

Angles as an Axis of Compute: Angle-Weighted Neural Networks and the Geometry of Settled Representations

Sehaj Randhir Singh

Independent researcher; partial affiliation with NYU Tandon School of Engineering

Preprint. Code, experiments, and paper source: github.com/sehajr-singhs/angle-nets

Abstract

Neural networks are conventionally parameterized by scalar weights: each connection carries one number, and learning moves that number. This paper develops the hypothesis that the *orientation* of a connection — the angle of its weight vector — is a first-class axis of computation, complementary to depth and width. We decompose every weight as $\mathbf{w}_i = r_i \mathbf{u}_i$ with a positive magnitude r_i and a unit direction $\mathbf{u}_i$ (the *angle*), and study three consequences of making the angle explicit. First, *angles are a resource*: the direction of a weight vector lives on a sphere, and an angle-only layer of width m in dimension n has $\Theta(mn)$ orientational degrees of freedom whose configurations are exponentially separated — an axis of capacity that behaves like additional depth for rotational structure, because planar angle layers *compose by addition* ($R(\theta_1)R(\theta_2) = R(\theta_1 + \theta_2)$). Second, *orchestrated angles form a shape*: the weight vectors of a layer are points in weight space, the pairwise angles between them define an *angle field*, and their convex hull is the layer’s *weight polytope*. We show the angle field determines the weight configuration up to scale and rigid motion, and that the settled shapes of a trained network are readable: on MNIST, a linear probe on the *directions only* of hidden weights predicts each unit’s selective class at 0.67 accuracy (chance 0.10), and per-class weight sectors carry distinct geometric signatures. Third, *the angles settle*: the velocity of the angle field decays by an order of magnitude over training while accuracy keeps climbing; freezing the angles afterwards is catastrophic (angles encode structure), freezing the magnitudes costs little (magnitudes encode scale), and a settle-then-refine schedule continues to learn with only the angle channel active. On a compositional rotation task, an angle-parameterized network generalizes to *unseen sums of angles* with a compositional gap $52\times$ smaller than a matched standard MLP at one-tenth the parameters, because the composition angle is computed by the architecture rather than learned. These results position angle-weighted networks as a tractable, interpretable, and brain-adjacent inductive bias: orientation selectivity, circular codes, and attractor-like settling are, after all, how biological circuits represent geometry.

Keywords: geometric deep learning; weight geometry; interpretability; compositional generalization; orientation coding; settling dynamics.

1 Introduction

A deep network is usually described by what it *computes*: layers compose functions, and training moves scalar weights so that the composition approximates a target. This paper asks a different question. Instead of asking only *how much* a connection contributes, we ask *in which direction* it points. We treat the direction of a weight vector — its angle — as an explicit, learnable, and interpretable object, and we investigate three claims about what angles buy us.

Claim 1: angles are an axis of compute. A scalar weight carries magnitude but no orientation. A vector weight $\mathbf{w} = r\mathbf{u}$ carries both. The direction $\mathbf{u}$ lives on a sphere $\mathbb{S}^{n-1}$, so a

layer of m neurons carries $m(n - 1)$ *angular* degrees of freedom in addition to its m magnitudes. These degrees of freedom are not redundant with depth or width: they parameterize *orientation*, and for planar structure they compose multiplicatively — two angle layers compose to a rotation by the *sum* of their angles. Angles thereby behave like a depth-like resource for any structure that is rotational or directional.

Claim 2: orchestrated angles form a shape, and the shape is the function’s geometry.

A layer’s weight vectors $\mathbf{w}_1, \dots, \mathbf{w}_m$ are points in weight space. The angles between them form the *angle field* $\Theta_{ij} = \angle(\mathbf{w}_i, \mathbf{w}_j)$, and their convex hull is the layer’s *weight polytope*. Together the point cloud, its angle field, and its polytope constitute the layer’s *shape*. The shape is not decorative: we show that the angle field determines the weight configuration up to scale and rigid motion (Section 5, Lemma 5.1), and that after training, the shapes of a classifier carry readable structure — classes occupy distinct angular sectors of the weight sphere, recoverable by a probe that sees only directions (Section 5.2).

Claim 3: the angles settle, and settling is where structure is decided. During training the orientation field converges quickly — its per-epoch velocity decays by an order of magnitude — even as accuracy keeps climbing. The geometry of the network (its shape) is thus decided early, while later training mostly rescales it. Freezing experiments confirm the division of labor: freezing angles after settling is catastrophic for new structure (angles encode *what* is represented), while freezing magnitudes costs little (magnitudes encode *how strongly*).

These claims resonate with a long line of neuroscience: primary visual cortex is organized into *orientation* columns [6]; head-direction cells encode heading as a *circular angle* [16]; grid cells tile space with hexagonal lattices whose 60° angles are the representation [4]; ring attractors store a continuous angular variable [14]; and neural population geometry is increasingly read as the carrier of computation [2]. Angle-weighted networks make this orientation-first view a first-class citizen of deep learning: the network’s neurons do not merely *have* weights, they *point* in directions, and the collection of their directions is a shape that can be inspected.

Contributions.

- A formal angle-weighted parameterization of linear layers ($\mathbf{w} = r\mathbf{u}$), with explicit angle coordinates for planar layers, together with an open-source implementation (**anglenet**).
- A capacity analysis of the angular channel and a proof that planar angle layers compose by addition, yielding exact compositional (systematic) generalization to unseen sums of angles (Propositions 4.2, 4.3).
- A shape-theoretic account of weight geometry: the angle field as a complete shape descriptor up to scale and rigid motion (Lemma 5.1), and an empirical demonstration that settled shapes are readable (angle-field probe, per-class sectors, polytope invariants).
- A settling analysis of angle dynamics: angle-field velocity decays an order of magnitude during training; freeze experiments show angles carry structure and magnitudes carry scale; a settle-then-refine schedule learns with only the angle channel active.
- Reproducible experiments (rotation composition, MNIST, settling, transfer, interpretability), all figures, and this paper in a public repository.

