Decoding Neural Networks: Analyzing Alignment Between MidLevel Vision Feature Detection and Human Perception

Yufei Chen* Dirk Bernhardt Walther Soehee Han

Abstract

This study investigates the ability of the first convolutional layer of the VGG16 neural network to capture edge orientations, a key aspect of mid-level vision in humans. By analyzing the feature maps generated by VGG16, we assess whether these maps reflect orientation sensitivity similar to human visual perception. We apply various filtering methods to input images and correlate the resulting edge orientations with the activations of VGG16's feature maps. The findings aim to enhance the interpretability of convolutional neural networks and provide insights into how these models align with biological vision systems.

1.1 Introduction

Human visual perception relies on mid-level representations—a critical stage of processing that bridges raw sensory input (low-level vision) and object recognition (high-level vision). This intermediate stage, termed mid-level vision, organizes fragmented visual data into coherent surfaces and structures, handling occlusion, illumination, and orientation to construct a stable perceptual world (Nakayama et al., 1995). Fundamental attributes like symmetry, curvature, and orientation are central to this process, enabling humans to infer object boundaries, parse ambiguous motion (Wallach, 1935), and even resolve illusions like the Necker cube or White's brightness effect (Kanizsa, 1979). These mechanisms suggest that vision is not monolithic but a "conglomerate of functions" (Harvard Vision Lab), each operating with specialized autonomy.

Yet, as artificial neural networks increasingly model visual tasks, a key question arises: *Do these systems develop internal representations analogous to mid-level vision in humans?* While CNNs excel at object recognition and classification, their alignment with human perceptual hierarchies – particularly in encoding mid-level features – remains underexplored. For instance, do early convolutional layers implicitly learn orientation or symmetry detectors akin to human biological vision? If so, that could possibly suggest inherent similarities between the ways human brains process the image and the way neural networks do. Furthermore, addressing this could reveal how neural nets compensate for the absence of evolved cortical mechanisms and whether their feature extraction mirrored human psychophysical biases.

This study investigates the correspondence between mid-level visual attributes and feature maps in CNNs, focusing on VGG16's initial convolutional layer. We analyze how filters specifically capture orientation through correlation analysis between edge orientations in input images and activated feature maps. By quantifying these relationships, we aim to:

1. Evaluate architectural biases: Do CNNs naturally prioritize mid-level features, or must they *learn* them from data?
2. Compare biological and artificial systems: How do CNNs representations diverge from or converge with human mid-level vision?
3. Inform model interpretability: Can mid-level principles guide more human-aligned network designs?

Our findings contribute to the broader discourse on neural network interpretability and biologically inspired vision systems. By dissecting the parallels between artificial and human mid-level processing, we shed light on the "black box" of deep learning while advancing models that better mimic the robustness of biological perception.

1.2 Literature Review

Human vision follows a hierarchical structure where early visual areas, such as V1, detect simple features like orientation, while later stages integrate these into coherent shapes (Wagemans et al., 2012). Hubel and Wiesel's (1962) classic findings showed that neurons in V1 respond selectively to edges of specific orientations—highlighting orientation as a core component of low- and mid-level vision. In convolutional neural networks (CNNs), particularly those trained on object recognition tasks, early layers are known to develop filters that resemble Gabor filters—similar to orientation-selective neurons in V1 (Khaligh-Razavi & Kriegeskorte, 2014). This resemblance has led to growing interest in whether CNNs exhibit similar mid-level feature processing as human vision, such as sensitivity to symmetry, curvature, and edge alignment.

Several studies have attempted to map internal CNN activations to interpretable features. Bau et al. (2017) identified units in CNNs that activate in response to visual concepts including symmetry and curvature, suggesting a partial overlap with mid-level visual processing in humans. Yet, this overlap is inconsistent. Biscione & Bowers (2023), for example, found that DNNs often fail to reproduce Gestalt grouping principles in intermediate layers, implying a lack of biologically-inspired mid-level organization.

