CNNs Don’t See Shape

— And That Won’t Change Without New Architectures

Ali Kayyam · BrainChip Inc.

Position Paper · ICML 2026, Seoul · PMLR 306

The Question — and Our Position

The open question

- Do deep vision models recognize objects by shape or by texture? The literature remains split and unresolved.
- Humans rely on global shape; CNNs tend to lean on local texture cues.

Our position

- Purely feed-forward convolutional networks remain fundamentally texture-biased. Texture bias is rooted in architectural inductive biases — not in data or optimization alone.
- It will not be fixed by more data or augmentation. New architectures supporting global integration and relational reasoning are required.

Why the Literature Disagrees

Much of the apparent conflict comes from methodological confounds — not real disagreement about CNNs.

Cue-conflict (Geirhos et al., 2018)

- Shape and texture deliberately put in conflict; the chosen class reveals the bias.
- Confound: texture applied to background too, creating unnatural scenes and extra cues.

Cue-suppression (Burgert et al., 2025)

- Selectively removes shape or texture cues to test what survives.
- Confound: neither cue is fully removed — residual information leaks and confounds interpretation.

Cue-conflict : Baker et al. (2018) design

- Texture restricted to the object; avoids background confounds and minimizes cue leakage.

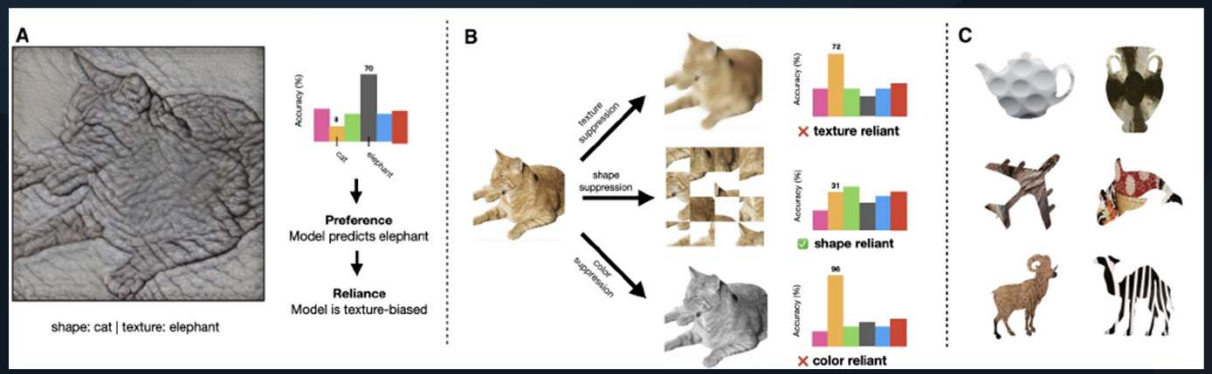


Fig. 1 — Three experimental paradigms for probing shape vs. texture bias. (A) Cue-conflict (Geirhos et al., 2018): shape and texture from different categories placed in conflict. (B) Cue-suppression (Burgert et al., 2025): residual cue leakage limits interpretability. (C) Baker et al. (2018: texture restricted to object region, background kept neutral.

Case Study: Experimental Design

Stimuli (Navon 1977 paradigm)

- Square → grating; Circle → dotted texture
- Deterministic pairing: either cue alone solves training
- 28×28 px; random size & location — no position shortcuts
- 5,000 train / 2,000 test, balanced
- Key asymmetry: removing shape while keeping texture is harder than the reverse

Goal

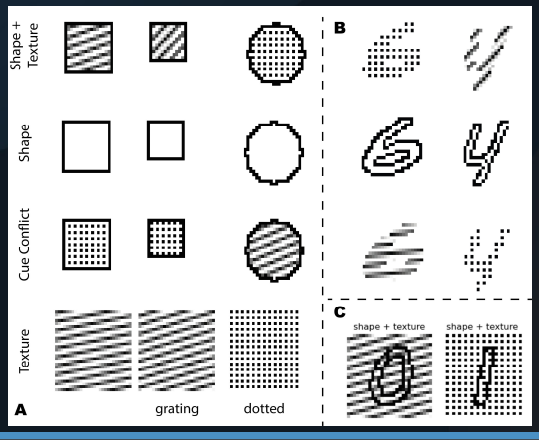
- Mechanistic isolation — which cue does the architecture prefer?