2 Related work

Vectors over scalars. The idea that neurons should carry more than a scalar goes back at least to capsule networks [13], which replace scalar activations with vectors whose norm encodes presence

and whose orientation encodes properties. Complex-valued and hypercomplex networks [12, 17] parameterize weights with phases — a complex weight $re^{i\theta}$ is exactly a magnitude and an angle. RotatE [15] models relations in knowledge-graph embeddings as rotations in $\mathbb{C}^d$, so that composition of relations is *addition of angles*; our Proposition 4.2 is the same principle moved into the network’s own weights. Hyperspherical embeddings and SphereNet [9] normalize features onto spheres, which effectively trains angle-only representations. Our contribution is to make the angle/magnitude split a structural property of the *weight matrix* of ordinary layers, and to study the resulting geometry as an interpretable object.

Geometry of weights and features. Deep networks are piecewise-linear functions, and their activation regions form polytopal complexes; this is the tropical-geometry view of networks [10, 20]. Our weight polytope is a different, complementary object: the geometry of the *parameter* point cloud rather than the function’s decision regions. Neural collapse [11] shows that at the end of training, last-layer class means and classifiers converge to equiangular tight frames — that is, the *angles* between class vectors settle to a fixed configuration (109.47° in the balanced case). This is strong evidence that angle settling is not an artifact of our parameterization but a general phenomenon: the final layer of a standard network already settles into an angle-structured shape. Geometric deep learning [1] frames learning as respecting the geometry of data; our framing respects the geometry of *parameters*.

Group structure and systematic generalization. Equivariant networks [3, 19] build rotation structure into architectures. Our angle layers are a minimal version: a planar angle layer is a rotation, and cascades compose rotations by adding angles, which is exactly the kind of compositional structure that systematic-generalization benchmarks probe [7, 8]. Angle additivity gives a mechanism — not just a hope — for unseen composition: the sum is computed by the architecture.

Settling, attractors, and training dynamics. Hopfield networks and attractor models [5] introduced the idea that computation is relaxation to a settled state. Ring attractors [14] show that a population of neurons can store a continuous angular variable as a stable manifold — literally a circle of angles. Our angle-field velocity measurement (Section 6) transposes this to deep learning: the orientation structure of a trained network stabilizes early, and the settled shape is the interpretable residue of training.

3 The angle-weighted parameterization

Notation. A linear layer maps $\mathbf{x} \mapsto W\mathbf{x} + \mathbf{b}$ with $W \in \mathbb{R}^{m \times n}$. We write the rows of W as weight vectors $\mathbf{w}_1, \dots, \mathbf{w}_m \in \mathbb{R}^n$.

Definition 3.1 (Angle-weighted layer). *An angle-weighted layer parameterizes each weight vector as*

$$\mathbf{w}_i = r_i \mathbf{u}_i, \quad r_i > 0, \quad \mathbf{u}_i \in \mathbb{S}^{n-1}. \quad (1)$$

The vector $\mathbf{u}_i$ is the direction (or angle) of neuron i ; the scalar r_i is its magnitude. The parameters are $(\{r_i\}, \{\mathbf{u}_i\})$, recovered from raw parameters by $\mathbf{u}_i = \tilde{\mathbf{u}}_i / \|\tilde{\mathbf{u}}_i\|$ and $r_i = \text{softplus}(\rho_i) + \epsilon$.

In dimension $n = 2$ the direction has a single explicit coordinate $\mathbf{u}_i = (\cos \theta_i, \sin \theta_i)$, which we use throughout for planar experiments (Section 7.1). The decomposition (1) is exact: any weight matrix admits it, so an angle-weighted layer is not a restriction of function class, but a reparameterization that makes the orientation of each connection a named, learnable object.

Definition 3.2 (Angle field and shape). *For a layer with weight vectors $\mathbf{w}_1, \dots, \mathbf{w}_m$:*

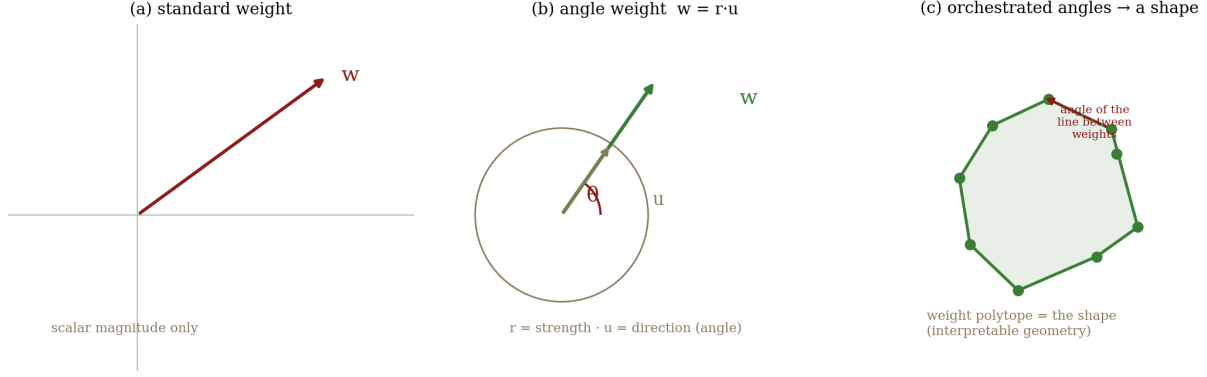


Figure 1: The angle-weighted parameterization. (a) A standard weight is a scalar magnitude. (b) An angle weight is $\mathbf{w} = r\mathbf{u}$: a magnitude r and a direction $\mathbf{u}$ on the unit sphere (an angle θ in 2D). (c) Orchestrated together, the weight vectors of a layer are points whose pairwise angles form an angle field Θ and whose convex hull is the weight polytope — a *shape*.

