1 summarises the results from all classification experiments.

Overall, PsychoNet demonstrates significantly less dependence on depth for scaling than traditional ResNet architectures, as illustrated in Figure 7(a). This is evidenced by Psycho-Eff-B and Psycho-Eff-L, which use 38% and 45.5% fewer layers than ResNet-101 and ResNet-152 while closely matching their parameters, FLOP counts, and performance. Similarly, Psycho-L and Psycho-H utilize $\sim 1.7\times$ and $\sim 3\times$ fewer layers than ResNet-152 and ResNet-270 respectively, while achieving improved accuracy across every dataset. This demonstrates that DVC is able to subsume the role of a significant portion of deep spatial layers, showing it is an effective high-level processing module. This aligns with the psychovisual framing that motivated our design: DVC operates on globally-focused frequency-domain abstractions, whereas spatial features are sparse and locally structured, requiring significant depth to progressively aggregate them. Furthermore, PsychoDW achieves performance comparable to ConvNeXt-S, demonstrating that PsychoNet can also extend to match more modern architectures. Figure 7(b) provides complementary qualitative evidence for this: unlike ResNet, which produces diffuse, unstructured activations throughout its depth, PsychoNet's spatial layers

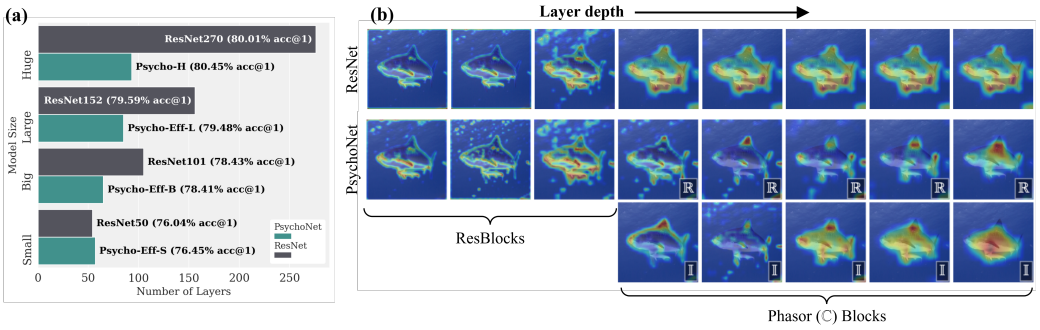

Figure 7: **(a)** Comparison of layer depth between different ResNet models and our shallowest PsychoNet model of comparable size. **(b)** Comparison between activation maps (via KPCA-CAM) of Psycho-B and ResNet101 for a range of layer depths.

Table 1: Summary of classification results (% top-1 accuracies) on CIFAR-10 (C10), CIFAR-100 (C100), ImageNet-100 (IN100) and ImageNet-1K (IN1K). Each group compares a baseline CNN to PsychoNet model(s) of comparable size; arrows show the difference in depth relative to the baseline.

Model	Param. (M)	Layers	GFLOPs	C10	C100	IN100	IN1K
ResNet50	25.56	**54**	8.18	94.14	78.10	80.90	76.04
Psycho-S	25.35	65 (↑20%)	12.31	**95.08**	**78.97**	82.50	**76.86**
Psycho-Eff-S	26.27	57 (↑6%)	8.01	94.86	78.15	**83.20**	76.45
ResNet101	44.55	105	15.60	93.64	79.13	81.90	78.43
Psycho-B	42.01	93 (↓11%)	30.13	**94.99**	**79.49**	83.60	**78.85**
Psycho-Eff-B	45.82	**65** (↓38%)	15.80	94.98	78.22	**83.92**	78.41
ResNet152	60.10	156	23.03	93.17	77.51	83.60	79.59
Psycho-L	61.28	93 (↓40%)	54.47	**94.95**	**79.64**	84.82	**79.85**
Psycho-Eff-L	62.03	**85** (↓46%)	23.06	**94.95**	78.18	**84.98**	79.48
ResNet270	89.60	276	40.50	76.51	50.87	83.80	80.01
Psycho-H	88.61	**93** (↓66%)	64.12	**94.68**	**79.89**	**85.00**	**80.45**
ConvNeXt-S	50.22	113	17.36	94.09	76.96	**86.98**	**80.78**
PsychoDW	49.51	**109** (↓4%)	27.42	**95.46**	**79.67**	86.76	80.59

consistently localise distinct morphological object parts, suggesting that Phasor Blocks and DVC together successfully take over the high-level processing otherwise requiring many additional spatial layers. To quantify this part localisation, we use ImageNet-S50 (Gao et al., 2023), which provides ground-truth segmentation masks for a subset of ImageNet, and define a Distinct Part Rate (DPR): the fraction of images whose salience map contains at least two topographically distinct peaks lying within the object mask (Appendix C.4). Psycho-B achieves a substantially higher DPR than ResNet-101 at three consecutive mid-to-late blocks (0.47–0.61 vs. 0.25–0.32, $p < 10^{-27}$), with the largest gains for animals and vehicles, whose parts are visually distinct.

4.3 Ablation Studies

We conduct two ablation studies to quantify the contribution of PsychoNet's key components: (1) minimal models (∼3M parameters) on CIFAR-10 and ImageNet-100, examining the isolated effects of DVC and Phasor Blocks (Table 2), and (2) Psycho-L on ImageNet-1K, targeting the components that our qualitative results suggested are particularly important at scale — Spectral Branching, which visibly improved DVC filter sparsity and expressivity, and full Phasor Blocks, which produced compelling interpretable salience maps on large-scale data (Table 3).

For the minimal ablation, models removing DVC pass Fast Fourier Transform (FFT) features directly into the classification layers, while models removing Phasor Blocks replace them with basic Conv.

Table 2: **Barebones architecture ablation** on CIFAR-10 and ImageNet-100 (IN100).

Model	DVC	Phasor Blocks	Top-1 Acc. (%) CIFAR-10	IN100
I	✓	✓	94.03	70.66
II	✗	✓	92.86	66.90
III	✓	✗	94.20	73.64
IV	✗	✗	93.33	69.04

Table 3: **ImageNet-1K ablation** on Psycho-L (Model A) and variants.

Model	Spectral Branch	Phasor Blocks	Top-1 Acc. (%)
A	✓	✓	79.85
B	✗	✓	79.27
C	✓	✗	79.60
D	✗	✗	79.12

Blocks (see Appendix E), at approximately matched parameter count and model depth. Table 2 shows that removing DVC drops performance on CIFAR-10 by 1.17% (Model I vs. II) and 0.87% (Model III vs. IV), and on ImageNet-100 by 3.76% and 4.60% respectively, highlighting its contribution to the framework. However, at this small scale, substituting Phasor Blocks for Conv. Blocks improves accuracy on both datasets, and by a wider margin on ImageNet-100. We attribute this to model capacity rather than task difficulty: Phasor Blocks are overly complex for a ∼3M parameter backbone. The effect reverses at scale, where Psycho-L benefits from full Phasor Blocks (Table 3, A vs. C).

For the ImageNet-1K ablation, we evaluate variations of the Psycho-L architecture (Table 3). Models without Spectral Branches replace the multiple branches in DVC with a single global filter lacking DropCrop operations. Models without Phasor Blocks replace all but one final Phasor (𝕀) block with ResBlocks while maintaining parameter size. Removing Spectral Branches reduces accuracy by 0.58% (A vs. B) and 0.47% (C vs. D), showing that this spectral decomposition improves performance as well as filter sparsity. The removal of Phasor Blocks results in smaller drops of 0.25% (A vs. C) and 0.14% (B vs. D). However, consistent with our earlier qualitative findings, the value of Phasor Blocks lies not in classification accuracy but in representational quality: as shown in Figure 8, Model A produces activation maps capturing meaningful object parts, whereas Model C yields comparatively diffuse activations lacking clear structure. Together, these ablations confirm distinct roles within the psychovisual pipeline: Spectral Branches structure DVC's abstractions, while Phasor Blocks preserve the part-level features feeding them.