This mixed evidence raises an important question: do CNNs naturally learn to detect fundamental mid-level features like edge orientation, or must they be explicitly trained to do so? Furthermore, if some filters do capture these properties, which filters are they? Our study addresses this by focusing specifically on the first convolutional layer of VGG16. By correlating orientation-based input images with feature map activations, we aim to identify whether certain filters inherently specialize in detecting edge orientations—a property central to both biological and artificial vision systems. This analysis will provide insight into the extent to which artificial neural networks can serve as models for human perception and highlight areas where improvements are needed to enhance their biological plausibility.

1.3 Background

Feedforward neural networks operate through a feedforward mechanism, where input data is passed through a series of layers to produce an output, such as a classification. Each layer consists of neurons that apply mathematical transformations to the input data, extracting features at different levels of abstraction. In the context of image processing, early layers typically detect low-level features such as edges, textures, and orientations, while deeper layers capture higher-level features like shapes, objects, and complex patterns.

Convolutional Neural Networks (CNNs) are a specialized type of neural network designed to process grid-like data, such as images. They use convolutional layers to apply filters to the input image, producing feature maps that highlight specific visual attributes. These feature maps are then passed through activation functions to introduce non-linearity, followed by pooling layers to reduce spatial dimensions

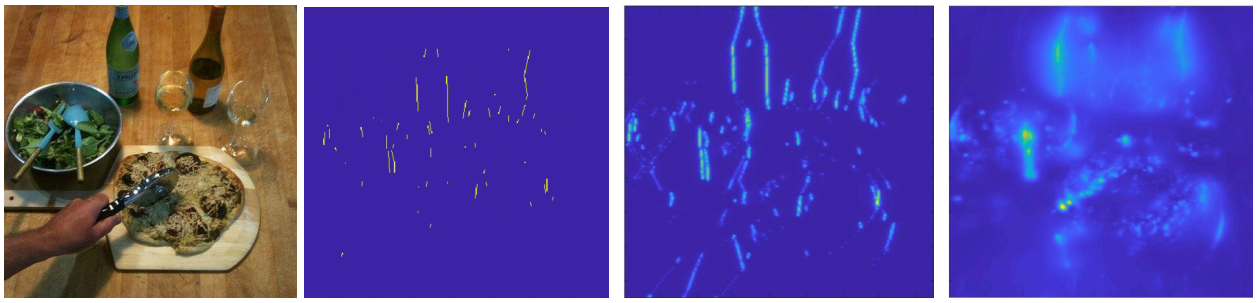
and enhance translational invariance. The final layers of a CNN are typically fully connected layers, which combine the extracted features to make predictions.

VGG16 is the deep CNN architecture used in this study. It is a widely recognized model developed by the Visual Geometry Group at the University of Oxford. VGG16 consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. The network is known for its simplicity and depth, using small 3x3 convolutional filters stacked in multiple layers to capture intricate features. VGG16 has been extensively used in computer vision research and has achieved state-of-the-art performance on benchmark datasets such as ImageNet.

Each convolutional layer in VGG16 produces a set of feature maps that represent different aspects of the input image. The first convolutional layer, in particular, is critical for detecting low-level features such as edges, orientations, and textures. This layer contains 64 feature maps, each corresponding to a specific filter applied to the input image. By analyzing these feature maps, we can gain insights into how the network processes visual information and whether its feature detection mechanisms align with human perception.

Different Filter Methods

To systematically evaluate the orientation sensitivity of VGG16's first layer, we employ three distinct filtering approaches. First, the contour filter method tests the network's sensitivity to continuous contours, including illusory boundaries that humans perceive as unified shapes. Second, the line drawing filter method isolates edge responses by reducing images to simplified 1-pixel-wide contours, allowing us to examine how individual feature maps respond to canonical edge orientations. Finally, the photo filter method assesses the robustness of orientation coding under naturalistic conditions by using real-world photographs with complex textures and lighting variations. These methods are depicted in Figure 1. Together, these methods provide a comprehensive framework for analyzing how CNNs encode low-level visual features and whether their representations mirror the hierarchical organization observed in biological vision systems.



(a) (b) (c) (d)

Figure 1: (a) original photo (b) contour filter method for vertically oriented edge detection (c) line drawing filter method for vertically oriented edge detection (d) photo filter method for vertically oriented edge detection

1.4 Research Question: Do different filter methods reveal whether certain feature maps in the first layer of VGG16 are better at detecting specific edge orientations?