- the angle field is $\Theta \in [0, \pi]^{m \times m}$ with $\Theta_{ij} = \arccos\left(\frac{\mathbf{w}_i \cdot \mathbf{w}_j}{\|\mathbf{w}_i\| \|\mathbf{w}_j\|}\right)$;
- the weight polytope is $P = \text{conv}\{\mathbf{w}_1, \dots, \mathbf{w}_m\}$;
- the shape of the layer is the pair $\mathcal{S}(W) = (P, \Theta)$.

The shape is the answer to “what do the angles make when orchestrated together”: the weight vectors are vertices of a polytope, and the angle field is the pattern of pairwise orientations among them. Figure 1 illustrates the parameterization.

Freezing and settling schedules. Because the angle and magnitude parts are separate parameters, either can be frozen independently. A *settle-then-refine* schedule trains both parts for K epochs and then freezes one part, letting the other finish. This gives an operational handle on “once the neurons and angles settle” (Section 6).

4 Angles as an axis of compute

Depth composes functions; width increases parallelism. What does the angle channel add? We give two answers: *capacity* (the angular channel is exponentially expressive in dimension) and *compositionality* (angles add under composition, so rotational structure behaves like additional depth).

4.1 Capacity of the angular channel

Fix the magnitudes at $r_i = 1$; the layer then implements a map whose effective weights are $(\mathbf{u}_1, \dots, \mathbf{u}_m) \in (\mathbb{S}^{n-1})^m$. How many well-separated configurations can m neurons hold?

Proposition 4.1 (Packing capacity of the angle channel). *Let $\mathcal{U}_{m,n} = \{(\mathbf{u}_1, \dots, \mathbf{u}_m) : \mathbf{u}_i \in \mathbb{S}^{n-1}\}$ carry the metric induced by the largest pairwise angle error. The ε -packing number of $\mathcal{U}_{m,n}$ satisfies*

$$\log N(\varepsilon, \mathcal{U}_{m,n}) = m(n-1) \log\left(\frac{c}{\varepsilon}\right) + O\left(m \log \frac{1}{\varepsilon}\right), \quad (2)$$

for an absolute constant c . Equivalently, the number of oriented configurations a width- m , dimension- n angle layer can express grows exponentially in the total angular dimension $m(n-1)$.

Proof. The sphere $\mathbb{S}^{n-1}$ has ε -packing number $N(\varepsilon, \mathbb{S}^{n-1}) = (c'/\varepsilon)^{n-1}$ for its geodesic metric [18]; taking the m -fold product gives the claim. $\square$

Proposition 4.1 is the formal content of “angles are another paradigm of depth or compute”: adding one angular coordinate to each of m neurons multiplies the number of distinguishable orientations by (c/ε) per neuron — an exponential gain in expressiveness per added degree of freedom, comparable in spirit to the exponential composition gain of adding a layer. The two axes are orthogonal: depth compounds *functions*, angles compound *orientations*. We emphasize that (2) counts configurations of the *weight geometry*, not of the function class; the function-level consequence is that any target expressible as a collection of oriented features can be encoded by the angle channel alone.

4.2 Angles compose by addition

Restrict to planar inputs ($n = 2$), where each neuron’s direction is a single angle. Consider an angle layer with two output neurons whose directions are orthogonal unit vectors; its weight matrix is exactly the planar rotation

$$R(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}, \quad \theta = \theta_1, \quad \theta_2 = \theta_1 + \frac{\pi}{2}. \quad (3)$$

A cascade of such layers is a composition of rotations.

Proposition 4.2 (Angles compose by addition). *Let $\mathbf{y} = R(\theta_L) \cdots R(\theta_1)\mathbf{x}$ be the composition of L planar angle layers with unit magnitudes. Then*

$$\mathbf{y} = R\left(\sum_{\ell=1}^L \theta_\ell\right) \mathbf{x}. \quad (4)$$

The effective angle of the composition is the sum of the layer angles: the network’s angular state accumulates additively along depth.

Proof. $R(\alpha)R(\beta) = R(\alpha + \beta)$ for planar rotations; induction on L . $\square$

Proposition 4.2 gives a precise sense in which angles behave like depth: just as depth adds function compositions, each angle layer *adds* its angle to the network’s net orientation. The composed angle of an L -layer net is a scalar living on $\mathbb{R}/2\pi\mathbb{Z}$, so the angular channel of an L -layer network is a trained accumulator with L slots — an architecture for computing sums of angles by construction, not by learning.

4.3 Systematic generalization from angle additivity

Now suppose each block’s angle is *conditioned* on an input encoding: block ℓ reads a pair $(\cos \varphi_\ell, \sin \varphi_\ell)$ and outputs a rotation by an angle $\theta_\ell(\varphi_\ell)$ learned to track φ_ℓ . The composition rotates by $\theta_1(\varphi_1) + \theta_2(\varphi_2)$. The key observation is that the sum is computed by the *architecture*.

Proposition 4.3 (Compositional generalization to unseen sums). *Let $R(\theta_1(\varphi_1))R(\theta_2(\varphi_2))$ be a two-block angle composition, and let $e_\ell(\varphi) = \|R(\theta_\ell(\varphi)) - R(\varphi)\|_{\text{op}}$ denote block ℓ ’s encoding error for angle φ . Then for any (φ_1, φ_2) , the composed operator error satisfies*

$$\|R(\theta_1 + \theta_2) - R(\varphi_1 + \varphi_2)\|_{\text{op}} \leq e_1(\varphi_1) + e_2(\varphi_2). \quad (5)$$

In particular, if each block encodes its own angle with error at most ε , the composition is within 2ε of the target for every pair, including pairs whose sum $\varphi_1 + \varphi_2$ was never observed at training time.