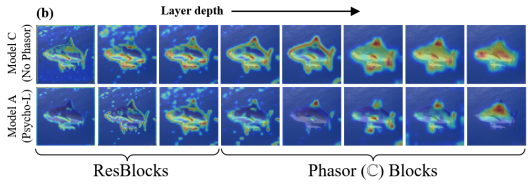

Figure 8: Comparison between activation maps of Ablation Model A and C.

4.4 Limitations and Future Work

While we show DVC organises and encodes selections of meaningful object parts, further work is required to determine how the deeper semantic meaning of these abstractions should be interpreted in relation to broader notions of reasoning, as studied in neuroscience (Quiroga et al., 2005; Kriegeskorte et al., 2008). We also aim to apply PsychoNet beyond classification, particularly to dense prediction, where DVC could utilise wider frequency ranges. Finally, aliasing is known to afflict standard CNN architectures (Grabinski et al., 2022); future work should assess its impact on our representations.

5 Conclusion

We proposed Deep Visual Coding, a data-driven adaptation of psychovisual coding. Integrated into our PsychoNet framework, DVC learns sparse coronal frequency patterns that closely resemble classical hand-crafted codes, suggesting a natural representation scheme for psychovisual abstraction. Unlike conventional CNNs, PsychoNet enforces a clear structural separation of stages: Phasor Blocks isolate semantically meaningful object parts, which DVC then organizes into interpretable, frequency domain representations for decision-making. Our results also show that DVC subsumes the high-level processing role of deep spatial layers, matching or exceeding ResNet baselines with far fewer layers. Together, these findings suggest that psychovisual coding is a principled basis for interpretable representation learning and for models whose internal reasoning is transparent by design.

AI USE STATEMENT

In this work, we used generative AI tools for the following tasks with required disclosure:

- Implementing methods. AI-assisted coding tools were used to implement and debug parts of the experiment code.
- Providing feedback on research methodology or experiments. When designing evaluation metrics and visualisation techniques, AI tools were used to review and compare candidate approaches.

Additionally, we used generative AI tools for the following tasks with recommended disclosure:

- Polishing and rewording text, and proofreading for grammar and spelling.
- Assisting with the literature review, including identifying and summarising related work.
- Improving the LaTeX formatting of the paper.

We did not use generative AI tools for the following tasks with required disclosure:

- Developing theoretical models or conceptual frameworks.
- Proposing or refining hypotheses.
- Cleaning or reformatting datasets.
- Supporting qualitative or thematic data analysis.
- Interpreting results.

The remaining tasks with required disclosure are not applicable to this work:

- Generating synthetic datasets.
- Formulating mathematical claims.
- Providing critical ingredients for proving mathematical claims.
- Assisting in the writing of proofs.
- Assisting with translation.

We have reviewed all AI-assisted work: all authors proofread the complete text, and all AI-generated code was manually tested and verified. We take full responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

All authors have reviewed the ICLR 2027 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 CIFAR and 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 E. Detailed model configurations are presented in Appendix D. The supplementary material includes code for training and evaluating our ResNet-based PsychoNet models (Psycho-S/B/L/H and Psycho-Eff-S/B/L) and ResNet baselines on ImageNet-1K, along with their model configurations. After deanonymization, we will release an extended code repository covering the full set of experiments and ablations, as well as model weights.

REFERENCES

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: 10.1098/rsta.1999.0444.

S. S. Chandra, N. Normand, A. Kingston, J. Guédon, and I. Svalbe. Robust Digital Image Reconstruction via the Discrete Fourier Slice Theorem. *Signal Processing Letters, IEEE*, 21(6):682–686, June 2014. ISSN 1070-9908. doi: 10.1109/LSP.2014.2313341.

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: 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 Ahmad Dedmari, Sailesh Conjeti, Santiago Estrada, Phillip Ehses, Tony Stöcker, and Martin Reuter. Complex fully convolutional neural networks for MR image reconstruction. In *Machine Learning for Medical Image Reconstruction (MLMIR 2018)*, volume 11074 of *Lecture Notes in Computer Science*, pp. 30–38. Springer International Publishing, 2018. doi: 10.1007/978-3-030-00129-2_4.

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.

Shanghua Gao, Zhong-Yu Li, Ming-Hsuan Yang, Ming-Ming Cheng, Junwei Han, and Philip Torr. Large-scale unsupervised semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45 (6):7457–7476, 2023.

Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Pearson Prentice Hall, Upper Saddle River, NJ, 3rd edition, 2008.

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.

Jeanpierre V. Guedon, Dominique Barba, and Nicole Burger. Psychovisual image coding via an exact discrete Radon transform. In *Proceedings Volume 2501, Visual Communications and Image Processing 1995*, volume 2501, pp. 562–572, 1995. doi: 10.1117/12.206765.

Jean-Pierre V. Guédon, Nicolas Normand, Pierre Verbert, Benoit Parrein, and Florent Autrusseau. Load-balancing and scalable multimedia distribution using the Mojette transform. *Internet Multimedia Management Systems II*, 4519(1):226–234, 2001.

Jeanpierre Guédon. *The Mojette Transform: Theory and Applications*. John Wiley & Sons, March 2013. ISBN 9781118622933.

Jose Hanen and Dominique Barba. High-quality subband image coding of TV signals at 5 Mbit/s with motion compensation interpolation and visually optimized scalar quantization. In *Visual Communications and Image Processing '93*, volume 2094, pp. 1477–1485. SPIE, October 1993. doi: 10.1117/12.157907.

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.

Wen Hou and Cishen Zhang. Parallel-Beam CT Reconstruction Based on Mojette Transform and Compressed Sensing. *International Journal of Computer and Electrical Engineering*, pp. 83–87, 2013. ISSN 17938163. doi: 10.7763/IJCEE.2013.V5.669.

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.

Zhipeng Huang, Zhizheng Zhang, Cuiling Lan, Zheng-Jun Zha, Yan Lu, and Baining Guo. Adaptive frequency filters as efficient global token mixers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6026–6036, 2023. doi: 10.1109/ICCV51070.2023.00556.

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/.

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.

Andrew Kingston and Imants Svalbe. Generalised finite Radon transform for *N*×*N* images. *Image and Vision Computing*, 25(10):1620–1630, October 2007. ISSN 0262-8856. doi: 10.1016/j.imavis.2006.03.002.

Andrew Kirmse and Jonathan de Ferranti. Calculating the prominence and isolation of every mountain in the world. *Progress in Physical Geography: Earth and Environment*, 41(6):788–802, 2017.

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. Technical report, University of Toronto, 2009.

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.

Patrick Le Callet, Abdelhakim Saadane, and Dominique Barba. Interactions of chromatic components in the perceptual quantization of the achromatic component. In *Human Vision and Electronic Imaging IV*, volume 3644, pp. 121–128. SPIE, May 1999. doi: 10.1117/12.348432.

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.

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.

Nicolas Normand, Jean-Pierre Guédon, O. Philippe, and D. Barba. Controlled redundancy for image coding and high-speed transmission. *Proc. of the SPIE - The International Society for Optical Engineering*, 2727: 1070–1081, 1996. ISSN 0277-786X.

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.

Benoit Parrein, Pierre Verbert, Nicolas Normand, and Jean-Pierre V. Guédon. Scalable multiple descriptions on packets networks via the n-dimensional Mojette transform. *Quality of Service over Next-Generation Data Networks*, 4524(1):243–252, 2001.

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.

