Is HMAX, a biologically plausible model, subject to the Müller–Lyer illusion?

Ha Dong

Abstract

I investigate whether a biologically inspired vision pipeline—combining the HMAX hierarchical feature extractor (Riesenhuber & Poggio, 1999) with a linear support vector machine (Cortes & Vapnik, 1995)—naturally exhibits the Müller–Lyer illusion despite being trained only on veridical length discrimination. First, I validated the model's pure length-comparison ability on a Control-Figure (CF) dataset by sweeping fin angle, fin length, shaft length, and vertical separation; performance remained uniformly high (>85 %) across all but the most extreme fin geometries, confirming robust CF classification. I then applied the trained classifier to a Müller–Lyer (ML) test set and observed a systematic bias: classification accuracy plunged to ~30 % when arrowheads pointed in opposite directions, well below chance, but recovered to ~82 % under uniform arrow direction. Further parametric analysis revealed that fin angle and shaft length are the principal determinants of illusion strength—mid-range angles and short shafts maximize misperception—while alterations of fin length or line separation have negligible impact. These findings mirror human psychophysics and demonstrate that simple, feedforward feature hierarchies can inherit core perceptual biases, offering a powerful framework for probing the mechanistic origins of visual illusions in both artificial and biological systems[1].

1. Introduction

Illusions, formally characterized as "consistent and persistent discrepancies between a physical state of affairs and its representation in consciousness," (Trestman, 2014) happens to all systems, biological or not. In simple terms, they represent a gap between how something 'really is' and how something 'appears to be'. Understanding this gap is instrumental. Just as studying disease states illuminates the normative functioning of a biological system, the study of illusions helps us understand the mental processing that leads to how something appears to be. The systematic and consistent nature of illusions, rather than their being random perceptual noise, suggests that they emerge from the inherent architecture and operational principles of any perceptual system. When a system consistently "misperceives" an illusory stimulus, it inadvertently reveals its underlying processing strategies, inherent biases, and the assumptions it makes about the world—characteristics that are typically veiled during accurate, or veridical, perception. Illusory stimuli often exploit these ingrained rules or present ambiguous information that the system attempts to resolve based on its established heuristics. Consequently, the perceptual "error" observed in an

[1] All code and experiments are uploaded at https://github.com/hadong1502/HMAX_illusion.git

illusion is a direct manifestation of the system's design and its learned or evolved strategies for interpreting sensory inputs.

For centuries, psychologists, neuroscientists and computer scientists have used visual illusions to methodically probe the limits, thresholds, and operational biases of biological and artificial intelligence systems. This line of inquiry extends back over a century, with works from early scientists such as Poggendorf (Zöllner, 1860), Hermann (1870), and Müller-Lyer (1889). As contemporary deep learning science emerged, this tradition continues with sophisticated modern methodologies to explore the nuances of perceptual experience and phenomenal awareness in artificial systems (Rostamkhani et al., 2024; Ullman, 2024).

The investigation of visual illusions within the domain of computer vision has gained considerable traction, fueled by the premise that if an artificial system succumbs to the same perceptual illusions as humans, it may have internalized comparable processing algorithms. Research emerging since 2018 has consistently demonstrated that CNNs, when trained on datasets of natural images for tasks like object recognition, can indeed be "fooled" by various visual illusions, exhibiting responses that are qualitatively similar to those observed in humans (Gomez-Villa et al., 2018). This susceptibility suggests that the perception of certain illusions might be an emergent property of any system optimized for complex visual tasks within naturalistic environments, rather than being solely dependent on the specific neural architecture of the biological brain. Certain authors have leveraged Generative Adversarial Networks (GANs), the use of two neural networks to compete against each other with the eventual goal of generating new data that resembles the training data, to synthesize entirely novel visual illusions (Gomez-Villa et al., 2019). By producing stimuli that elicits a maximal illusory effect on a given vision model, one can create new illusions using this approach.

This project focuses on biologically inspired models, a certain class of deep learning architecture that aims to replicate aspects of the biological intelligence system (Susan, 2024). Often, the study of biological organisms can serve as a rich source of inspiration for developing novel, computationally efficient, and robust vision models[2]. While having been the focus of study ever since artificial neuron was introduced by McCulloch and Pitts in 1943, much less attention has been paid to how such models perform given illusory inputs. The central hypothesis of this project is that a computational model designed to imitate the structure and function of specific cortical visual areas should also be susceptible to the same illusions that are thought to be processed within those biological regions. In that sense, illusions could act as a "useful diagnostic tool": by assessing whether computational models can replicate human susceptibility to specific illusions, one can make inferences about the extent of mapping between such artificial architecture and the actual human neural network, with potential insights into the underlying mental processes and computational principles.

[2] https://bicv.github.io/

Due to the limited time and resources, I focus on one of the classical visual illusions, the Müller-Lyer Illusion (Müller-Lyer, 1889), with a standard representation of two parallel lines equal in lengths. However, the design of such lines often indulges the misjudgment of line length: a line segment terminating in inward-pointing arrowheads (fins-in) appears shorter than an objectively identical line segment terminating in outward-pointing arrow-tails (fins-out) (**Fig. 1**). To date, there has been some efforts to investigate whether computational architectures experience this type of illusion. Zhang et al. (2024) investigated whether video-based deep neural networks (DNNs), equipped with spatiotemporal features and trained using a self-supervised teacher-student framework, could reproduce the Müller-Lyer illusion. They transformed synthetic Müller-Lyer stimuli into 16-frame video clips and fine-tuned several state-of-the-art models (R3D-18, MViT-V1-B, S3D, Swin3D-T, and PredNet) on two datasets—one varying line lengths, the other arrow features. By analyzing features from the penultimate convolutional layer and constructing Representational Dissimilarity Matrices (RDMs), they found that these spatiotemporal DNNs exhibited diagonal-shifting patterns in their RDMs resembling human perceptual distortions. In contrast, static CNNs like AlexNet, VGG19, and ResNet101 showed more localized sensitivity to arrowheads without replicating the illusion as clearly. Ward (2019) evaluated VGG19, alongside VGG16 and InceptionV3, on four classical geometric illusions—Müller-Lyer, Ebbinghaus, Ponzo, and Vertical–Horizontal—using a forced-choice paradigm based on cosine distance in penultimate-layer activations. While VGG19 appeared to “perceive” the Müller–Lyer illusion, it failed to reproduce the others. Notably, deeper models such as InceptionV3 achieved closer alignment with human Müller–Lyer performance. However, these models often lack biological plausibility. Additionally, in Ward (2019), decisions are based on cosine distances within a fixed feature space (FC7), providing only a descriptive measure of similarity between embeddings generated by networks trained solely for object recognition. While their approach offers some insight into how illusory images are represented in the embedding space, it does not directly correlate to how the illusion affects performance in a length-classification task, nor does it relate specific image characteristics to perceptual errors.

This project, therefore, builds upon the analytical framework proposed by Zhang et al. (2024) to ask a more targeted question: Can a biologically inspired model also be subject to the Müller–Lyer illusion, and how can we quantify them?