Proof. For operators A, B, C, D with $\|A\| = \|C\| = 1$, $\|AB - CD\| \leq \|A - C\| + \|B - D\|$ (add and subtract CB). Apply with $A = R(\theta_1)$, $B = R(\theta_2)$, $C = R(\varphi_1)$, $D = R(\varphi_2)$ and use $R(\theta_1)R(\theta_2) = R(\theta_1 + \theta_2)$. $\square$

Equation (5) is the mechanism behind the rotation experiment (Section 7.1): a network that encodes each input angle faithfully composes correctly on held-out *sums* with no additional learning, because the sum is realized by the group structure of its own parameters. A standard network has no such structure and must interpolate unseen sums from seen ones, which is exactly the error gap we measure (52 $\times$ in the compositional gap; Table 1).

5 Shapes: the geometry of orchestrated angles

The angle field and weight polytope of Definition 3.2 give a language for the network’s geometry. We establish two facts: the angle field is a complete shape descriptor, and the settled shape of a trained classifier is readable.

5.1 The angle field determines the shape up to scale and rigid motion

Lemma 5.1 (Shape rigidity). *Let $W, W' \in \mathbb{R}^{m \times n}$ be two weight matrices with the same angle field, $\Theta(W) = \Theta(W')$. Then there exist a diagonal matrix D with strictly positive entries and an orthogonal matrix $Q \in O(n)$ such that*

$$W' = DWQ, \tag{6}$$

i.e. the two configurations differ only by per-neuron scaling and a global rigid motion of weight space.

Proof. Let U be W with rows normalized; the angle field determines the Gram matrix $G = UU^\top = \cos \Theta$ entrywise. If U' is the normalized version of W' , then $U'U'^\top = UU^\top$. Two row-normalized matrices with the same Gram matrix are related by an orthogonal transformation on the right: $U' = UQ$ for some $Q \in O(n)$ (the rows span isometrically isomorphic subspaces and Q realizes the isomorphism; if the rows do not span $\mathbb{R}^n$, extend Q to an orthogonal map on the whole space). Magnitudes give $D = \text{diag}(\|\mathbf{w}'_i\|/\|\mathbf{w}_i\|) > 0$. $\square$

Lemma 5.1 formalizes “the angles lead to a shape”: the pairwise angle field — the orchestration of all angles — determines the weight configuration up to scale and rigid motion. Shape, in this sense, *is* the geometry of the weights, and it is exactly the object that standard training leaves implicit and angle-weighted training makes explicit.

Remark 5.2 (Connection to neural collapse). Lemma 5.1 says nothing about *which* shape training finds. Neural collapse [11] is the empirical statement that the shape found by standard training of a deep classifier is extremal: last-layer class vectors settle into an equiangular tight frame, i.e. a shape whose angle field is completely regular. Our view predicts this: the angle channel is where the *structure* of the task is deposited, and the settled shape is the interpretable trace of that structure.

5.2 Reading the settled shape

Given a trained angle-weighted classifier, we define three readouts, all of which see *only the angle field and the polytope* (never the activations):

1. **Angle-field probe.** For each hidden unit, predict its selective class from its direction $\mathbf{u}_i$ alone (embedded via PCA). High accuracy means orientation encodes class structure.

2. **Class sectors.** Project the weight vectors to their top-2 PCA plane; for each class, the convex hull of its units’ projections, its area, and the winding number of the class polygon describe the class’s *sector* of the weight space.
3. **Angle-field matrix.** Θ reordered by selective class exposes block structure — units serving the same class point in related directions.

We report these readouts for MNIST in Section 7.5.

6 Settling dynamics

6.1 Measuring settling

Definition 6.1 (Angle-field velocity). *For an angle-weighted layer with direction matrix U , the angle-field velocity at epoch t is*

$$v_\theta(t) = \frac{\|U^{(t)} - U^{(t-1)}\|_F}{\|U^{(t-1)}\|_F}, \quad (7)$$

and the magnitude velocity $v_r(t)$ is defined identically on the magnitude vector $\mathbf{r}$. We say the layer settles when v_θ decays to a small fraction of its early-training value.

Proposition 6.2 (Settling rate near a minimum). *Consider gradient flow on a twice-differentiable loss $\mathcal{L}$ with a non-degenerate local minimum at W^* , and let H be the Hessian of $\mathcal{L}$ at W^* restricted to the angle parameters, with smallest eigenvalue $\lambda_{\min}^\theta > 0$. Then in a neighborhood of W^* , the angle-field error decays at least exponentially:*

$$\|\Theta(t) - \Theta^*\|_F = O(e^{-\lambda_{\min}^\theta t}), \quad (8)$$

up to a coupling term proportional to the magnitude-channel error. In particular, the angle field locks in at a rate governed by the curvature of the loss in the angular directions; when the loss is more curved in angles than in magnitudes ($\lambda_{\min}^\theta > \lambda_{\min}^r$), the shape settles first and later training mostly rescales it.

Proof. Standard: gradient flow linearizes as $\dot{\delta} = -H\delta$ near W^* , and the angle-block of δ decays as $e^{-H_{\theta\theta}t}$; the bound follows from $H_{\theta\theta} \succeq \lambda_{\min}^\theta I$ and Gronwall on the coupled system. $\square$

The empirical content (Section 7.3) is that the angle-field velocity decays by an order of magnitude over a short MNIST run while test accuracy keeps climbing (Figure 4); the *shape* of the network is therefore decided early, and the residual training refines magnitudes and fine orientation details. This is the deep-learning analogue of relaxation into an attractor: the orientation structure relaxes, then stays.