Four test conditions:

- **Shape + Texture** In-distribution baseline — both cues aligned
- **Cue Conflict** Shape/texture reversed — which cue wins?
- **Texture Only** Shape removed; whole image textured
- **Shape Only** Texture neutralized; global shape isolated

Baselines

- Random / dotted-only / grating-only → ~chance
- Replicated on MNIST & FashionMNIST



A & B) Stimuli used in our case study (grating square vs. dotted circle). Objects appear at varying locations and scales within a 28×28 image. In the texture-only condition, the entire image is textured to avoid inadvertently revealing shape information. Notably, while texture can be removed while preserving shape, the converse is considerably more difficult: eliminating shape while retaining texture is inherently challenging. This asymmetry alone may help explain why human perception is often shape-biased. C) We also consider a condition in which texture is overlaid on texture.

Case Study: Models & Training Protocol

Models compared

- **CNN —** primary focus; convolution = core inductive bias
- MLP and ViT as comparison points
- **Sparse CNN —** activation sparsity enforces global features; no data/loss changes

Training protocol

- Trained from scratch; cross-entropy loss; Adam lr=1e–3
- 10 epochs, batch 256; 10 runs per condition (mean reported)
- No augmentation or regularization — isolates architectural bias

Why these models?

- CNN: convolution directly instantiates local feature bias
- MLP: no spatial inductive bias — texture reliance must emerge from data alone
- ViT: self-attention allows global integration — but patch embedding retains local bias
- Sparse CNN: structural change only; tests whether architecture alone can shift cue reliance

Paradigm approach

- Both cue-conflict and cue-suppression evaluated under identical conditions
- Directly tests the Baker et al. (2018) design: texture on object only, neutral background
- No conflicting results between paradigms — confirms texture bias is robust

Key Result: Texture Dominates

- Near-ceiling accuracy when shape and texture agree — every model solves the task.
- Accuracy collapses under cue conflict: predictions follow texture, ignoring fully-predictive shape.
- Texture-only beats shape-only — cue-suppression agrees with cue-conflict.
- No contradiction between paradigms: shape sensitivity ≠ shape dominance when cues compete.