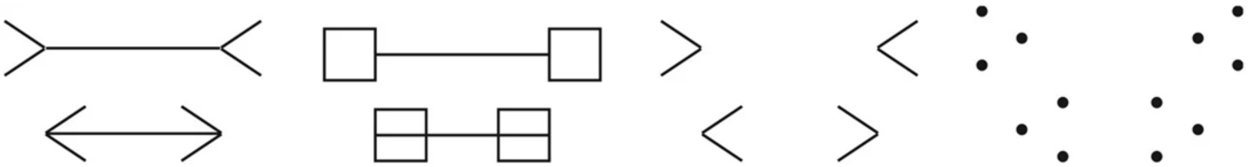

Figure 1. Variations of the Müller–Lyer illusion. Figure adapted from Howe & Purves (2005).

2. Methods

2.1. Data generation

I synthesized two distinct image sets—Cross-Fin (CF) (**Fig. 2**) and Müller–Lyer (ML) (**Fig. 3**)—each containing 10,000 gray-background, 256×256-pixel stimuli with fully annotated geometric parameters.

2.1.1. Cross-Fin Stimuli

For the CF set, every image displays two horizontally oriented line segments ("shafts") whose centers are collinear along the horizontal axis but whose vertical positions are jittered independently within the upper (20–40% of frame height) and lower (60–80%) bands to preclude reliance on spatial placement cues. Shaft lengths are sampled uniformly between 45 px and 80% of the image width; in the "EQUAL" condition (label "0"), both top and bottom shafts share the same length, whereas for images labeled "1", one shaft exceeds the other by a random offset between 2 px and 62 px. Images where the top shaft is longer is denoted "LONG", and "SHORT" otherwise. At each shaft endpoint renders an "X"-shaped fin pair whose lengths are drawn from 10 px up to the smaller of 35 px or one-third of the shaft length, and whose angles vary uniformly between 15° and 75°. To probe the role of endpoint geometry, half of the images use identical fin parameters on both shafts ("same config") and the other half use independent sampling for top and bottom fins ("different config"). All generation parameters—including label, length condition, fin configuration, individual shaft lengths, fin lengths and angles, and vertical positions—are encoded in each filename and stored within the data loaders to facilitate downstream supervised training and analysis.

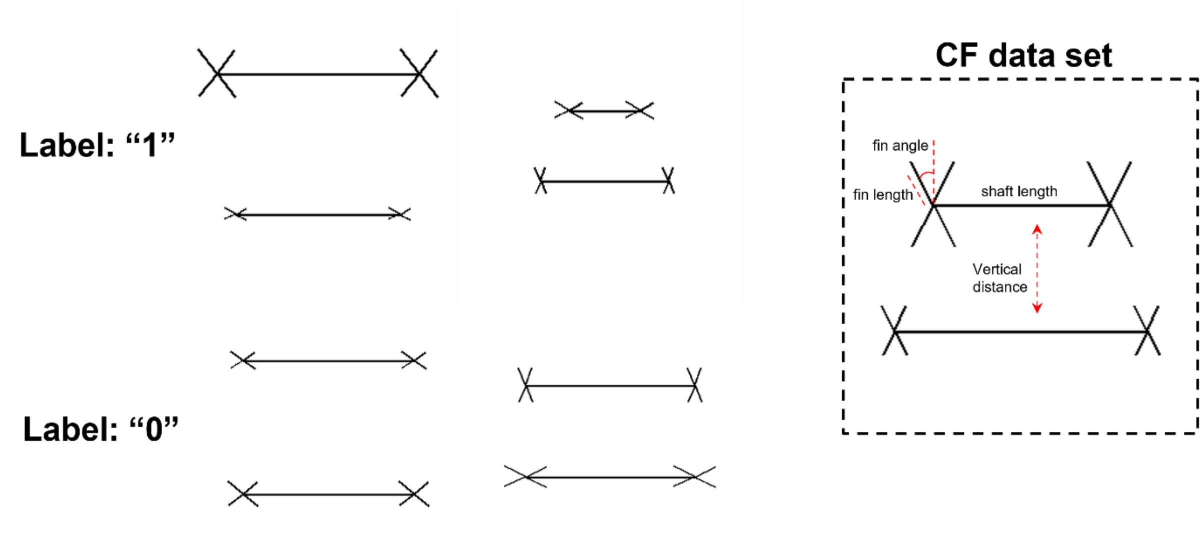

Figure 2. CF stimulus set used to verify pure length discrimination. Fin angle was varied from shallow (15°) to steep (75°); fin length was swept from 17 px to 35 px; vertical separation between the two central shafts ranged from 53 px up to 153 px; and central-shaft length itself spanned from 45 px to 204 px.

2.1.2. Müller–Lyer Stimuli

The ML dataset reuses the same uniform distributions for shaft length and vertical jitter but replaces cross fins with classic Müller–Lyer arrowheads to isolate directional cue effects. Here, both line segments share identical shaft lengths, ensuring that no length difference can inform the task. The top segment receives inward-pointing arrows (“LONG”, representing the illusory effect) or outward-pointing arrows (“SHORT”) with fin lengths and angles drawn from the same ranges as the CF fins. The bottom segment’s arrow orientation either matches (“same direction”) or opposes (“different direction”) that of the top, and again half of the images employ “same config” (matching fin geometry) while the remainder use “different config.” This factorial manipulation of arrow direction and configuration yields a rigorously controlled test set to isolates the perceptual impact of arrowhead cues on length judgments.

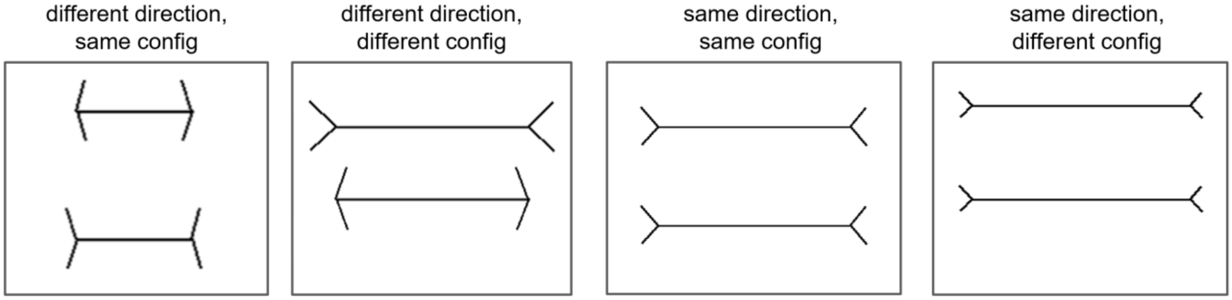

Figure 3. The panels illustrate the combinatorial conditions used to disentangle arrowhead direction and fin-size effects in the Müller–Lyer test set. From left to right: Different direction, Same config: One line has inward fins and the other outward, with fin lengths matched across both lines. Different direction, Different config: Opposing fin orientations paired with differing fin lengths/angles. Same direction, Same config: Both lines bear identical inward (or outward) fins of equal length. Same direction, Different config: Uniform fin orientation but unequal fin lengths/angles.

2.2. Model

The research presented herein necessitated the selection of a computational model with strong biological plausibility, specifically one capable of functionally mirroring aspects of the human visual ventral stream and offering a quantitative account of its object recognition processes. While several neurophysiologically inspired models have emerged, including foundational systems like the Neocognitron (Fukushima, 1980) and early convolutional neural networks (LeCun et al., 1989), my selection criteria prioritized a model that not only draws direct inspiration from primate visual cortex but also has demonstrated extensive validation against neurological and psychological findings. The Hierarchical Model and X (HMAX) distinguishes itself in this regard. For instance, HMAX has shown compelling congruence with psychological data, notably through versions of the model successfully accounting for human performance in rapid visual categorization tasks by providing quantitative fits to behavioral data (Serre et al., 2007). Psychophysical analysis shows HMAX’s intermediate (C2) features achieve human-level classification with similarly low numbers of training examples—mirroring human ease of learning in rapid-recognition tasks (Crouzet & Serre, 2011). Serre, T. (2004) also showed that HMAX’s C1/C2 max-pooling units could quantitatively reproduce V1- and V4-like invariance (shift/scale tolerance) and IT “view-tuned” cell properties observed in macaque electrophysiology.