A. Saadane, H. Sénane, and D. Barba. Visual Coding: Design of Psychovisual Quantizers. *Journal of Visual Communication and Image Representation*, 9(4):381–391, December 1998. ISSN 1047-3203. doi: 10.1006/jvci.1998.0393.

Abdelhakim Saadane, Hakim Senane, and Dominique Barba. Design of psychovisual quantizers for a visual subband image coding. In *Visual Communications and Image Processing '94*, volume 2308, pp. 1446–1453. SPIE, September 1994. doi: 10.1117/12.185903.

Abdelhakim Saadane, Nachida Bekkat, and Dominique Barba. On the masking effects in a perceptually based image quality metric. In *Imaging and vision systems: theory, assessment and applications*, pp. 161–177. Nova Science Publishers, Inc., USA, January 2001. ISBN 978-1-59033-033-3.

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].

H. Senane, A. Saadane, and D. Barba. Image coding in the context of a psychovisual image representation with vector quantization. In *Proceedings., International Conference on Image Processing*, volume 1, pp. 97–100 vol.1, October 1995. doi: 10.1109/ICIP.1995.529048.

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.

Pierre Verbert, Jean-Pierre V. Guédon, and Benoit Parrein. Distributed and compressed multimedia transmission using a discrete backprojection operator. *Internet Multimedia Management Systems III*, 4862(1):315–325, 2002. doi: 10.1117/12.473047.

Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945.

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 lists the abbreviations used throughout our work. Appendix B presents additional background material. Appendix C presents dataset details and training recipes for our classification experiments, as well as further analysis of PsychoNet, including an efficiency benchmark (Appendix C.3) and a quantitative interpretability evaluation (Appendix C.4). Appendix D provides detailed information about the architectural configurations of all models used in our main classification experiments. Appendix E provides architecture details for our ablation studies.

A Abbreviations

DVC Deep Visual Coding 2
CNN Convolutional Neural Network 2
FT Fourier Transform 4
DFT Discrete Fourier Transform 2
IDFT Inverse Discrete Fourier Transform 16
FFT Fast Fourier Transform 8
MRI Magnetic Resonance Imaging 17
RF Receptive Field 2
PC Principal Component 18
DPR Distinct Part Rate 8

B Background

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

B.1 Psychovisual Coding

Our work is inspired by groundbreaking research conducted in the 1990s by French researchers led by Dominique Barba in understanding the human aspect of mammalian vision, i.e. the psychovisual capability of the human brain for visual perception arising from the need for early television signal compression (Hanen & Barba, 1993). At the time, statistical approaches based on Shannon's information theory and rectilinear methods such as discrete (Haar) wavelets were popular, and they argued that these approaches were sub-optimal because they treated all errors equally. They would propose psychovisual quantizers as an efficient form of image coding that would retain the important image information pertaining to its interpretation by the human vision system and quantization matched the detection thresholds of the visual cortex (Senane et al., 1995). These quantizers were proposed to be the coronas of the 2D Fourier space, where the model of the vision system assumes Fourier space is analyzed using radial symmetric functions (Saadane et al., 1994; 1998) (see Figure A.1), which they showed can also be mapped to colors in human vision (Le Callet et al., 1999). The premise is that visual recognition and feature extraction could be performed by selecting coronal sectors of Fourier space directly through the quantisation of adjacent frequencies, thereby providing directional band limited filtering within the scene. Their psychophysical experiments were also used to select the optimal sub-bands that allowed image compression that was difficult for humans to distinguish (Saadane et al., 2001). Our DVC is a data-driven adaptation of this approach, using band-limited frequency filters to learn sparse frequency selections using supervisory signals from classification and segmentation tasks.

This work on visual codes over the course of a decade would result in among the first uses of vector quantization for image coding (Senane et al., 1995), a perceptually based image quality metric (Saadane et al., 2001) and one of the foundations of discrete projection theory, where a central slice theorem is established for discrete Fourier space based as exact 1D forms of these psychovisual radial functions as slices and therefore projections in image space (Guedon et al., 1995). This work would even pioneer the use of the wavelet transform to projection data before it would be

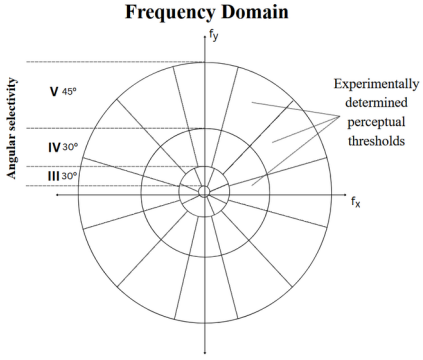

Figure A.1: Coronal frequency sub-bands used in psychovisual coding from Saadane et al. (1998).

formalized as ridgelets by Candes and Donoho (Candés & Donoho, 1999). The Mojette transform would itself form the basis of an entire area of discrete tomography that creates discrete projections of images (Normand et al., 1996) in diverse areas such as image reconstruction (Kingston & Svalbe, 2007; Chandra et al., 2014) and compression (Guédon et al., 2001), computed tomography (Hou & Zhang, 2013) and network transmission (Parrein et al., 2001; Verbert et al., 2002). Although the number of publications is too numerous to list here, a summary of these works and areas can be found in the Mojette transform book (Guédon, 2013).

B.2 The Frequency Domain

Intuitively, Discrete Fourier Transform (DFT) is the projection of any (finite) signal on the set of (harmonic) orthonormal basis functions created via the unit circle as the nth root of unity $e^{2\pi i/N} = 1$ with $N \in \mathbb{Z}$ and $i^2 = -1$ (Cooley et al., 1969), i.e., the cyclotomy or division of a unit circle into equal parts. This projection results in weights for each of these harmonics known as the Fourier coefficients required to represent the signal and comprise in its frequency domain representation. Formally, the DFT $\hat{I}(u, v)$ of an $N \times N$ image $I(j, k)$ is defined as

$$\hat{I}(u,v) = \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} I(j,k) e^{-2\pi i u j/N} e^{-2\pi i v k/N}. \tag{2}$$

The Inverse Discrete Fourier Transform (IDFT) is defined as

$$I(j,k) = \frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \hat{I}(u,v) e^{2\pi i u j/N} e^{2\pi i v k/N}. \tag{3}$$

Both the DFT and IDFT are row/column separable.

This DFT space has many useful properties, the main being the transformation of convolution operations into element-wise multiplication (i.e. the Hadamard product) known as the circular convolution property. Let $x[j,k], y[j,k], j, k \in 0, ..., N-1$ be two discrete $N \times N$ spatial signals. The circular convolution of these two signals is defined as

$$x[j,k] * y[j,k] = \frac{1}{N^2} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x[m,n] y[\langle (j-m) \rangle_N, \langle (k-n) \rangle_N], \tag{4}$$

where $\langle \cdot \rangle_N$ denotes modulo N. The Convolution Theorem (Gonzalez & Woods, 2008) then states that:

$$\mathcal{F}[x * y] = \mathcal{F}[x] \odot \mathcal{F}[y] \text{ or equivalently } x * y = \mathcal{F}^{-1}[\mathcal{F}[x] \odot \mathcal{F}[y]] \tag{5}$$

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).

B.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:

$$\boldsymbol{W} * \boldsymbol{h} = (\boldsymbol{W_R} * \boldsymbol{a} - \boldsymbol{W_I} * \boldsymbol{b}) + i\,(\boldsymbol{W_I} * \boldsymbol{a} + \boldsymbol{W_R} * \boldsymbol{b}) \tag{6}$$

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 (MRI), 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 (Phasor Blocks) that learn to generate *complementary* complex-valued features to given real features, as described in Section 3.

C Experiments

In this section we present complete dataset details, training recipes and some additional results for the 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.1 presents the training recipe used for ImageNet experiments.

Table A.1 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.2 compares DVC 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.3 compares DVC 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.