2 Methodology

For each of the three filter methods (line drawing, contour, and photo filters), we employ a systematic approach to evaluate how VGG16's first convolutional layer encodes edge orientations. The process involves: (1) extracting ground-truth edge orientations from input images, (2) generating feature maps from VGG16's first layer, and (3) quantifying the correspondence between neural network activations and physical edge orientations through correlation analysis. This tripartite methodology allows us to assess orientation sensitivity across controlled and naturalistic stimuli. The entire experiment is conducted on the ImageNet dataset, where 14600 are selected to test the research question.

2.1 Edge Orientation Computation

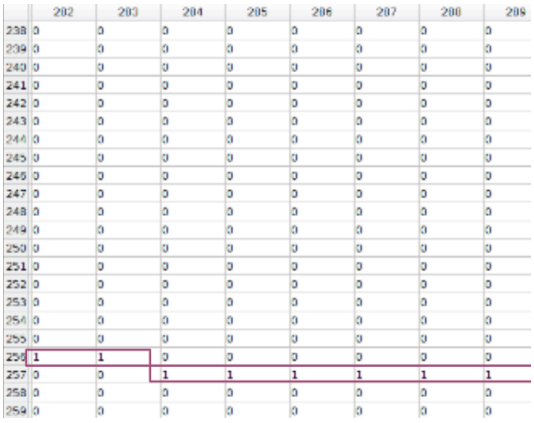
Ground Truth Establishment

For each input image, we first compute ground-truth edge orientations using three specialized edge detection approaches:

1. Edge Extraction: Use an edge detection toolbox[1] to extract edges from the original image, and we have three different methods: **photo filter, line drawing filter, and contour filter.** The output is a set of edges, each characterized by its orientation.
2. Orientation Binning: Divide the edge orientations into 8 bins, each representing a 25.5° interval (since 180°/ 8 = 25.5°). This circular binning system (where 0° = 180° = vertical) ensures comprehensive coverage of all possible edge orientations. The vertical orientation bin (Bin 1) serves as our primary reference axis due to its perceptual significance in both biological and artificial vision systems.
    - Bin 1: (-11.25°, 11.25°)
    - Bin 2: (11.25°, 33.75°)
    - ...
    - Bin 8: (168.75°, 191.25°)
3. OribinMap Generation: For each edge detection method and each orientation bin, we generate specialized OribinMaps (Orientation-Binned Maps) through the following process:
    - Spatial Alignment:
        - Contour method: 425 x 425 resolution
        - Photo/Line Drawing methods: 512 x 512 resolution (Maintaining each filter's native output dimensions preserves edge localization accuracy)
    - Binary encoding: The OribinMap is filled with binary values. For every detected edge in the original image, if the edge's orientation falls within the current bin's angular range (e.g., Bin 1: ±11.25° from vertical), set the corresponding pixel to 1. All other pixels remain 0 (no edge or edge outside the bin).

    Figure 2 shows an example of a generated OribinMap.

[1] BwLab, University of Toronto, *MLV Toolbox (Mid-level Vision Toolbox)*, computer software, 2023, GitHub repository, https://github.com/bwlabToronto/MLV_toolbox.



	202	203	204	205	206	207	208	209
238	0	0	0	0	0	0	0	0
239	0	0	0	0	0	0	0	0
240	0	0	0	0	0	0	0	0
241	0	0	0	0	0	0	0	0
242	0	0	0	0	0	0	0	0
243	0	0	0	0	0	0	0	0
244	0	0	0	0	0	0	0	0
245	0	0	0	0	0	0	0	0
246	0	0	0	0	0	0	0	0
247	0	0	0	0	0	0	0	0
248	0	0	0	0	0	0	0	0
249	0	0	0	0	0	0	0	0
250	0	0	0	0	0	0	0	0
251	0	0	0	0	0	0	0	0
252	0	0	0	0	0	0	0	0
253	0	0	0	0	0	0	0	0
254	0	0	0	0	0	0	0	0
255	0	0	0	0	0	0	0	0
256	1	1	0	0	0	0	0	0
257	0	0	1	1	1	1	1	1
258	0	0	0	0	0	0	0	0
259	0	0	0	0	0	0	0	0

Figure 2: A section of an OribinMap generated using MatLab

2.2 Feature Map Extraction from VGG16

The first convolutional layer of VGG16 consists of 64 filters with 3×3 kernels, stride 1, and padding 1. This layer processes input images (224×224×3) to produce 64 feature maps (224×224) through convolution followed by ReLU activation. The process is as follows.