2.2.1. HMAX Model

HMAX is a computational model of object recognition in the primate visual cortex, first proposed by Riesenhuber and Poggio (Riesenhuber & Poggio, 1999). The "X" in HMAX has been colloquially understood to represent the MAX operation central to its function. It is fundamentally a feedforward architecture designed to mimic the initial rapid processing stages (approximately the first 100-150 ms) in the ventral visual pathway, responsible for object identification (Chikkerur, S. & Poggio, T., 2011). The model's core design principles are heavily inspired by the seminal work of (Hubel & Wiesel, 1962) on the simple and complex cells in the primary visual cortex (V1). HMAX aims to achieve robust object recognition by building a hierarchy of feature representations that become increasingly complex and invariant to transformations such as position, scale, and, to some extent, point of view (Riesenhuber & Poggio, 1999). This approach so far has demonstrated excellent performance comparable to human levels on certain rapid object recognition tasks while achieving biological plausibility and computational efficacy (Deng et al., 2018; Li et al., 2015).

The development of HMAX was driven by the need to understand how the brain achieves two critical, yet seemingly contradictory, aspects of vision: specificity (distinguishing between different objects) and invariance (recognizing the same object despite variations in its appearance). The model posits that these are achieved through a series of alternating layers performing distinct computational operations (Riesenhuber & Poggio, 1999). While initially focused on explaining cortical mechanisms, HMAX has also found applications and inspired further research in computer vision due to its structured approach to feature learning and invariance (Louie, J., 2003).

2.2.1.1. General HMAX architecture

The HMAX architecture is characterized by a hierarchical structure composed of alternating layers of "Simple" (S) units and "Complex" (C) units. This layered organization is a direct reflection of the hierarchical processing observed in the primate ventral visual stream, where neurons in successive areas respond to increasingly complex stimuli and exhibit greater invariance (Serre, 2014). The model typically consists of at least two pairs of S and C layers (S1, C1, S2, C2), though extensions can include more (Li et al., 2015).

S-layers (Simple layers): These layers perform a template matching or filtering operation. S-units are tuned to specific patterns in their input. For instance, S1 units are often modeled as Gabor filters, responding to oriented edges at particular locations and scales, analogous to simple cells in V1. Higher S-layers (e.g., S2) combine inputs from the preceding C-layer to detect more complex feature conjunctions or patterns. The response of an S-unit generally increases with the similarity between the input and its preferred template.

C-layers (Complex layers): These layers perform a pooling operation, typically a maximum (MAX) operation, over a group of afferent S-units (or C-units from a lower layer) that share similar feature selectivity but differ in aspects like position or scale. This pooling mechanism is crucial for building invariance. For example, C1 units pool responses from S1 units of the same orientation over a local spatial neighborhood and across a small range of scales, thereby achieving some tolerance to shifts in position and changes in size. This operation is inspired by the properties of complex cells in V1, which have larger receptive fields and exhibit phase invariance.

2.2.1.2. Model Design

S1 Layer: Gabor Filtering

These neurons are analogous to simple cells in the primary visual cortex (V1). The input image (typically grayscale) is convolved with a bank of Gabor filters, resembling the receptive field of simple cells, at multiple orientations (typically 4) and scales. A 2D Gabor filter in the spatial domain can be defined as a Gaussian kernel modulated by a sinusoidal plane wave (Rangayyan et al., 2010).

To achieve optimal spatial and frequency localization at $(x,\ y)$, one can start with a 2D Gaussian function:

$$\mathrm{G_{env}(x,\ y)\ =\ exp}\left(-\frac{\mathrm{x}^2\ +\ \mathrm{y}^2}{2\sigma^2}\right),$$

where σ determines the size of the receptive field.

In some specific cases, an aspect ratio (γ) is included to adjust the elliptical shape of the envelope:

$$G_{\mathrm{env}}(x,y) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right).$$

Selectivity to specific spatial frequencies is introduced through a cosine function:

$$G(x,y) = G_{\mathrm{env}}(x,y) \cdot \cos\left(2\pi\frac{x}{\lambda} + \psi\right),$$

where λ is the wavelength of the sinusoidal factor.

For each orientation (θ) of the Gabor filter, the rotated coordinates are defined as

$$x' = x \cos \theta + y \sin \theta$$

$$y' = -x \sin \theta + y \cos \theta\,.$$

A Gabor filter is then defined as

$$G(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{(x \cos \theta + y \sin \theta)^2 + \gamma^2(-x \sin \theta + y \cos \theta)^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x \cos \theta + y \sin \theta}{\lambda} + \psi\right),$$

with ψ being the phase offset and can be included to approximate a complex cell's phase invariance or Gabor energy (Movellan, 1996).

The convolution of the input image $I(x, y)$ with each Gabor filter $G_{s,\theta}(x, y)$ results in a set of S1 feature maps, denoted as $FMS1_{s,\theta}(x, y)$. For 4 orientations (90, -45, 0 and 45 degrees) and 16 scales, like in this project, this yields 64 S1 feature maps, with each S1 variable being a list of length 16 containing the output at each scale.

C1 Layer: Local Max-Pooling

These neurons represent complex cells in V1, which exhibit larger receptive fields and some degree of shift and scale invariance. Each C1 feature map is generated by applying a local max-pooling operation over neighborhoods within S1 feature maps of the same orientation and within a "scale band." A scale band typically groups S1 maps from two adjacent scales.

The response of a C1 unit at position (x, y) for a given scale band s' and orientation θ is

$$FMC1_{s',\theta}(\mathrm{x,y}) = \max_{(u,v) \in N(x,y), s \in B_{s'}} \{FMC1_{s,\theta}(u, v)\},$$

where $N(x, y)$ is a local spatial neighborhood (pooling window) around (x, y), $B_{s'}$ is the set of S1 scales belonging to scale band s'.

Every two adjacent S1 scales (among 16 scales) are band-pooling for C1 processing, resulting in 8 scale bands.

S2 Layer: Prototype Matching

This layer corresponds to cells in higher visual areas (e.g., V4 or posterior inferotemporal (PIT) cortex, performing 2D convolution template matching against a set of pre-trained filters [3]. They compare patches extracted from C1 feature maps (across all orientations at a given scale band) to a set of learned "prototypes." The response is typically a Gaussian-like function of the distance between the C1 patch input and the prototype, effectively implementing a Radial Basis Function (RBF).

[3] https://github.com/wmvanvliet/pytorch_hmax.git

In particular, the response of an S2 unit at (x, y) for a given prototype P_m and C1 patch $X^s_{(x,y)}$ (from scale band s) is

$$FMS2^s_{(x,y),m} = \exp\left(-\beta \left\| X^s_{(x,y)} - P_m \right\|^2\right),$$

where ||·|| denotes the L_2 norm and

$$\beta = \frac{1}{2\sigma^2_{RBF}}$$

for some parameters σ_{RBF} defining the sharpness (width) of the tuning.