C.1.1 PsychoDW representation analysis

Figures A.4 through A.6 present qualitative visualisations and analysis for PsychoDW identical to those applied to Psycho-B we presented in Section 4. Overall, these show similar results:

- Figure A.4 **(a)** shows that PsychoDW's DVC filters also learn sparse selections of frequencies across each sub-band.

Table A.1: 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

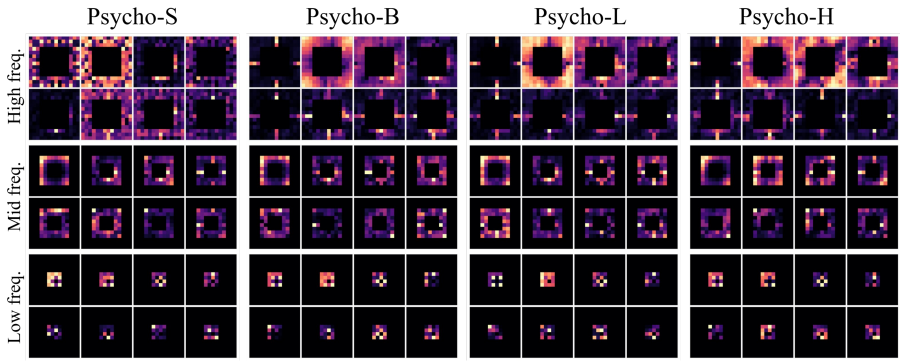

Figure A.2: Top Principal Components (PCs) of DVC filters learnt by different sized ResNet-based PsychoNet models on ImageNet-1K. 'High/mid/low freq.' refer to the [14, 8], [8, 4] and [4, 1] frequency sub-bands created by Spectral Branches.

- Figure A.4 **(b)** shows that similar to the Psycho-B vs. ResNet-101 comparison in Figure 7 **(b)**, salience maps of PsychoDW's low-mid level Phasor Blocks clearly emphasis specific object parts, while those of ConvNeXt-S are much more general and diffuse. Further examples of the former are shown in Figure A.5.
- Figure A.6 shows that similar to for Psycho-B in Figure 6, PsychoDW's DVC appears to distribute object parts by scale between the three sub-bands, and individual filters within each sub-band target distinct selections of object parts.

Overall, these results are highly consistent with those for Psycho-B, showing that DVC abstractions and object-part-centric Phasor Block representations also translate to ConvNeXt-S.

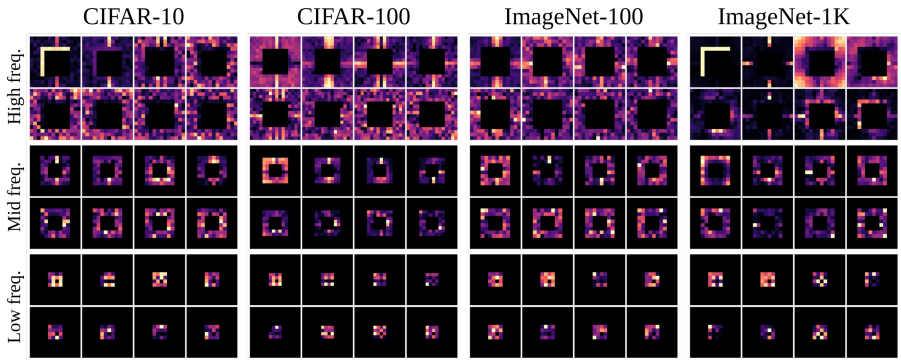

Figure A.3: Top PCs of DVC filters learnt by Psycho-B on different resolution and size datasets. 'High/mid/low freq.' refer to the [14, 8], [8, 4] and [4, 1] frequency sub-bands created by Spectral Branches.

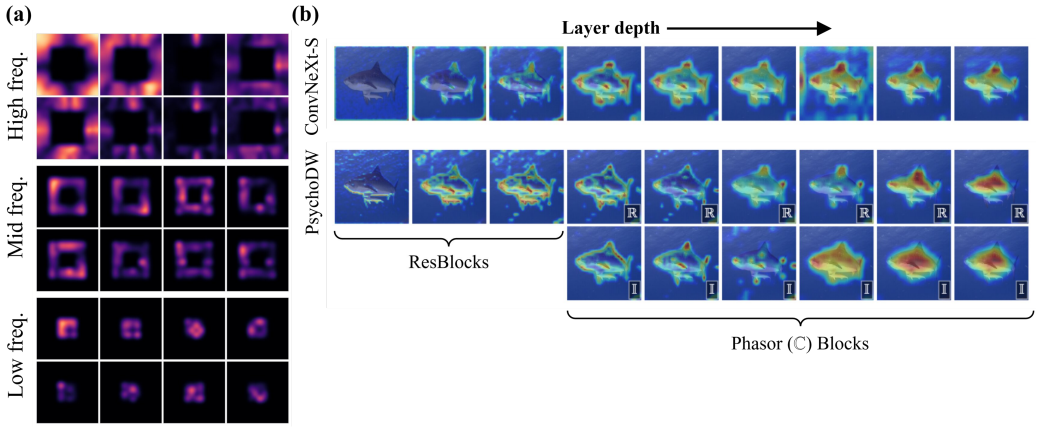

Figure A.4: **(a)** Top spatial principal components of DVC filters learnt by PsychoDW trained on ImageNet-1K. Bilinear smoothing has been applied. **(b)** Comparison between activation maps (via KPCA-CAM) of PsychoDW and ConvNeXt-S for a range of layer depths. Real and imaginary components are denoted by $\mathbb{R}$ and $\mathbb{I}$.

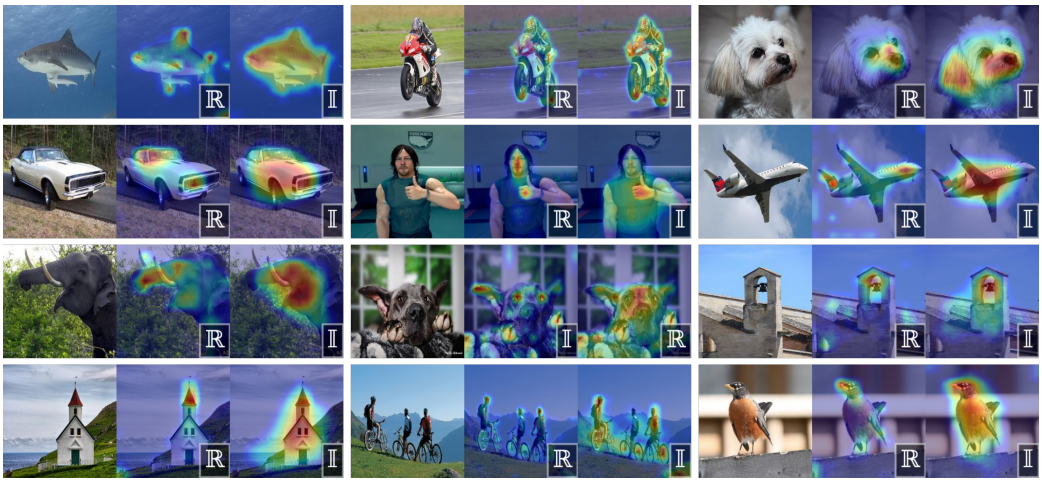

Figure A.5: Assorted activation maps (via KPCA-CAM) for mid-level Phasor Blocks of PsychoDW. Real and imaginary components are denoted by $\mathbb{R}$ and $\mathbb{I}$.