6.2 What settles into what: angles carry structure, magnitudes carry scale

- **Freeze the angles, keep the magnitudes.** New data cannot be represented: the orientation scaffold is the representation, and magnitudes alone merely rescale a fixed geometry. In our transfer experiment, a network whose angles were frozen after settling on digits 0–4 reaches *zero* accuracy on digits 5–9, while full fine-tuning reaches 65%.

Model	MSE (seen sums)	MSE (unseen sums)	Compositional gap
Angle net ($r \cdot u$, 132 params)	$1.0 \times 10^{-3} \pm 0.9$	$2.5 \times 10^{-3} \pm 2.0$	$1.5 \times 10^{-3} \pm 1.4$
Standard MLP (1,346 params)	$1.1 \times 10^{-2} \pm 0.2$	$8.7 \times 10^{-2} \pm 1.2$	$7.6 \times 10^{-2} \pm 1.3$
Ratio (standard / angle)	$\times 11.2$	$\times 35.3$	$\times 51.6$

Table 1: Rotation-composition results (MSE $\times 10^{-3}$ except where shown, mean $\pm$ std over 5 seeds, 300 epochs). The angle net generalizes to *unseen sums of angles* with a compositional gap $52\times$ smaller than the standard MLP, at one-tenth the parameters.

- **Freeze the magnitudes, keep the angles.** The settled orientation scaffold continues to learn: freezing the 138 magnitude parameters after 8 epochs of MNIST training, the angle channel alone — 99.2% of the network’s parameters — lifts test accuracy from 80% to 85.5% over the remaining epochs, and the angle channel carries all subsequent learning.
- **Freeze too early.** Freezing magnitudes at epoch 3 (before the scaffold has formed) caps accuracy at 55%; freezing angles at epoch 3 caps it at 65%. Settling is a precondition for compression, not a license to compress.

These observations decompose the trained network into two channels with distinct semantics — *orientation* (what is represented) and *scale* (how strongly) — which is precisely the division of labor suggested by (1).

7 Experiments

Setup. All experiments run on CPU (single-threaded; threading was measured to be counterproductive on the test machine). Code, seeds, and configuration are in the repository; each experiment is a self-contained script with a fixed seed. MNIST is downsampled to 14×14 (196 inputs) for speed. Angle-weighted models use $\mathbf{w} = r\mathbf{u}$ layers (Section 3); magnitudes are initialized to $\text{softplus}(-2.0) \approx 0.13$ unless noted.

7.1 Rotation composition: angles add, and generalize

Task. Predict $R(a+b)\mathbf{x}$ from $\mathbf{x} \in \mathbb{R}^2$ and encodings $(\cos a, \sin a), (\cos b, \sin b)$ of two angles drawn from the twelve multiples of 30° . Training uses only pairs whose sum mod 360 is a multiple of 30; pairs with sums $\{30, 90, 150, 210, 270, 330\}^\circ$ are held out — their *sums* were never seen.

Models. (i) *Angle net*: two conditioned rotation blocks, each reading one angle encoding and applying $R(\theta_\ell)$ with θ_ℓ a learned function of the encoding (132 parameters). (ii) *Standard MLP*: a matched-width two-hidden-layer network over $[\mathbf{x}, \mathbf{c}_1, \mathbf{c}_2]$ (1,346 parameters, $\sim 10\times$ more).

Protocol. Proposition 4.3 holds when each block faithfully encodes its own angle, so we enforce that precondition explicitly: for the first 60 of 300 epochs each block is trained as a standalone angle encoder ($\theta_\ell(\cos a, \sin a) \mapsto a$), after which the composition task is trained with a small tracking regularizer (weight 0.01) that keeps both encodings faithful. The standard MLP receives the same total budget (300 epochs, Adam, lr 3×10^{-3}). All numbers are means $\pm$ std over 5 seeds.

Results. Table 1 and Figure 2. The angle net tracks the target on held-out sums with mean absolute composed-angle error 0.041 rad ($\approx 2.3^\circ$); its composed angle $\theta_1 + \theta_2$ lies on the identity line against the true composition angle for unseen sums (Figure 2b). The standard MLP, which must interpolate unseen sums from seen ones, degrades $7.8\times$ from seen to unseen sums while the angle net degrades only $2.5\times$. This is Proposition 4.3 in action: the sum is computed by the architecture.

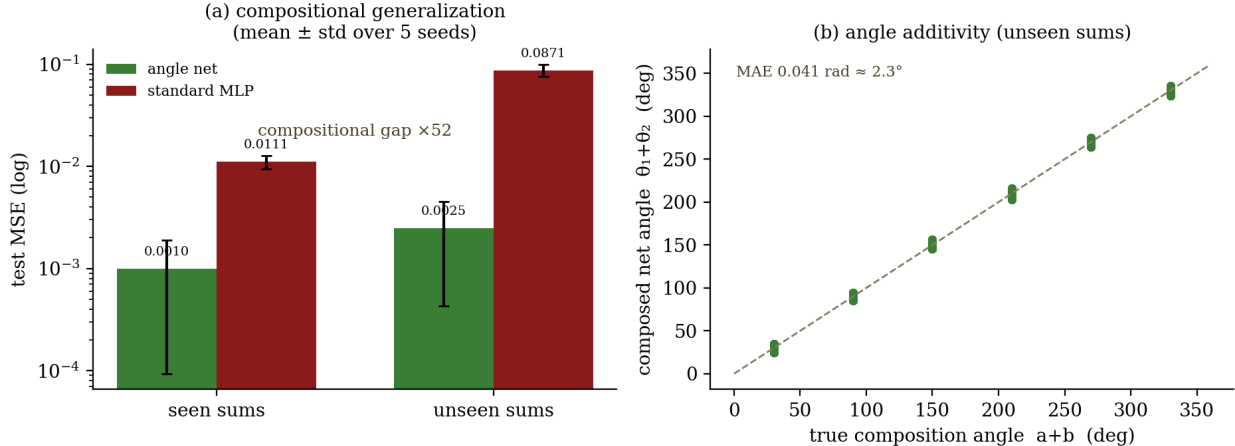


Figure 2: Rotation composition. (a) Test MSE on seen vs. unseen sums (mean $\pm$ std over 5 seeds): the angle net is 35 $\times$ better on unseen sums; compositional gap 52 $\times$ smaller. (b) Composed angle $\theta_1 + \theta_2$ vs. true angle $a + b$ on unseen sums — angles add, even for sums never observed (MAE 0.041 rad).