The output of this layer is a list of length 8, corresponding to the 8 scales of the C1 unit.

C2 Layer: Global Max-Pooling

This last layer resembles neurons in higher IT cortex, which often exhibit position and scale invariance across large portions of the visual field. For each prototype P_m, it takes the maximum response of its corresponding S2 units across all spatial positions and all scale bands:

$$C2_m = \max_{x,y,s}\left\{FMS2^s_{(x,y),m}\right\}.$$

For each input image, this layer outputs a vector of M entries, each representing the maximum response to prototype P_m.

This alternating sequence of S- and C-units allows the model to gradually build representations that are both highly specific to complex objects and robust to common visual transformations. The complexity of the features to which S-units are tuned increases up the hierarchy, as does the degree of invariance achieved by the C-units. The choice of the MAX operation for pooling, as opposed to, for instance, linear summation (SUM), is critical. Riesenhuber and Poggio argued that MAX pooling preserves feature specificity better in cluttered environments; a SUM operation, otherwise, might average out a strong feature signal with weaker, irrelevant ones, whereas a MAX operation ensures the strongest signal, indicative of the feature's presence, propagates (Riesenhuber & Poggio, 1999).

SVC Layer: Decision Making

In my pipeline, the final C2 feature vector is passed to a support vector classifier (SVC) (Cortes & Vapnik, 1995), a margin-maximizing binary classifier that finds the hyperplane in high-dimensional space which best separates the “Equal” and “Unequal” classes by balancing margin width against misclassification penalties.

I employ the scikit-learn implementation (Pedregosa et al., 2011), which interfaces with LIBSVM/LibLinear to provide both linear and kernelized SVCs and supports efficient grid-search over the regularization parameter and kernel parameters to optimize classification accuracy[4].

2.3. Experiment

In my framework, the training stage serves to establish that the model can reliably perform pure length discrimination: I train the SVM on classic CF stimuli and confirm its mastery on a held-out CF control set to ensure it truly learns to compare inner-line lengths rather than spurious cues. The testing stage then probes illusion susceptibility by applying this trained classifier to Müller–Lyer images—the degree to which its predictions deviate from veridical length judgments reveals the illusion's impact (**Fig. 4**). Crucially, to counter the concern that the model might be exploiting overall object size (including invariant crossfins) instead of the central line, I introduced two fin-configuration conditions— "same config" (constant fin geometry between the two lines) and "different" (variable fin geometry between the two lines) and varied the different image characteristics, as introduced in the previous section. By forcing the model to contend with changing fin configurations, we demonstrate that any systematic misclassification must stem from the illusion of the inner line itself, not from superficial differences in total stimulus extent.

[4] https://scikit-learn.org/

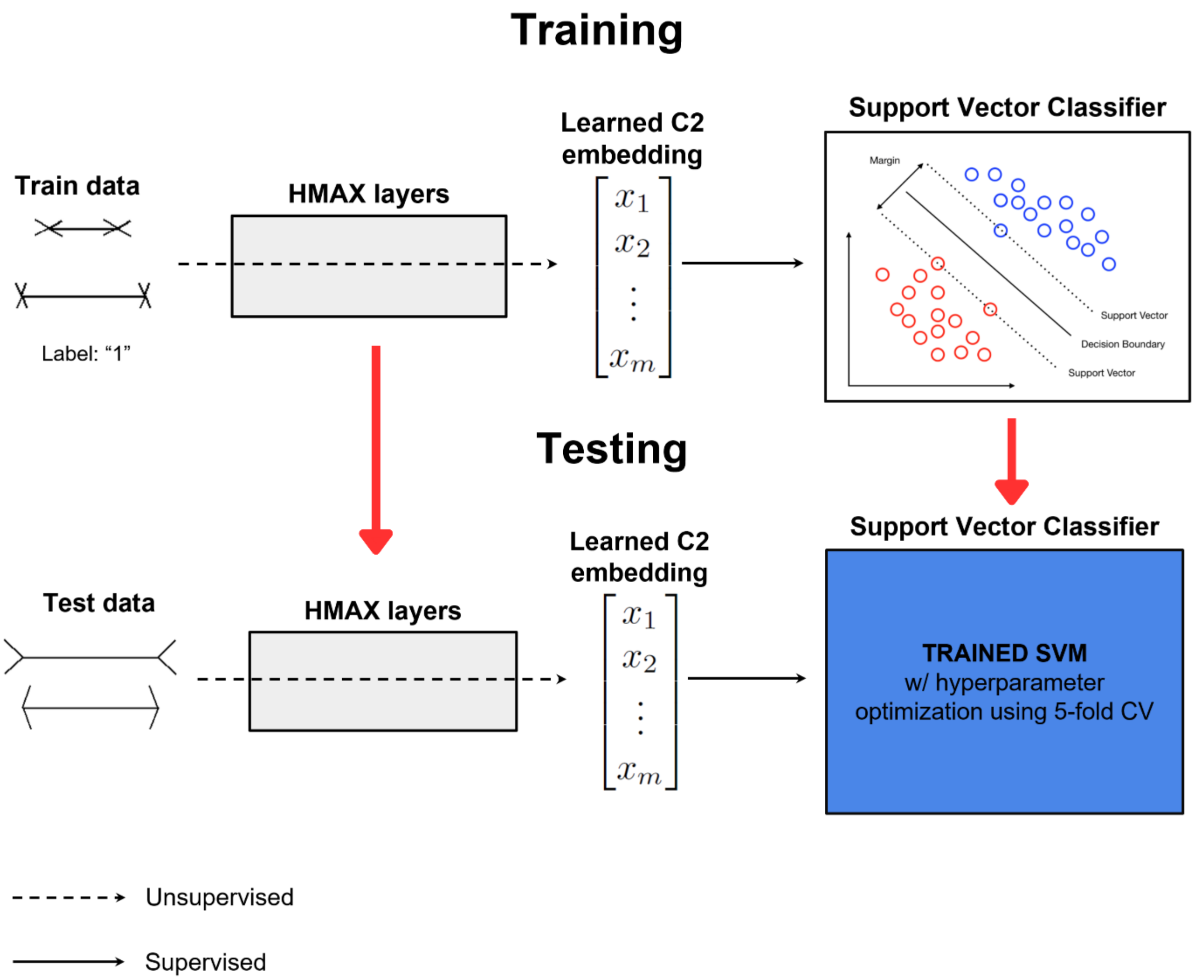

Figure 4. HMAX + SVM experimental pipeline

2.3.1. Training Stage

Each CF input image, converted to single-channel tensors and normalized to zero mean and unit variance, was represented as a high-level feature vector extracted from the C2 layer of the HMAX model (Riesenhuber & Poggio, 1999). Each vector $C2_m \in \mathbb{R}^M$. captures position- and scale-invariant prototype responses and serves as the input to the classifier.

The full dataset of 10,000 samples was partitioned into a training set (80%) and held-out test set (20%) (or "control CF test set") via Scikit-learn's stratified random sampling on the binary class labels. The training subset is used exclusively for hyperparameter tuning, while the test subset remains untouched until final evaluation.