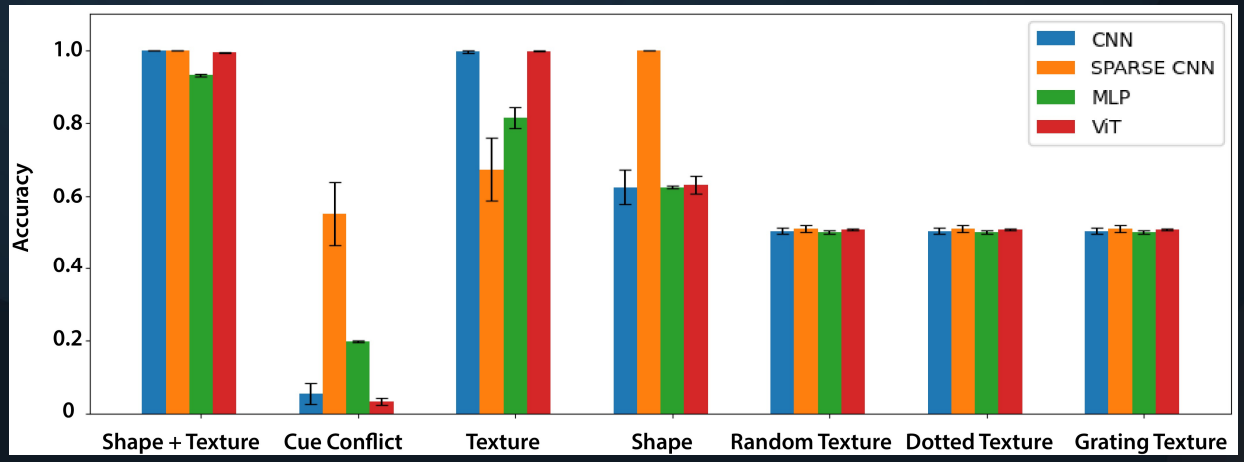


Fig. 3 — Classification accuracy across four test conditions (CNN, MLP, ViT, Sparse CNN). Texture-only consistently outperforms shape-only; cue-conflict results align with cue-suppression — no paradigm contradiction. Sparse CNN shows stronger shape bias but does not generalise to MNIST/FashionMNIST.

The Sparse CNN Probe: A Proof of Principle

What it shows

- A purely structural change — no new data, loss, or training tricks — measurably shifts cue reliance.
- Sparse CNN gains substantial accuracy in cue-conflict and shape-only conditions, with less reliance on texture.
- Sparsity nudges representations toward more global, structured shape — without sacrificing in-distribution accuracy.

Why it is not the solution

- It does not generalize: no improvement on textured MNIST / FashionMNIST.
- Suppressing local activations ≠ building genuine global shape representations.
- It opens the door by architectural means — but a more deliberate key is still needed.

Texture Bias Is Architectural, Not a Data Artifact

The bias is rooted in the inductive biases of convolution itself:

- Local receptive fields prioritize fine-grained, local statistics over long-range structure.
- Pooling — designed for translation invariance — discards the precise spatial relationships that define global shape.
- Consistent with classic findings that CNNs struggle with relational and spatial-reasoning tasks.
- ViTs are only a partial fix: self-attention enables holistic integration, yet remains insensitive to patch shuffles that destroy meaning for humans.
- Data-driven fixes (augmentation, auxiliary losses) induce behavioral invariance but leave the mechanism intact — models re-bias under distribution shift.

Alternative Views — and Our Responses

"Bias is task-dependent"

- True — texture can help in-distribution (e.g. fine-grained recognition); shape matters for robustness and generalization. We argue for balanced cue use, not strict shape dominance.

"It's just shortcut learning"

- We agree models prefer easy-to-extract features — but they are not 'cheating.' They behave exactly as architecture, objective, and optimization dictate.

"Two questions get conflated"

- Whether CNNs are inherently biased is logically distinct from which cue is more informative in natural images. Both matter, but they are not the same question.

Call to Action: Three Architectural Directions

Equivariant Networks

Group-equivariant convolutions encode geometric symmetries directly into the architecture, enforcing invariances CNNs must otherwise learn statistically. Targets the local-receptive-field pathology.

Part-Based & Capsule Nets

Capsule networks encode part–whole spatial relationships via dynamic routing, preserving the geometric structure that pooling discards. The natural candidate for configural shape understanding.

Recurrent & Feedback Nets

Horizontal and feedback connections — largely absent from feed-forward CNNs — support contour integration, perceptual grouping, and figure–ground segregation as in cortex.

Bottom Line

Mechanisms missing from today's models

- Contour integration — recognizing fragmented or sparse objects as a global whole.
- Feedback & recurrence — boundary ownership, figure–ground segregation, perceptual grouping.
- Task-driven attention and active background suppression.

The takeaway

- Texture bias is a structural property of CNNs — incremental 'patches' will not close the gap with human vision.
- Progress requires architectures that prioritize global structure over local shortcuts.

Solving the shape–texture gap may unlock the next generation of robust, trustworthy vision.

Code: github.com/alikayyam/shape_vs_texture