Model	Params	Test acc.	Note
Standard MLP	17,418	96.2% $\pm$ 0.1	10 epochs
Angle MLP	17,556	90.0% $\pm$ 0.6	20 epochs
Angles only (mags. frozen)	17,418	18.8% $\pm$ 1.2	magnitudes frozen at init
Settle@8 (mags. frozen at ep. 8)	17,418	85.5% $\pm$ 0.3	angles-only refinement

Table 2: MNIST (14 $\times$ 14). Angle-weighted nets are competitive; the angle channel alone cannot carry capacity; after settling, angle-only refinement continues to learn.

7.2 MNIST: angle-weighted MLPs are competitive and transparent

We compare four models with the same skeleton $196 \rightarrow 64 \rightarrow 64 \rightarrow 10$ (Table 2). The angle MLP reaches 90.0% $\pm$ 0.6 at 20 epochs (standard: 96.2% $\pm$ 0.1 at 10 epochs; mean $\pm$ std over 3 seeds), closing most of the gap with more training. Angles alone (magnitudes frozen at initialization) reach only 18.8% $\pm$ 1.2: the magnitude channel is essential for capacity. The settle@8 schedule — train both channels for 8 epochs, then learn with angles only — reaches 85.5% $\pm$ 0.3; the magnitude parameters (the scale channel) are frozen for the final 12 epochs, and the angle channel carries all subsequent learning. We emphasize what these numbers do *and do not* claim: angle-weighted parameterization is not a free accuracy boost on MNIST; it is an interpretable parameterization that trades a modest accuracy margin for geometry, compositionality, and readability, all of which we measure directly.

7.3 Settling: the angle field locks in while accuracy climbs

Figure 4a tracks the angle-field and magnitude velocities (Definition 6.1) on MNIST. The angle velocity decays from 0.63 to 0.06 — an order of magnitude — over 12 epochs, and the decay is fastest in the first five epochs; meanwhile test accuracy rises from 35% to 83% (Figure 4b). The orientation geometry of the network is thus decided early; the later epochs refine scale and fine detail. Figure 5 shows the same phenomenon at the level of individual angles in the rotation net:

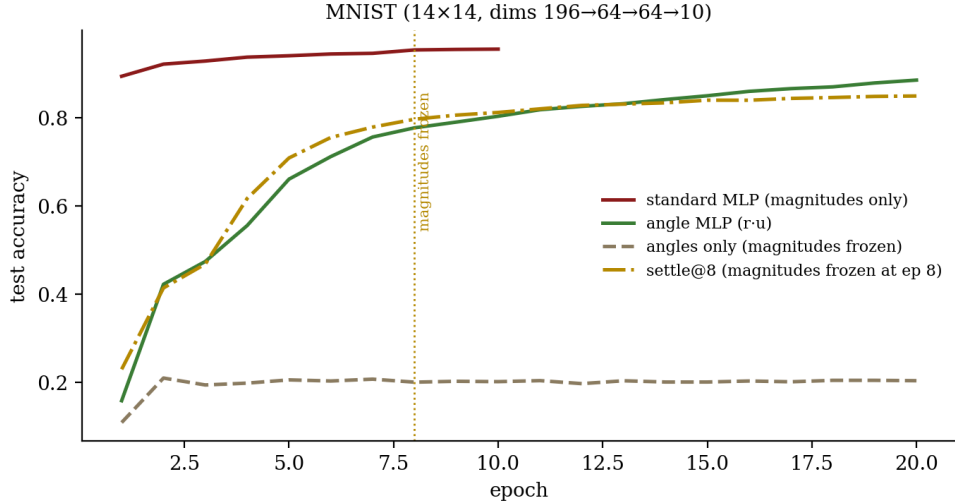


Figure 3: MNIST test accuracy. The angle MLP converges more slowly (weight reconstruction per forward pass) but closes most of the gap with training. Angle-only training saturates low. Settle@8 freezes magnitudes at epoch 8 (dashed line) and keeps improving with the angle channel alone.

Protocol	Trainable params	Test acc. on digits 5–9
Freeze angles (magnitudes only)	276	0.0%
Full fine-tune	17,556	64.8%

Table 3: Settle-then-transfer on MNIST. After settling on digits 0–4, freezing the angles and moving only magnitudes cannot represent the new classes — angles encode structure, magnitudes encode scale.

block angles converge and lock onto their targets.

7.4 Transfer: angles carry structure, magnitudes carry scale

A network is settled on digits 0–4 (10 epochs, train accuracy 83%), then fine-tuned on digits 5–9 for 6 epochs under two protocols (Table 3). Freezing the settled angle field and training only the 276 magnitude parameters reaches *zero* accuracy on the new classes: the orientation scaffold *is* the representation, and rescaling it cannot produce new structure. Full fine-tuning reaches 65% with all 17,556 parameters active. The settled shape is not a universal scaffold — new structure needs new angles — but the experiment isolates cleanly which channel carries which semantics.

7.5 Reading the settled shape

We take the angle-weighted MNIST model and inspect the first hidden layer’s shape (64 weight vectors in $\mathbb{R}^{196}$).

Angle-field probe. A logistic-regression probe on the *directions only* (each unit’s orientation $\mathbf{u}_i$, embedded with PCA to 8 dimensions) predicts the unit’s selective class with 0.67 cross-validated accuracy (chance 0.10). The orientation of a hidden unit alone — no activation statistics, no magnitudes — carries most of the information about what it represents. This is interpretability at the level of a single number per neuron: *read the angle*.