For classification, I employed a linear support vector classifier (LinearSVC) from Scikit-learn, which implements the primal form of the soft-margin SVM using the LIBLINEAR solver (Fan et al., 2008). I perform an exhaustive grid search over the regularization parameter $C \in$

$\{0.001, 0.01, 0.1, 1, 10, 100\}$, convergence tolerance $\in \{10^{-4}, 10^{-3}, 10^{-2}\}$, and class-weight options $\{None, \text{“}balanced\text{”}\}$, with a maximum number of either 1,000 or 10,000 iterations for solver convergence. Hyperparameter optimization was conducted via 5-fold cross-validation on the training set: the data were partitioned into five equal folds, and for each candidate parameter combination, the mean classification accuracy across folds was computed. Evaluation was then performed over the held-out test set to confirm model accuracy and downstream analysis of the model performance.

2.3.2. Testing Stage

After confirming the model performance exceeds 80% on the “control CF test set”, in the testing stage, I applied the trained support vector classifier to the Müller–Lyer image set (or the “ML test set”) to measure illusory effect. ML images were similarly converted to single-channel tensors and normalized to zero mean and unit variance. The HMAX feature extractor was then re-initialized to extract the C2 features from the ML tensors. The pre-trained SVM model was then loaded from its saved joblib file and perform its prediction on the ML image set.

Finally, I computed classification accuracy from SVM outputs against true labels, and recorded the feature matrices, prediction arrays, and parameter tables for subsequent analysis.

3. Results & Discussion

During the training stage, the HMAX + SVM pipeline achieved strong length-classification performance, with the best accuracy of 93.2% during hyperparameter tuning (best paramerters: {'C': 1, 'gamma': 'scale', 'kernel': 'linear'}) and an accuracy of 82.8% on the “control CF test set”. As shown in **Fig. 5A**, the model was most accurate when the two lines were truly equal (87.9 %), and slightly less accurate when the top line was longer (78.7 %) or shorter (76.8 %). **Fig. 5B** breaks this down by class (Label 0 = “Equal,” Label 1 = “Unequal”) and fin-configuration condition. When both lines shared the same fin geometries, accuracy rose to 94.1 % on Label 0 and 82.6 % on Label 1, whereas accuracy under “Different” fin configurations dropped to 84.6 % (Label 0) and 75.3 % (Label 1). This modest but consistent gap confirms that holding fins constant simplifies the length-comparison task and validates my use of both “same” and “different” configurations to quantify how much the classifier truly learns to compare inner-line lengths rather than relying on overall object extent. In addition, the model achieves uniformly high accuracy on the different subsets of the held-out CF control set, with a modest fluctuation as stimulus parameters vary. For example, for the “same configuration” subset, accuracy remains above 85 % across all fin angles, ranging from 98.6 % at 15°–27° to 85.7 % (**Fig. 6A**). Similarly, stable performance was observed for fin lengths, fluctuating narrowly between 91.0 % and 98.4 % (**Fig. 6B**). In **Fig. 6C**, showing results across different vertical distance setting within the “control CF test set”, vertical line separation yields accuracies from 79.8 % to 85.7 %. **Fig. 6D** shows that shaft length variations produce a consistent 81.0 %–92.2 % range within the

"EQUAL" subset. These small variations confirm that the classifier's length-comparison ability is robust to changes in fin geometry and spatial arrangement.

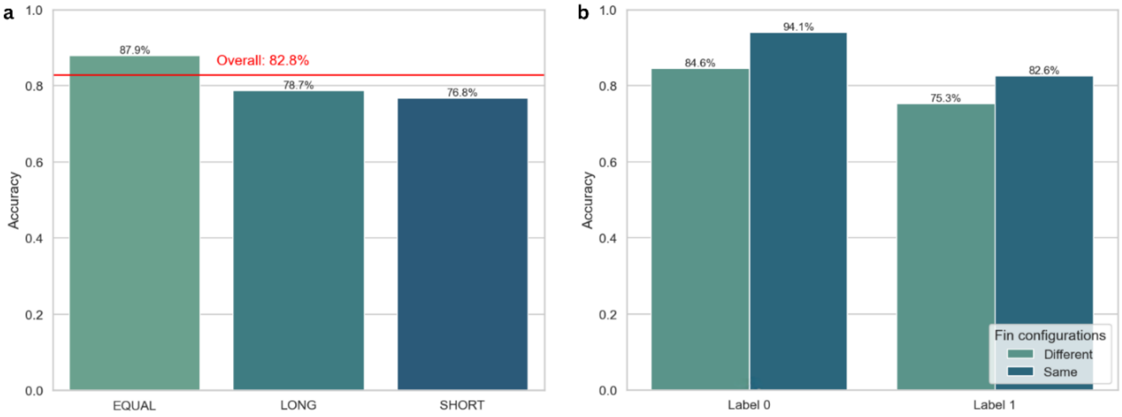

Figure 5. (A) Prediction accuracy on the "control CF test set" when the two parallel lines are equal (EQUAL), the top line is longer (LONG) and shorter (SHORT). (B) Accuracy slightly decreases when the two lines do not share the same fin configurations (fin angle & fin length).

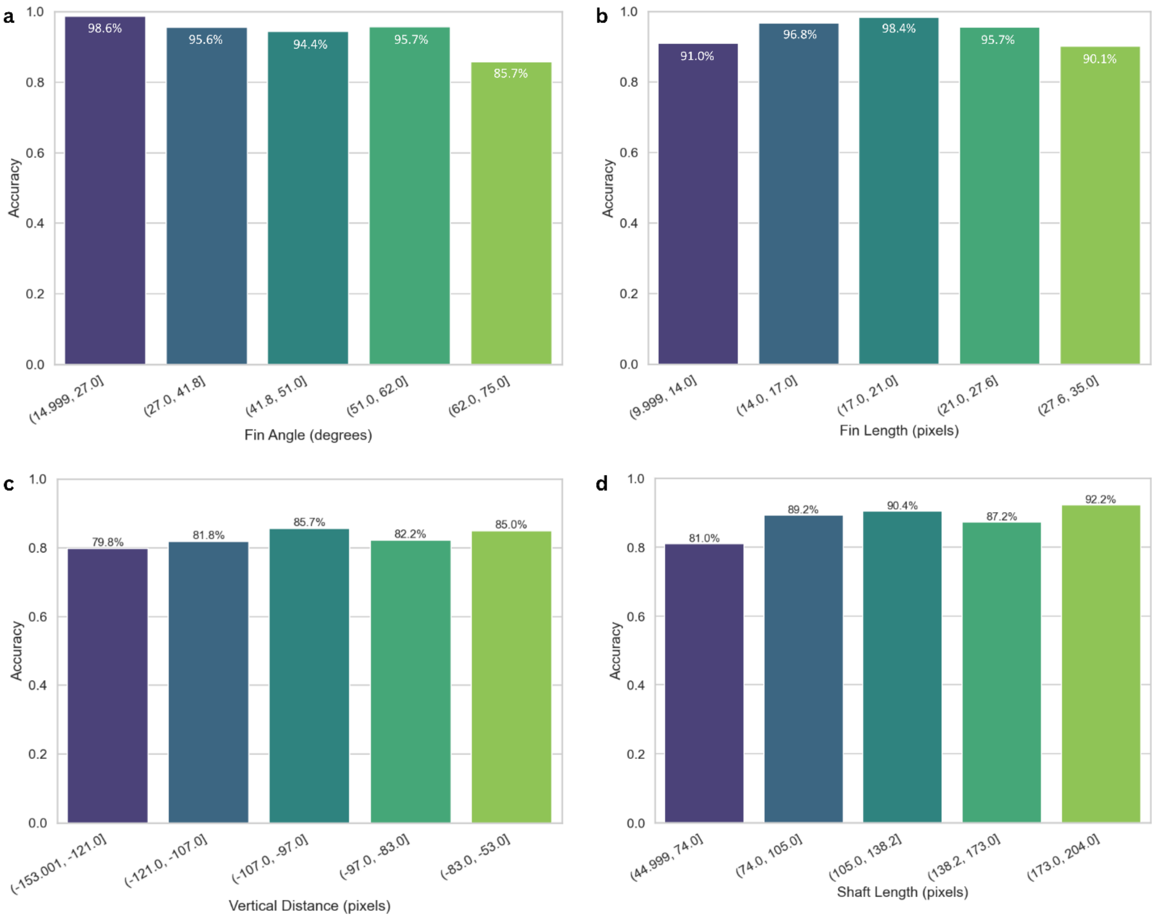

Figure 6. Prediction accuracy "control CF test set" shows modest changes by varying (a) fin angle, (b) fin length, (c) vertical distance between the two lines, and (d) shaft length.