1. Feature Map Extraction: Pass the original image through the VGG16 network and extract the 64 feature maps from the first convolutional layer. Each feature map $F_k \in R^{DxD}$ ($k \in \{1, \ldots, 64\}$) is a matrix encoding localized feature responses. The intensity values in F_k quantify the relative activation strength of specific convolutional filters at each spatial location. Higher values indicate stronger filter responses to local image patterns within that filter's receptive field. Each feature map can be visualized as depicted in Figure 3. These visualizations reveal filter-specific sensitivities to low-level features - edge-selective filters show strong activations along corresponding orientations, while texture-sensitive filters exhibit distributed response patterns. Notably, first-layer visualizations primarily capture basic visual elements (edges, color transitions) rather than higher-order semantic features.

Figure 3: First five feature maps (of 64 total) extracted from VGG16's initial convolutional layer. Brighter pixels indicate stronger filter activations, revealing localized responses to low-level image features. Each map corresponds to a distinct learned filter's activation pattern when processing the input image.

2.3 Compare Feature Maps with OribinMaps

We quantify orientation selectivity in VGG16's first layer by computing pixel-wise correlations between each of the 64 feature maps and the 8 binarized OribinMaps. This identifies which convolutional filters respond preferentially to specific edge orientations.

1. Intensity Matrix Extraction: For each feature map $F_k \in R^{DxD}$ ($k \in \{1, \ldots, 64\}$), extract the intensity matrix preserving spatial correspondence with the original image. The matrix elements $F_k(i, j)$ represent the activation strength of filter k at pixel location (i, j), normalized to zero mean and unit variance for comparative analysis.
2. Correlation Metic: For each orientation bin $b \in \{1, .., 8\}$, we compute the Pearson correlation coefficient $\rho_{k,b}$ between the flattened feature map f_k and OribinMap σ_b:

$$\rho_{k,b} = \frac{\sum_{p=1}^{N} (f_k(p) - \overline{f_k})(\sigma_b(p) - \overline{\sigma_b})}{\sqrt{\sum_{p=1}^{N} (f_k(p) - \overline{f_k})^2} \sqrt{\sum_{p=1}^{N} (\sigma_b(p) - \overline{\sigma_b})^2}}$$

where $N = n \times m$ pixels, $\overline{f_k}$ and $\overline{\sigma_b}$ denotes mean intensities, and p indexes flattened spatial positions. A high $\rho_{k,b}$ value indicates filter k strongly responds to edges in bin b's orientation range.

The optimal filter for bin b is identified as $argmax_k \rho_{k,b}$. This filter exhibits the strongest statistical alignment with the target edge orientation. The process iterates across all 8 bins to construct a complete orientation selectivity profile for the first convolutional layer.

The above process is repeated for all three filter methods. However, for the contour method, Gaussian blur is applied to the OribinMaps before the correlation is computed between the feature map intensities and OribinMaps.

3 Results and Analysis

The purpose of this study is to analyze orientation selectivity in VGG16's first convolutional layer by comparing neural activations across three edge detection methods (line drawing, photo, and contour filters) applied to 14,600 ImageNet images. For each of the 8 orientation bins, we computed the mean correlation coefficients between all 64 feature maps and their corresponding OribinMaps, averaged across all valid images. The plotted results (Figures 4-11) show these averaged correlations for all three methods, revealing both similarities and differences in how each filter type influences orientation encoding.