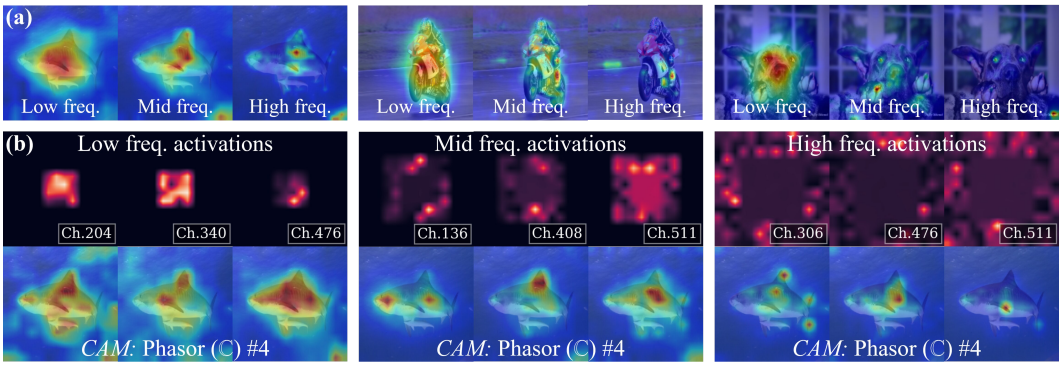

Figure A.6: PsychoDW Phasor Block salience maps (via HiResCAM) conditioned on gradients **(a)** from individual Spectral Branch sub-bands and **(b)** from individual frequency domain feature channels.

C.2 Smaller classification datasets

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.2 presents the training recipe for the CIFAR-10 experiments.

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.2 presents the training recipe for the CIFAR-10 experiments. Overall, all of our PsychoNet models outperformed their respective CNN baselines.

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.1), but reduce the batch size to 128. Psycho-S/B, Psycho-Eff-S/B and

Table A.2: 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, Psycho-Eff-S/B, ResNet50/101 $1\times$ NVIDIA H100: All other models

ResNet50/101 were trained on $1\times$ NVIDIA A100, while all other models used $1\times$ AMD MI300X. Overall, the ResNet-based PsychoNet models outperformed their respective baselines, but PsychoDW fell slightly short of ConvNeXt-S.

C.3 Efficiency benchmark

Table A.3 reports inference throughput, latency, and peak memory usage (measured on a single RTX 3080), as well as total ImageNet-1K training time, for Psycho-B (standard), Psycho-Eff-B (efficient), and the FLOP/param-matched ResNet-101 baseline. Throughput was measured with a batch size of 64 over 50 forward passes (10 warm-up passes). Latency was measured for a single forward pass at batch size 1, averaged over 100 passes (20 warm-up passes).

Table A.3: Efficiency benchmark (single RTX 3080; throughput and peak memory at batch size 64, latency at batch size 1). Param. = parameter count (millions); Tput = inference throughput (images/sec); Lat. = inference latency (ms); Mem = peak GPU memory (GB); Train (IN1K) = total ImageNet-1K training time.

Model	IN1K Top-1 (%)	Param. (M)	GFLOPs	Tput (img/s)	Lat. (ms)	Mem (GB)	Train (IN1K)
Psycho-B	78.85	42.01	30.13	246.4	24.32	1.43	2d 0h 31m
Psycho-Eff-B	78.41	45.82	15.80	378.2	21.11	1.47	1d 6h 21m
ResNet-101	78.43	44.55	15.60	751.5	9.58	0.89	19h 23m

At matched FLOPs and parameter counts, Psycho-Eff-B has lower throughput, higher latency, higher peak memory usage, and longer training time than ResNet-101. We attribute this gap primarily to the lack of hardware and framework optimisation for complex-valued operations (e.g., in cuDNN) compared to real-valued operations. Developing specialised CUDA kernels for complex-valued operations is an important direction for future work to make PsychoNet wall-clock efficient.

C.4 Quantitative interpretability evaluation

The salience maps in Figure 7 (b) suggest that the early layers of both ResNet and PsychoNet capture low-level visual features, but that in the middle-to-late Phasor Blocks PsychoNet's CAMs isolate distinct object parts while ResNet's produce a single diffuse region covering the whole subject, with both collapsing in the final layer. Figure 5 shows this object parts-focus holds across a wide range of images.

As an initial exploration into quantifying this pattern, we designed a new evaluation on the ImageNet-S50 validation set (Gao et al., 2023) (752 ImageNet images from 50 classes, with pixel-level segmentation masks), and we define a Distinct Part Rate (DPR), which applies topographic prominence (Kirmse & de Ferranti, 2017) to the peaks of the CAM salience maps. We count a peak as corresponding to a meaningful object part if it is prominent, lies mostly inside the ground-truth object

mask, and is disjoint from any already counted parts. DPR is then the fraction of images whose CAM contains at least two such parts. Section C.4.1 formally defines DPR, and Section C.4.2 describes the evaluation protocol and reports the results.

C.4.1 DEFINITION

Let $\Omega = \{1, \dots, H\} \times \{1, \dots, W\}$ be the pixel grid over which our KPCA-CAM salience maps are computed. Each salience map is min-max normalised, and so is a mapping $S : \Omega \to [0, 1]$. Computing the DPR of S comprises three steps: identifying distinct peaks of S with topographic prominence, selecting the peaks that correspond to object parts, and counting the images with at least two such parts.

Topographic prominence. Topographic prominence is a conservative criterion for identifying distinct local maxima, which tests whether a peak is separated from every higher peak by a sufficiently deep saddle. For a threshold t, let $U_t = \{u \in \Omega : S(u) \geq t\}$ be the superlevel set of S. For a local maximum p, let $U_t(p)$ denote the connected component (under 8-connectivity) of U_t containing p. The saddle level of p, given by

$$s(p) = \max \{t \leq S(p) \, : \, \exists \, q \in U_t(p) \text{ with } S(q) > S(p)\}, \tag{7}$$

is the highest level at which the component around p joins that of a higher peak, and its prominence is

$$\pi(p) = S(p) - s(p). \tag{8}$$

For the global maximum, we set $s(p) = \min_u S(u)$, so that $\pi(p) = 1$. This is an adaptation of topographic prominence, the criterion used in topography and mountaineering to determine whether a summit is a distinct peak or just a bump on the ridge of a higher one (Kirmse & de Ferranti, 2017). When computing DPR, we treat p as a distinct peak if $\pi(p) \geq \delta$, with $\delta = 0.2$, which requires the salience to dip by at least 20% of the map's range between p and any higher peak, a threshold we found suitable to exclude minor fluctuations within a single salient region.

Selecting on-object parts. After identifying peaks of S, we compare them to the ground-truth ImageNet-S50 segmentation map to see if they lie within the object of interest. To do so, we first give each distinct peak p a spatial extent, assigning it the region around p halfway down from the peak to its saddle,

$$R(p) = U_{S(p) - \pi(p)/2}(p). \tag{9}$$

Letting $G \subseteq \Omega$ be the binary foreground mask of the ground-truth segmentation, we keep p as an object part if its region meets three conditions:

- at least half of $R(p)$ lies inside G ($\tau = 0.5$), so that the peak sits mostly on the object rather than the background;
- $R(p)$ covers at least 0.5% of the image ($\alpha = 0.005$), which excludes small speckles; and
- $R(p)$ does not lie inside the region of another peak meeting the first two conditions, so that parts are not counted more than once.

We denote the set of object parts by $\mathcal{P}$.

Distinct Part Rate. For a given network block, let S_n be the salience map of the n-th evaluation image and G_n its ground-truth mask, for $n = 1, \dots, N$. The DPR of the block is the fraction of images whose salience map contains at least two object parts,

$$\mathrm{DPR} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{1} \left[|\mathcal{P}(S_n, G_n)| \geq 2 \right], \tag{10}$$

where $\mathcal{P}(S_n, G_n)$ is the set of parts selected above for image n. Requiring at least two parts is intended to separate part decomposition from object localisation, so that a map highlighting the whole object as a single region does not score, however accurately it covers the object.