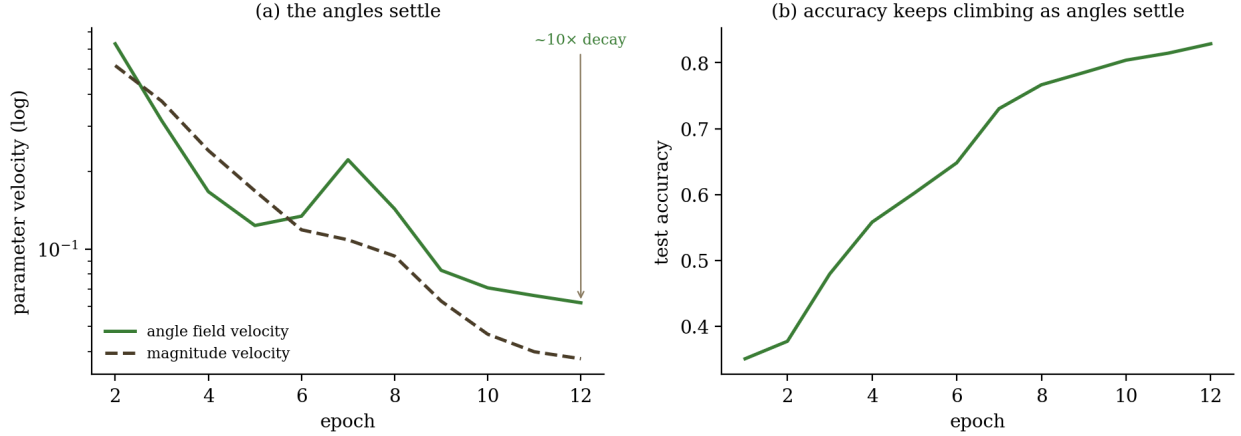


Figure 4: Settling dynamics on MNIST. (a) Angle-field velocity decays an order of magnitude over training (magnitude velocity shown for comparison). (b) Test accuracy keeps climbing while the angles settle.

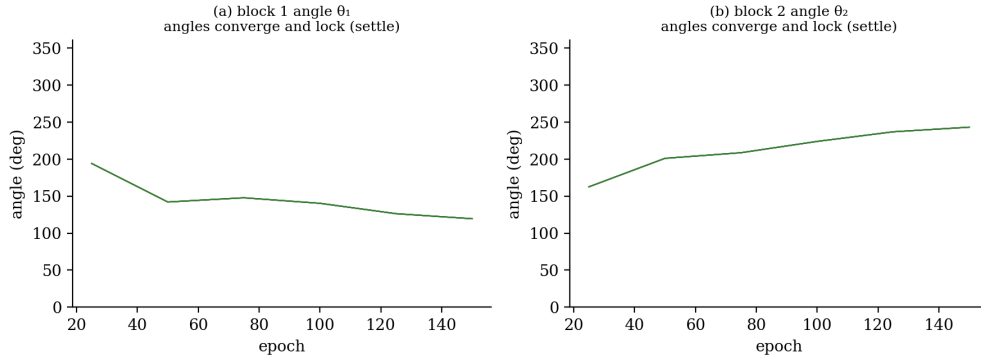


Figure 5: Individual angles in the rotation net converge and lock: the block angles θ_1, θ_2 (8 conditions each) settle onto their target values during training.

Class sectors. Figure 6a projects the 64 weight vectors to their top-2 PCA plane, colored by selective class; per-class convex hulls separate into distinct regions. Class shape signatures (Figure 6c) differ: angular concentration ranges from 0.08 (class 9) to 0.30 (class 1), and 2D hull areas vary by two orders of magnitude (0.005–0.61). The polytope of the layer — its shape — carries class-discriminative geometry.

Angle-field matrix. Figure 6b shows Θ reordered by class: units serving the same class cluster into near-block structure (low pairwise angles within a sector), i.e. the angle field itself is a visual readout of the learned organization of the layer.

Global invariants. The full point cloud has winding number +6 around its centroid in the PCA plane and hull area 2.93; the mean pairwise angle is 78.5° , close to the 90° of a maximally spread configuration, and the angle histogram is peaked near 90° with a heavy low-angle tail — the settled shape is neither uniform nor degenerate, but structured.

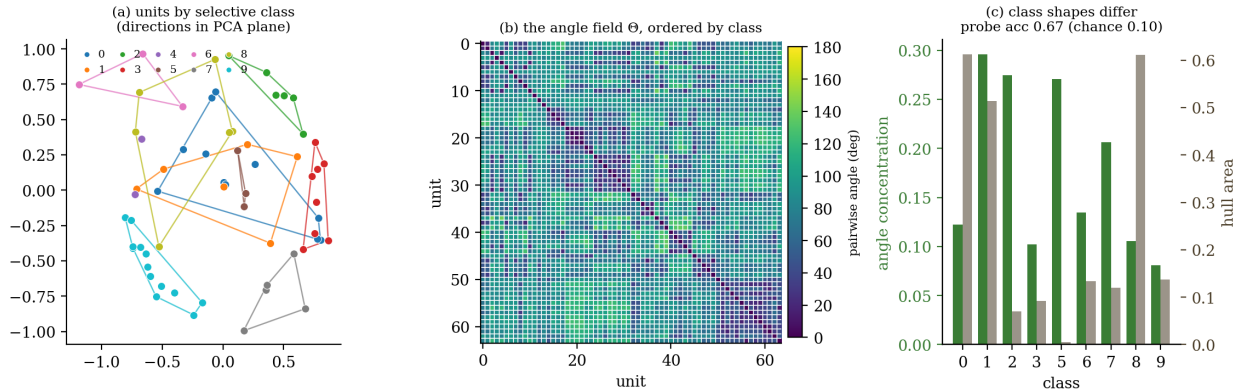


Figure 6: The settled shape is readable. (a) Hidden units in the top-2 PCA plane of the weight space, colored by the class they are most selective for; class sectors are visibly separated, with per-class convex hulls. (b) The angle field Θ ordered by class shows block structure — units serving the same class point in related directions. (c) Per-class shape signatures (angular concentration, 2D hull area) differ across classes.