Having first confirmed robust length-comparison performance on the CF control stimuli, I then evaluated the same HMAX + SVM pipeline on the Müller–Lyer test set to probe its susceptibility to the classic geometric illusion. When the Müller–Lyer stimuli preserve consistent arrowhead orientation on both lines—that is, both fins point either inward or outward—the classifier still "sees" length almost correctly, achieving 82.6 % accuracy (**Fig. 7**). This suggests that when the global configuration remains uniform, the model can largely ignore the arrowheads and focus on the inner shaft as it was trained to do so. However, when the two lines bear opposing arrowhead directions, accuracy plunges to just 31.0 %, well below the 50 % chance level. Such a dramatic reversal indicates not random guessing but a systematic misreading of the illusory cues: the model appears to interpret one of the lines to be "longer-looking", even when it is not. In other words, conflicting arrow directions not only disrupt the classifier's ability to abstract away from whole-object length but actively invert its judgments, revealing that its decision boundary is heavily influenced by the conflicting stimulus extent—the opposing directions of the arrowheads—rather than by a veridical comparison of central shaft lengths alone. This pattern mirrors the classic human susceptibility to the Müller–Lyer illusion and suggests that my HMAX + SVM pipeline might encode the very perceptual biases that give rise to the illusion.

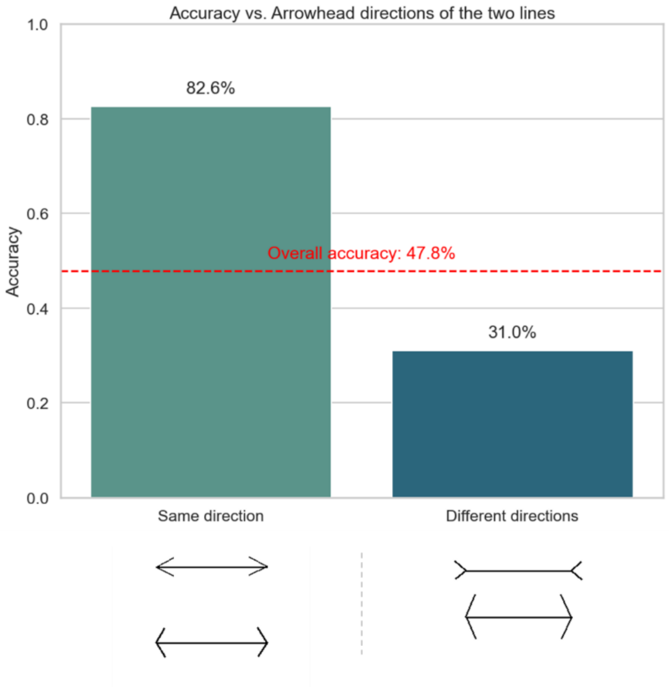

Figure 7. Müller–Lyer Test-Set Accuracy by Arrowhead Configuration. The trained HMAX + SVM model achieves 82.6 % accuracy when both lines share the same arrowhead orientation but drops to 31.0 % when arrowheads point in opposite directions, falling well below the 50 % chance level (red dashed line).

The consistency of fin geometry does not determine the Müller–Lyer illusion in my HMAX + SVM pipeline. When arrowheads align (both inward or both outward), matching fin sizes significantly boosts accuracy—from 77.3 % with differing fins to 93.4 % when fins are held constant—as expected from results on the "control CF test data" (**Fig. 5B**). On the other hand, the core illusion persists regardless of fin matching: under opposing arrowhead directions, the classifier's accuracy remains near 31 % in both "same configuration" and "different configuration" subsets. In other words, while uniform fin geometry can aid pure length judgments when the global cue is nonillusory, it cannot override the systematic bias induced by conflicting arrowheads. This invariance confirms that my model faithfully reproduces the Müller–Lyer illusion independent of low-level fin-size discrepancies.

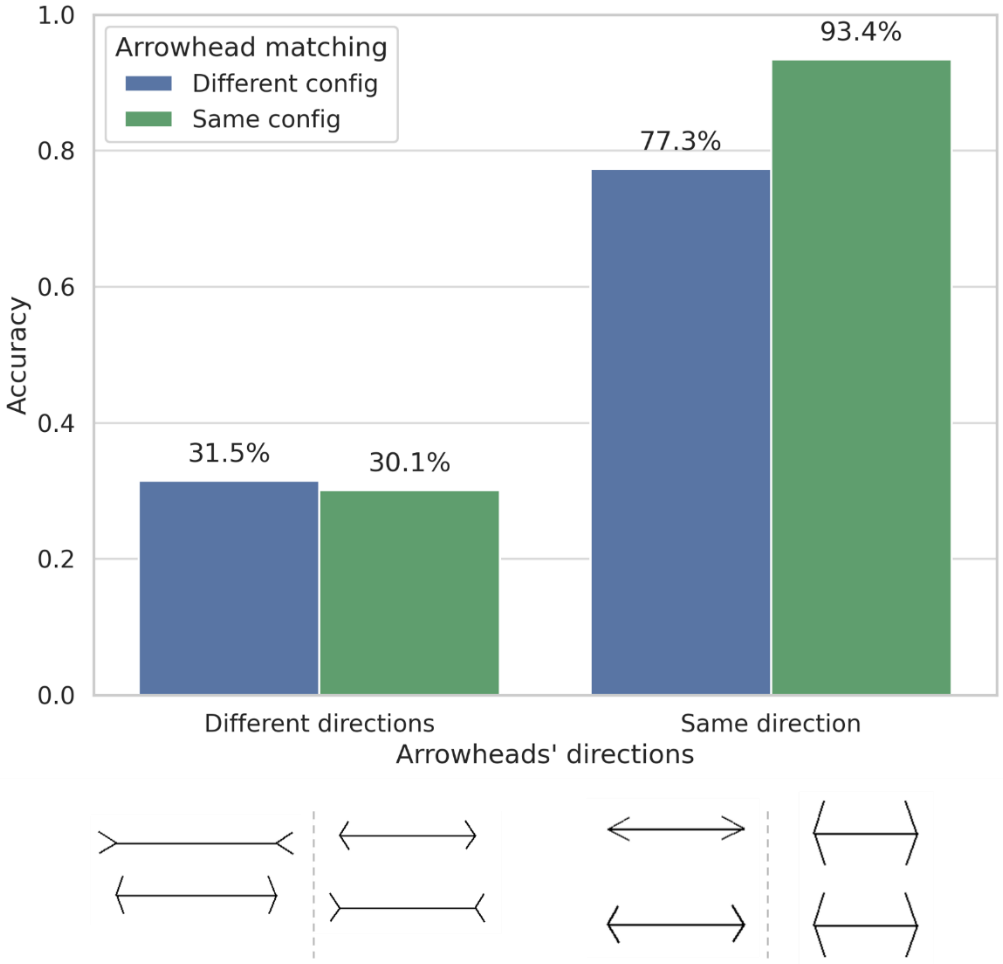

Figure 8. Changes in accuracy in different arrowhead (fin) geometry subsets

The modest difference in accuracy between placing inward-pointing arrowheads on the top line (26.8 %) versus the bottom line (35.1 %) (**Fig. 9**) suggests a slight positional asymmetry in how the model processes the illusion: it is marginally more prone to misjudge length when the illusion-inducing fins appear above the fixation axis. However, both conditions remain well below chance, indicating that the Müller–Lyer effect dominates irrespective of whether the “shortening” cues occur on the top or bottom.