C.4.2 RESULTS

Evaluation protocol. We compare Psycho-B against ResNet-101, which have comparable parameter counts and ImageNet-1K accuracy (Table 1), evaluating all nine Phasor Blocks of Psycho-B. As the two architectures differ in depth, each Phasor Block is paired with the ResNet-101 residual block of nearest relative depth. Salience maps are computed with KPCA-CAM (Karmani et al., 2024) from each block's activations, using the real component for Phasor Blocks as in the $\mathbb{R}$ rows of Figures 5 and 7 (b). As both models are evaluated on the same images, we compare their per-image scores with a two-sided paired Wilcoxon signed-rank test (Wilcoxon, 1945).

Depthwise comparison. Table A.4 reports DPR for each matched pair of blocks. Psycho-B has higher DPR at three consecutive depths, consistent with the trajectory in Figure 7 (b) that the mid-to-late Phasor Blocks capture clearer object parts. At the remaining depths the differences are small ($|\Delta| \leq 0.072$) and go in both directions, and both models fall to their lowest DPR at the deepest blocks.

Table A.4: DPR by relative network depth, Psycho-B vs. ResNet-101 at matched depth, on ImageNet-S50 (752 images). Rel. depth = block depth as a fraction of total network depth; Δ = DPR(Psycho-B) − DPR(ResNet-101); p = paired Wilcoxon signed-rank test p-value. Psycho-B blocks: $\mathbb{I}$ = Phasor ($\mathbb{I}$) block, $\mathbb{C}_k$ = k-th Phasor ($\mathbb{C}$) block; ResNet-101 blocks are indexed within its third stage. Shaded rows mark the depths at which Psycho-B's advantage is concentrated.

	Block		DPR ↑			
Rel. depth	Psycho-B	ResNet-101	Psycho-B	ResNet-101	Δ	p
0.31	$\mathbb{I}$	`layer3.1`	0.430	0.358	+0.072	2e-04
0.38	$\mathbb{C}_1$	`layer3.4`	0.364	**0.422**	−0.057	6e-03
0.45	$\mathbb{C}_2$	`layer3.6`	0.359	0.407	−0.048	1e-02
0.52	$\mathbb{C}_3$	`layer3.9`	0.382	0.328	+0.053	8e-03
0.59	$\mathbb{C}_4$	`layer3.11`	**0.609**	0.319	+0.290	5e-41
0.66	$\mathbb{C}_5$	`layer3.14`	0.508	0.251	+0.257	1e-34
0.73	$\mathbb{C}_6$	`layer3.16`	0.469	0.250	+0.219	7e-28
0.80	$\mathbb{C}_7$	`layer3.19`	0.250	0.226	+0.024	0.13
0.88	$\mathbb{C}_8$	`layer3.21`	0.133	0.177	−0.044	5e-03

Category-wise comparison. Having observed that the DPR advantage is concentrated at specific depths, we next explore whether it is also concentrated in specific image categories. Qualitatively, the clearest CAMs were for classes with distinct characteristic parts, such as animals and vehicles (Figure 5). To check this quantitatively, we group images by category and compare Psycho-B and ResNet-101 at each model's respective best-scoring block (relative depth 0.59 for Psycho-B and 0.38 for ResNet-101, per Table A.4). Three food and fungus classes (53 images) are omitted.

Table A.5: DPR by image category, at each model's best-scoring block. Classes / images = number of ImageNet-S50 classes / images in the category; Δ, p as in Table A.4.

Category	Classes	Images	Psycho-B	ResNet-101	Δ	p
Animal	16	235	0.630	0.383	+0.247	4e-12
Vehicle	7	104	0.663	0.356	+0.308	4e-07
Object	24	360	0.561	0.464	+0.097	2e-04

By this metric, Psycho-B shows an advantage on Animal and Vehicle classes, with a smaller margin for generic Objects, broadly consistent with the qualitative results. This aligns with what these classes look like: animals have visually distinct parts such as fins, eyes, and limbs, and vehicles have parts like wheels and wings, whereas many generic objects present as a single, less separable form. This tentatively supports that Psycho-B's part-decomposition advantage is not merely an artefact of the DPR metric or a handful of favourable images, but may instead follow a category-dependent property of the underlying objects. We note that DPR is only one possible lens on interpretability, and developing more rigorous evaluations of this kind is an important direction for future work.

C.5 CLUSTERING VISUALISATION

Figure A.7 presents an initial visualisation of clustering characteristics of Phasor Block activations and DVC for Psycho-B. These show 2D PCA projections of features computed on samples from 10 randomly-selected classes from ImageNet-1K. Observable clustering emerges across both the real and imaginary/magnitude-phase feature components, and becomes increasingly pronounced at deeper layers.

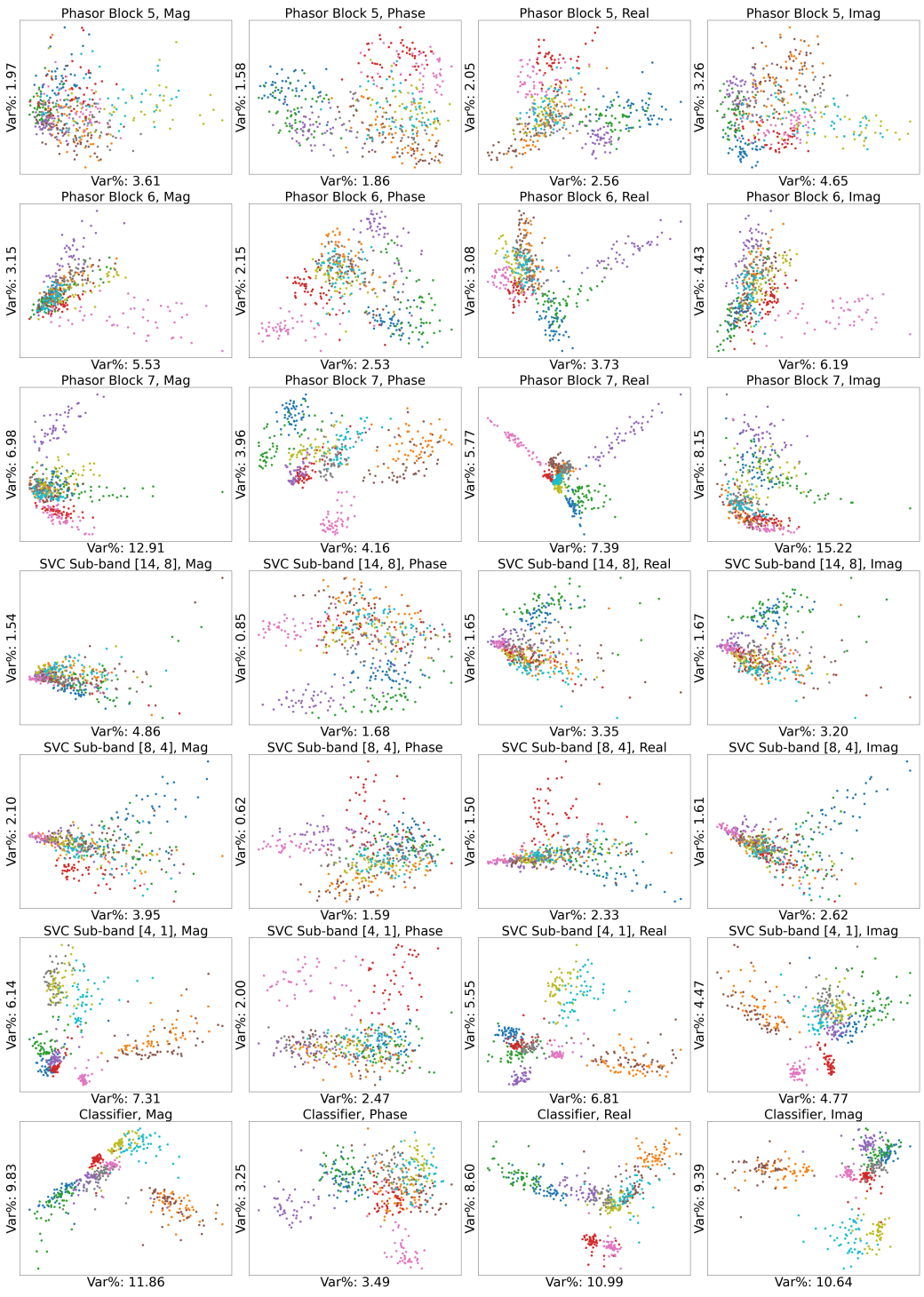

Figure A.7: Psycho-B clustering visualisation.