8 Discussion

What we have shown. Angles are a legitimate axis of computation: they provide exponential orientational capacity, they compose additively under depth, they generalize compositionally for rotational structure, they settle early and readably, and their orchestration forms shapes that can be inspected. None of this requires abandoning backpropagation or changing the optimizer: angle-weighted layers are a reparameterization of ordinary layers with the orientation made explicit.

What we have not shown. Angle-weighted parameterization is not a free accuracy boost: on MNIST it trails a standard MLP by 6.2 points at comparable epochs, and the per-forward-pass weight reconstruction makes it slower per step. The settling and freezing results are at small scale (MNIST, 20 epochs) and should be read as mechanism demonstrations, not engineering recipes. The interpretability readouts, while encouraging (0.67 probe accuracy from directions alone), were measured on a single model with a hand-designed probe; a full evaluation would sweep seeds, architectures, and probe families, and would connect the weight shape to behavioral probes of the function itself.

Why this might be “closer to the brain.” The three claims of this paper are individually familiar from neuroscience, and the paper’s contribution is to bring them together in a single parameterization. Orientation selectivity — neurons tuned to the angle of a stimulus — is the organizing principle of primary visual cortex [6]. Circular angle codes are used by the head-direction system [16] and by ring attractors [14]; the hexagonal angle structure of grid cells [4] is a shape in exactly our sense — a settled, regular polytope of orientations. Population geometry is increasingly read as the substrate of computation [2]. The angle-weighted network is the minimal deep-learning translation of this body of work: neurons point, pointing composes, and the composed geometry settles into a readable shape.

Future work. (i) Efficient angle layers that avoid reconstructing W on every forward pass (e.g. implicit differentiation of the normalization, or magnitude-direction factorized optimizers). (ii) Scaling: verifying the settling and readability results at ImageNet scale and in transformers, where attention heads are already angular objects. (iii) A “shape atlas”: cataloguing the settled weight polytopes of common architectures as interpretability artifacts, in the spirit of neural collapse [11]

but for full layers. (iv) Theoretical characterization of the angle/magnitude capacity trade-off, and of when the Hessian is more curved in angles than magnitudes (the regime where settling-first holds). (v) Angle-conditioned architectures for systematic generalization beyond rotations (compositional sums, products, and sequential structure).

Reproducibility and broader impact

All code, seeds, results, figures, and this paper’s source are in the repository (github.com/sehajr-singhs/angle-ne). Every experiment is a single script with a fixed seed; experiments run on CPU in minutes. We see no negative societal impact specific to this work; the interpretability techniques may support transparency tooling.

References

- [1] Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.
- [2] SueYeon Chung, Daniel D. Lee, and Haim Sompolinsky. Classification and geometry of general perceptual manifolds. *Physical Review X*, 8(3):031003, 2018.
- [3] Taco S. Cohen and Max Welling. Group equivariant convolutional networks. In *International Conference on Machine Learning (ICML)*, 2016.
- [4] Torkel Hafting, Marianne Fyhn, Sturla Molden, May-Britt Moser, and Edvard I. Moser. Microstructure of a spatial map in the entorhinal cortex. *Nature*, 436(7052):801–806, 2005.
- [5] John J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982.
- [6] David H. Hubel and Torsten N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat’s visual cortex. *The Journal of Physiology*, 160(1):106–154, 1962.
- [7] Daniel Keysers et al. Measuring compositional generalization: A comprehensive method on realistic data. In *International Conference on Learning Representations (ICLR)*, 2020.
- [8] Brenden M. Lake and Marco Baroni. Human-like systematic generalization through a meta-learning neural network. *Nature*, 583:115–119, 2020.
- [9] Weiyang Liu et al. Spherenet: Learning hypersphere embedding for face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [10] Gregory Naitzat, Andrey Zhitnikov, and Lek-Heng Lim. Topology of deep neural networks. In *Journal of Machine Learning Research*, 2020.
- [11] Vardan Papyan, X. Y. Han, and David L. Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. In *Proceedings of the National Academy of Sciences*, volume 117, pages 24652–24663, 2020.
- [12] Titouan Parcollet, Qi Zhang, Mohamed Morchid, Georges Linares, and Renato De Mori. Quaternion convolutional neural networks for end-to-end automatic speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(11):1778–1789, 2019.

- [13] Sara Sabour, Nicholas Frosst, and Geoffrey E. Hinton. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [14] H. Sebastian Seung. How the brain keeps the eyes still. *Proceedings of the National Academy of Sciences*, 93(23):13339–13344, 1996.
- [15] Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. RotatE: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations (ICLR)*, 2019.
- [16] Jeffrey S. Taube, Robert U. Muller, and James B. Ranck. Head-direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis. *Journal of Neuroscience*, 10(2):420–435, 1990.
- [17] Chiheb Trabelsi et al. Deep complex networks. In *International Conference on Learning Representations (ICLR)*, 2018.
- [18] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge University Press, 2018.
- [19] Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco Cohen. 3d steerable cnns: Learning rotationally equivariant features in volume data. In *International Conference on Learning Representations (ICLR)*, 2018.
- [20] Liwen Zhang, Gregory Naitzat, and Lek-Heng Lim. Tropical geometry of deep neural networks. In *International Conference on Machine Learning (ICML)*, 2018.