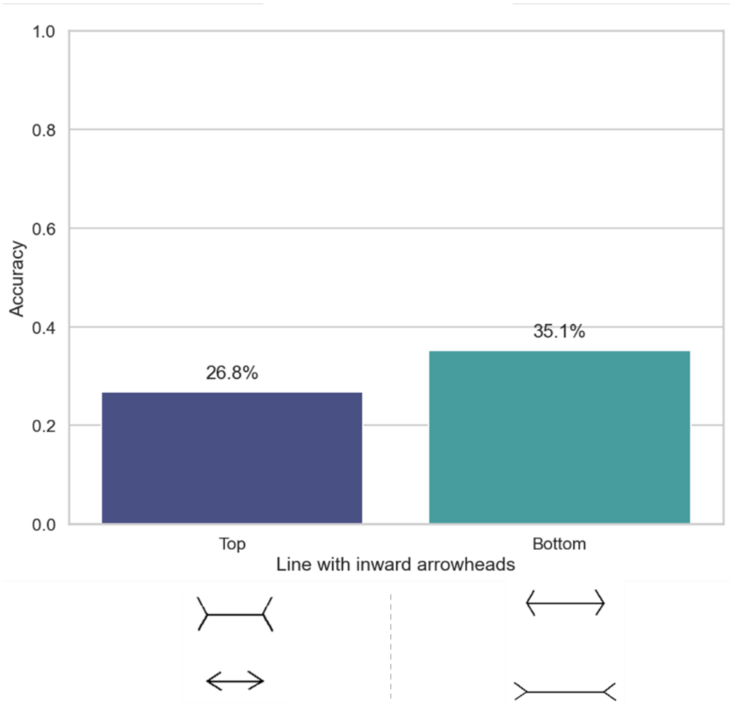

Figure 9. Accuracy vs. positional asymmetry

My analysis reveals that the Müller–Lyer illusion in the HMAX + SVM model is driven primarily by two factors—fin angle and shaft length—while fin length and vertical distance exert negligible influence. Specifically, fin angle produces a pronounced U-shaped effect: classification accuracy dips to its lowest (≈20 %) at mid-range angles (≈39°–51°) where the illusion is strongest, then recovers at both shallow (31.2 % at 15°–26°) and steep (45.8 % at 63°–75°) orientations. Shaft length shows an even more dramatic impact: performance rises sharply from near zero (3.8 % for the shortest shafts) to robust accuracy (73.2 % for the longest shafts), indicating that long central segments can overpower the fin-induced distortion. In contrast, fin length remains flat (≈30 %–33 % accuracy) and vertical separation shows only minor, non-systematic fluctuations (≈28 %–32 %), confirming that these parameters do not modulate the illusion in my model. Human psychophysical studies confirm that both the angular orientation of fins and the relative shaft–fin ratio critically modulate the Müller–Lyer illusion. For example, several psychological experiments demonstrated that illusion magnitude is largest at intermediate angles and attenuating at very shallow or steep orientations (Pressey & Martin, 1990). Similarly, changes in the inter-fin shaft length (the distance between fin bases) can even reverse the illusion’s direction, with short central shafts producing maximal misperception and long shafts markedly reducing or inverting the effect (Dragoi & Lockhead, 1999). These results suggested that despite their vastly different substrates, biological and artificial vision might share sensitivity to the principal determinants of the Müller–Lyer illusion in the same manner.

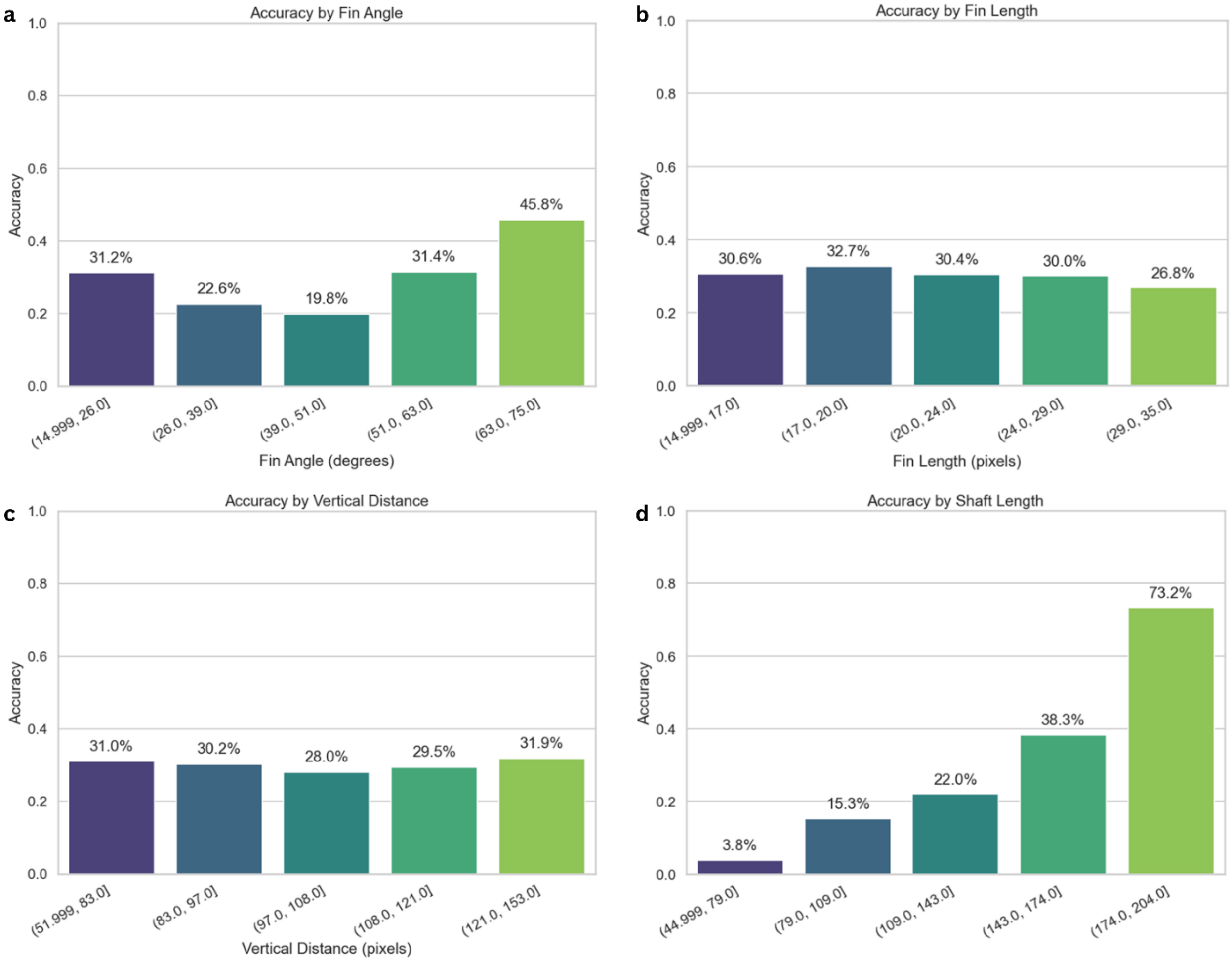

Figure 10. Illusory effect shows strong dependence on (a) fin angle and (d) shaft length but less on (b) fin length and (c) vertical distance between the two lines.