D 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.

D.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.6 compares the sizes of the four ResNet models as well as their block configurations, grouped by feature resolution (which are $56 \times 56,\ 28 \times 28,\ 14 \times 14$ and 7×7).

Table A.6: 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).

D.2 PSYCHONET

Table A.7 summarises the key configuration details of each base PsychoNet variant, namely the feature resolution and channel width at each Phasor Block and the number of filters used in the DVC module. Further architectural details for each model are reported in Tables A.8 through A.12. 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. Interestingly, we found that using the initial layers of ResNet50, 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.8) to depthwise convolution blocks to maintain general faithfulness to the ConvNeXt model.

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

$$x \in \mathbb{C}, \quad \text{Compand} : x \to |x|^{\frac{1}{1.25}} \cdot \exp(i \angle x) \tag{11}$$

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.

Complex-valued Softmax In Hadamard Blocks (Eq. 1), our complex softmax operation applies softmax only to the magnitude of complex features, with phase preserved and reapplied after normalisation:

$$\text{softmax}_{\mathbb{C}}(\mathbf{z}) = \text{softmax}\left(|\mathbf{z}|\right) \odot \exp\left(i \angle \mathbf{z}\right) \tag{12}$$

where $|\mathbf{z}|$ and $\angle \mathbf{z}$ denote the magnitude and phase of $\mathbf{z} \in \mathbb{C}$ respectively.

Table A.7: Configuration summary of different PsychoNet variants. For Phasor Blocks, we display **resolution**: [#channels per block], and '($\mathbb{I}$)' denotes a Phasor Block ($\mathbb{I}$)). #Filters denotes the number of channels of element-wise filters per sub-band of DVC. #Layers show overall layers / complex convolution layer counts.

Model	Phasor Blocks	#Filters	#Layers	Params (M)
Psycho-S	$\mathbf{14 \times 14}$: [256 ($\mathbb{I}$), 256, 384, 512, 512]	512	65 / 9	25.35
Psycho-B	$\mathbf{28 \times 28}$: [256 ($\mathbb{I}$), 256, 256, 384] $\mathbf{14 \times 14}$: [384, 384, 512, 512, 512]	512	93 / 13	42.01
Psycho-L	$\mathbf{28 \times 28}$: [256 ($\mathbb{I}$), 512, 512, 512] $\mathbf{14 \times 14}$: [512, 512, 512, 512, 512]	512	93 / 13	61.28
Psycho-H	$\mathbf{28 \times 28}$: [256 ($\mathbb{I}$), 512, 512, 512] $\mathbf{14 \times 14}$: [512, 512, 512, 640, 1024]	1024	93 / 13	88.61
Psycho-DW	$\mathbf{28 \times 28}$: [256 ($\mathbb{I}$), 256, 256, 512] $\mathbf{14 \times 14}$: [512, 1024, 1024, 1024, 1024]	2048	109 / 13	49.512

Table A.8: Detailed architecture of Psycho-S.

Psycho-S - comparable size to ResNet50	
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 ResNet50 (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.9: Detailed architecture of Psycho-B.

Psycho-B architecture - comparable size to ResNet101	
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 ResNet101 (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.10: Detailed architecture of Psycho-L.

Psycho-L architecture - comparable size to ResNet152	
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 ResNet152.
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.11: Detailed architecture of Psycho-H.

Psycho-H architecture - comparable size to ResNet270	
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 ResNet270.
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.12: Detailed architecture of PsychoDW.

PsychoDW architecture - comparable size to 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 ResNet50.
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

D.3 PHASOR BLOCK ARCHITECTURE

Figure A.8 provides detailed architectural diagrams of Phasor Blocks, with key design choices discussed below.

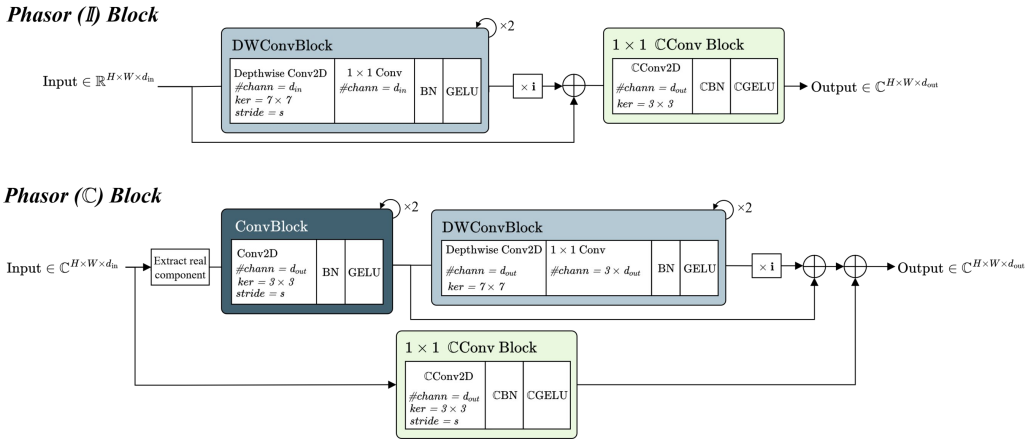

Figure A.8: Further architecture details for the Phasor Blocks presented in Figure 3. For ConvNeXt-based PsychoNet, we replace the two ConvBlocks at the start of Phasor $(\mathbb{C})$ blocks with two DWConvBlocks with the same number of channels. ℂConv/BN/GELU denote complex-valued convolution, batch norm and GELU operations - see Appendix B.3. The following PsychoNet architecture tables specify the values of d_{in}, $d_{\text{out}}$and stride (s) for all of their Phasor Blocks.

Phasor $(\mathbb{I})$ blocks generate an initial set of imaginary components using depthwise convolution ('DWConv') blocks, comprising pairs of depthwise and 1×1 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, 2008; 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-based PsychoNet, we replace Phasor $(\mathbb{C})$'s regular convolution ('Conv') blocks with further DWConv blocks with 7×7 kernel size, matching ConvNeXt's main computational block.

D.4 EFFICIENT PSYCHONET

The base PsychoNet models have considerably higher FLOP count compared to ResNet primarily because they only downsample spatial resolution to 14×14 at smallest, whereas ResNet (and many other standard CNNs) typically downsample further to 7×7. To address this, we introduced the FLOP-efficient PsychoNet variants Psycho-Eff-S/B/L, which incorporate two main architectural changes to improve computational efficiency. First, we add an additional downsampling stage to the Phasor Blocks to reduce the feature resolution to 7×7. In these models, the $[14, 7]$ sub-band is extracted from the last 14×14 Phasor Block, while the remaining sub-bands are extracted from the final 7×7 block. The specific Phasor Blocks from which these sub-bands are extracted for DVC are explicitly labelled in the architecture tables below. Second, within the Phasor $(\mathbb{C})$ blocks, we streamline the architecture by using only one Conv Block and one DWConv block respectively, whereas the base PsychoNet models utilize two of each (as shown in Figure 3). Table A.13 summarises the key configuration details of each efficient PsychoNet variant, namely the feature resolution and channel width at each Phasor Block and the number of filters used in the DVC module. Further architectural details for each model are reported in Tables A.14 through A.16.