Conclusion

In this study, I demonstrated that a hierarchical vision pipeline—combining the biologically inspired HMAX feature extractor with a linear SVM classifier—naturally reproduces the Müller–Lyer illusion despite being trained exclusively on veridical length comparisons. The model's illusory judgments hinge critically on fin angle and shaft–fin ratio: mid-range fin angles and short central shafts produce maximal misperception, whereas extreme fin orientations and sufficiently long shafts attenuate the effect. In contrast, fin length and line separation exert little influence, underscoring that arrowhead configuration and relative segment length are the principal drivers of illusion susceptibility in both artificial and human vision. These results validate the presented experimental framework as a useful tool for probing perceptual biases, and they point toward future work exploring mechanistic origins—through layer-wise ablation, recurrent dynamics, or spiking models—and extending this approach to broader visual illusion classes.

References

Chikkerur, S. & Poggio, T. (2011). *Approximations in the HMAX Model* (MIT-CSAIL-TR-2011-021CBCL-298). CSAIL Technical Reports.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. https://doi.org/10.1007/BF00994018

Crouzet, S. M., & Serre, T. (2011). What are the Visual Features Underlying Rapid Object Recognition? *Frontiers in Psychology*, *2*, 326. https://doi.org/10.3389/fpsyg.2011.00326

Deng, L., Wang, Y., Liu, B., Liu, W., & Qi, Y. (2018). Biological modeling of human visual system for object recognition using GLoP filters and sparse coding on multi-manifolds. *Machine Vision and Applications*, *29*(6), 965–977. https://doi.org/10.1007/s00138-018-0928-9

Dragoi, V., & Lockhead, G. (1999). Context-dependent changes in visual sensitivity induced by Müller–Lyer stimuli. *Vision Research*, *39*(9), 1657–1670. https://doi.org/10.1016/S0042-6989(98)00198-9

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, *36*(4), 193–202. https://doi.org/10.1007/BF00344251

Gomez-Villa, A., Martín, A., Vazquez-Corral, J., & Bertalmío, M. (2018). *Convolutional Neural Networks Deceived by Visual Illusions* (arXiv:1811.10565). arXiv. https://doi.org/10.48550/arXiv.1811.10565

Gomez-Villa, A., Martín, A., Vazquez-Corral, J., Malo, J., & Bertalmío, M. (2019). *Synthesizing Visual Illusions Using Generative Adversarial Networks* (arXiv:1911.09599). arXiv. https://doi.org/10.48550/arXiv.1911.09599

Hermann, L. (1870). Eine Erscheinung simultanen Contrastes. *Pflüger, Archiv für die Gesammte Physiologie des Menschen und der Thiere*, *3*(1), 13–15. https://doi.org/10.1007/BF01855743

Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, *160*(1), 106–154. https://doi.org/10.1113/jphysiol.1962.sp006837

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, *1*(4), 541–551. https://doi.org/10.1162/neco.1989.1.4.541

Li, Y., Wu, W., Zhang, B., & Li, F. (2015). Enhanced HMAX model with feedforward feature learning for multiclass categorization. *Frontiers in Computational Neuroscience*, *9*. https://doi.org/10.3389/fncom.2015.00123

Louie, J. (2003). *A Biological Model of Object Recognition with Feature Learning*. Department of Electrical Engineering and Computer Science. http://hdl.handle.net/1721.1/29678

Movellan, J. R. (n.d.). *Tutorial on Gabor Filters*.

Pressey, A., & Martin, N. S. (1990). The effects of varying fins in M□ller-Lyer and Holding illusions. *Psychological Research*, *52*(1), 46–53. https://doi.org/10.1007/BF00867211

Rangayyan, R. M., Zhu, X., Ayres, F. J., & Ells, A. L. (2010). Detection of the optic nerve head in fundus images of the retina with Gabor filters and phase portrait analysis. *Journal of Digital Imaging*, *23*(4), 438–453. https://doi.org/10.1007/s10278-009-9261-1

Riesenhuber, M., & Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature Neuroscience*, *2*(11), 1019–1025. https://doi.org/10.1038/14819

Rostamkhani, M., Ansari, B., Sabzevari, H., Rahmani, F., & Eetemadi, S. (2024). *Illusory VQA: Benchmarking and Enhancing Multimodal Models on Visual Illusions* (arXiv:2412.08169). arXiv. https://doi.org/10.48550/arXiv.2412.08169

Serre, T. (2004). Realistic Modeling of Simple and Complex Cell Tuning in the HMAX Model, and Implications for Invariant Object Recognition in Cortex. *Massachusetts Institute of Technology*, *(AI Memo 2004-017; CBCL Memo 239)*.

Serre, T. (2014). Hierarchical Models of the Visual System. In D. Jaeger & R. Jung (Eds.), *Encyclopedia of Computational Neuroscience* (pp. 1–12). Springer New York. https://doi.org/10.1007/978-1-4614-7320-6_345-1

Serre, T., Oliva, A., & Poggio, T. (2007). A feedforward architecture accounts for rapid categorization. *Proceedings of the National Academy of Sciences*, *104*(15), 6424–6429. https://doi.org/10.1073/pnas.0700622104

Susan, S. (2024). Neuroscientific insights about computer vision models: A concise review. *Biological Cybernetics*, *118*(5–6), 331–348. https://doi.org/10.1007/s00422-024-00998-9

Trestman, M. (2014). The modal breadth of consciousness. *Philosophical Psychology*, *27*(6), 843–861. https://doi.org/10.1080/09515089.2013.776476

Ullman, T. (2024). *The Illusion-Illusion: Vision Language Models See Illusions Where There are None* (arXiv:2412.18613). arXiv. https://doi.org/10.48550/arXiv.2412.18613

Ward, E. J. (2019). *Exploring Perceptual Illusions in Deep Neural Networks*. https://doi.org/10.1101/687905

Zhang, H., Matsuzaki, K., & Yoshida, S. (2024). Assessing Brain-Like Characteristics of DNNs With Spatiotemporal Features: A Study Based on the Müller-Lyer Illusion. *IEEE Access*, *12*, 147192–147208. https://doi.org/10.1109/ACCESS.2024.3475478

Zöllner, F. (1860). Ueber eine neue Art von Pseudoskopie und ihre Beziehungen zu den von Plateau und Oppel beschriebenen Bewegungsphänomenen. *Annalen Der Physik*, *186*(7), 500–523. https://doi.org/10.1002/andp.18601860712