Table A.13: Configuration summary of different efficient PsychoNet variants. For Phasor Blocks, we display **resolution**: [#channels per block], and '$(\mathbb{I})$' denotes a Phasor Block $(\mathbb{I})$). #Filters denotes the number of channels of element-wise filters per sub-band of DVC. #Layers show overall layers / complex convolution layer counts.

Model	Phasor Blocks	#Filters	#Layers	Params (M)
Psycho-Eff-S	$\mathbf{14 \times 14}$: [256 $(\mathbb{I})$, 256, 512] $\mathbf{7 \times 7}$: $[512, 512, 768]$	768	57 / 9	25.35
Psycho-Eff-B	$\mathbf{28 \times 28}$: [256 $(\mathbb{I})$] $\mathbf{14 \times 14}$: $[512, 512, 512, 512]$ $\mathbf{7 \times 7}$: $[768, 768, 768]$	768	65 / 11	45.82
Psycho-Eff-L	$\mathbf{28 \times 28}$: [256 $(\mathbb{I})$] $\mathbf{14 \times 14}$: $[512, 512, 512, 512, 512, 512, 512]$ $\mathbf{7 \times 7}$: $[1024, 1024]$	1024	85 / 13	62.03

Table A.14: Detailed architecture of Psycho-Eff-S.

Psycho-Eff-S	
Parameters (M) # Layers (overall) # Layers (complex)	25.35 57 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 ResNet50.
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}=512] $\rightarrow$ extract [14, 7] sub-band $\mathbf{7 \times 7}$: $(\mathbb{C})$ [d_{in}=512, d_{out}=512, stride=2] $\mathbf{7 \times 7}$: $(\mathbb{C})$ [d_{in}=512, d_{out}=512] $\mathbf{7 \times 7}$: $(\mathbb{C})$ [d_{in}=512, d_{out}=768] $\rightarrow$ extract [7, 4], [4, 1] sub-bands
Spectral filters	Sub-bands ([crop, drop]): [14, 7], [7, 4], [4, 1]
Output layer	Average pool, ComplexLinear(d_{in}=2304, d_{out}=1000), Softmax

E Ablation studies

CIFAR-10 Below we present architecture details for the barebones ablation architectures used for the experiments in Table 2. Table A.17 presents the architecture of Model I; Model II just removes the DVC entirely, directly feeding the features from the Phasor Blocks into the output layer after applying FFT. Table A.18 presents the architecture of Model III, which replaces the Phasor Blocks of Model I with simple Conv. blocks, matching the overall parameter size and layer depth of Model I. Each simple Conv. block contains two stacks of 3×3 Conv2D $\rightarrow$ BatchNorm2D $\rightarrow$ GeLU. Model IV removes DVC from Model III in the same manner as Model I $\rightarrow$ Model II.

ImageNet-1K ablations Below we present architecture details for the ablation experiments presented in Table 3.

Model A is identical to Psycho-L (Table A.10). Model B just removes Spectral Branches from Model A, replacing it with a single Hadamard Block with full band 14×14 filters and no prior DropCrop operations. Model C removes all Phasor $(\mathbb{C})$ Blocks and makes up for the resultant layer and parameter deficit by adding additional ResBlocks; its architecture is presented in Table A.19. Model D removes Spectral Branches from Model C in the same manner as Model A $\rightarrow$ Model B.

Table A.15: Detailed architecture of Psycho-Eff-B.

Psycho-Eff-B - efficient variant of Psycho-B	
Parameters (M)	45.82
# Layers (overall)	65
# Layers (complex)	11
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 ResNet101.
Phasor Blocks	$\mathbf{28 \times 28}$: ($\mathbb{I}$) [d_{in}=128, d_{out}=256] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=256, 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] $\rightarrow$ extract [14, 7] sub-band $\mathbf{7 \times 7}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=768, stride=2] $\mathbf{7 \times 7}$: ($\mathbb{C}$) [d_{in}=768, d_{out}=768] $\mathbf{7 \times 7}$: ($\mathbb{C}$) [d_{in}=768, d_{out}=768] $\rightarrow$ extract [7, 4], [4, 1] sub-bands
Spectral filters	Sub-bands ([crop, drop]): [14, 7], [7, 4], [4, 1]
Output layer	Average pool, ComplexLinear(d_{in}=2304, d_{out}=1000), Softmax

Table A.16: Detailed architecture of Psycho-Eff-L.

Psycho-Eff-L - efficient variant of Psycho-L	
Parameters (M)	62.03
# Layers (overall)	85
# 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 11 ResBlocks from ResNet152.
Phasor Blocks	$\mathbf{28 \times 28}$: ($\mathbb{I}$) [d_{in}=128, d_{out}=256] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=256, 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] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\mathbf{14 \times 14}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=512] $\rightarrow$ extract [14, 7] sub-band $\mathbf{7 \times 7}$: ($\mathbb{C}$) [d_{in}=512, d_{out}=1024, stride=2] $\mathbf{7 \times 7}$: ($\mathbb{C}$) [d_{in}=1024, d_{out}=1024] $\rightarrow$ extract [7, 4], [4, 1] sub-bands
Spectral filters	Sub-bands ([crop, drop]): [14, 7], [7, 4], [4, 1]
Output layer	Average pool, ComplexLinear(d_{in}=3072, d_{out}=1000), Softmax

Table A.17: Detailed architecture of Ablation Model I.

Ablation Model I	
Parameters (M)	2.366
# Layers (overall)	16
# Layers (complex)	5
Blocks	
Input layer	Conv2D(3×3, d_{in}=3, d_{out}=64, stride=1), No MaxPool
Phasor Blocks	($\mathbb{I}$) [d_{in}=64, d_{out}=256, stride=2] ($\mathbb{C}$) [d_{in}=256, d_{out}=256]
Spectral filters	Sub-bands ([crop, drop]): [8, 4], [4, 1], d_filter = 256
Output layer	Average pool, ComplexLinear(d_{in}=512, d_{out}=10), Softmax

Table A.18: Detailed architecture of Ablation Model III.

Ablation Model III	
Parameters (M)	2.360
# Layers (overall)	17
# Layers (complex)	1
Blocks	
Input layer	Conv2D(3×3, d_{in}=3, d_{out}=64, stride=1), No MaxPool
Simple Conv. blocks	[d_{in}=64, d_{out}=64] [d_{in}=64, d_{out}=64] [d_{in}=64, d_{out}=128, stride=2] [d_{in}=128, d_{out}=192] [d_{in}=192, d_{out}=256]
Phasor Blocks	None
Spectral filters	Sub-bands ([crop, drop]): [8, 4], [4, 1], d_filter = 256
Output layer	Average pool, ComplexLinear(d_{in}=512, d_{out}=10), Softmax

Table A.19: Detailed architecture of the Model C ablation model.

Ablation Model C 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_{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] $\times$ 5 $\mathbf{28 \times 28}$: [d_{in}=512, d_{bot}=256, d_{out}=1024] $\mathbf{28 \times 28}$: [d_{in}=1024, d_{bot}=256, d_{out}=1024] $\times$ 6 $\mathbf{14 \times 14}$: [d_{in}=1024, d_{bot}=512, d_{out}=2048, stride=2] $\mathbf{14 \times 14}$: [d_{in}=2048, d_{bot}=512, d_{out}=2048] $\times$ 7 $\mathbf{14 \times 14}$: [d_{in}=2048, d_{bot}=128, d_{out}=512]
Phasor Blocks	$\mathbf{14 \times 14}$: ($\mathbb{I}$) [d_{in}=512, d_{out}=512, stride=1]
DVC 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