3. 1 Correlation Interpretation

Positive correlation ($\rho > 0$) indicates the feature map's activation intensity increases where edges of orientation b appear in the OribinMap. This suggests the filter k is directly responsive to edges in bin b's orientation range and is likely tuned to detect that specific edge orientation. On the other hand, negative correlation ($\rho < 0$) indicates an inverse relationship between the feature map activations and the OribinMap edges, where higher filter responses occur in regions without edges of orientation b. This implies the filter may respond preferentially to orthogonal orientations, like a vertical edge filter suppressing horizontal edges, and the activation pattern could represent center-surround inhibition. While theoretically meaningful, negative correlation actually complicate orientation bin assignment for three main reasons:

1. Bin Misalignment: Strong negative responses to bin b edges may suggest the filter is actually tuned to orthogonal orientations ($b \pm 90°$), which would properly belong to different bins
2. Contrast Ambiguity: Negative weights may simply detect inverted edge polarities (dark-light vs light-dark) of the same orientation.
3. Interpretation Conflict: Center-surround inhibition patterns don't cleanly map to discrete orientation bins

Lastly, when the correlation is near-zero, the filter is implied to be insensitive to the tested orientation, and it could be tuned to other visual features such as textures or colors.

Bin	Line Draw Rank 1	Line Draw Rank 2	Line Draw Rank 3	Photo Rank 1	Photo Rank 2	Photo Rank 3	Contour Rank 1	Contour Rank 2	Contour Rank 3
1	FM 60 (0.0701)	FM 3 (0.0444)	FM 48 (0.0433)	FM 48 (0.0091)	FM 62 (0.0061)	FM 60 (0.0057)	FM 48 (0.0092)	FM 62 (0.0078)	FM 19 (0.0062)
2	FM 60 (0.0567)	FM 48 (0.0421)	FM 3 (0.0371)	FM 48 (0.0117)	FM 60 (0.0100)	FM 35 (0.0080)	FM 48 (0.0070)	FM 62 (0.0060)	FM 19 (0.0047)
3	FM 60 (0.0478)	FM 48 (0.0425)	FM 35 (0.0391)	FM 48 (0.0139)	FM 60 (0.0136)	FM 35 (0.0116)	FM 48 (0.0057)	FM 62 (0.0048)	FM 19 (0.0040)
4	FM 35 (0.0618)	FM 48 (0.0589)	FM 60 (0.0558)	FM 48 (0.0093)	FM 60 (0.0057)	FM 35 (0.0052)	FM 48 (0.0076)	FM 62 (0.0064)	FM 19 (0.0050)
5	FM 35 (0.0851)	FM 48 (0.0743)	FM 60 (0.0594)	FM 62 (0.0050)	FM 48 (0.0037)	FM 10 (0.0027)	FM 48 (0.0107)	FM 62 (0.0092)	FM 19 (0.0072)
6	FM 35 (0.0621)	FM 48 (0.0588)	FM 60 (0.0556)	FM 48 (0.0104)	FM 60 (0.0058)	FM 35 (0.0052)	FM 48 (0.0077)	FM 62 (0.0067)	FM 19 (0.0050)
7	FM 60 (0.0489)	FM 48 (0.0433)	FM 35 (0.0395)	FM 48 (0.0150)	FM 60 (0.0138)	FM 35 (0.0116)	FM 48 (0.0059)	FM 62 (0.0050)	FM 19 (0.0039)
8	FM 60 (0.0578)	FM 48 (0.0453)	FM 3 (0.0379)	FM 48 (0.0121)	FM 60 (0.0099)	FM 35 (0.0080)	FM 48 (0.0074)	FM 62 (0.0063)	FM 19 (0.0052)

Table 1. Top three feature maps (FM) with the highest averaged correlations for each filter method across the eight bins. The highlighted cells are ones with the highest correlations along each column.

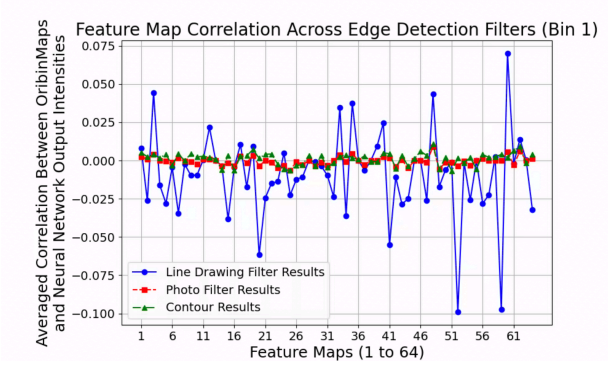


Figure 4: Bin 1 Plot

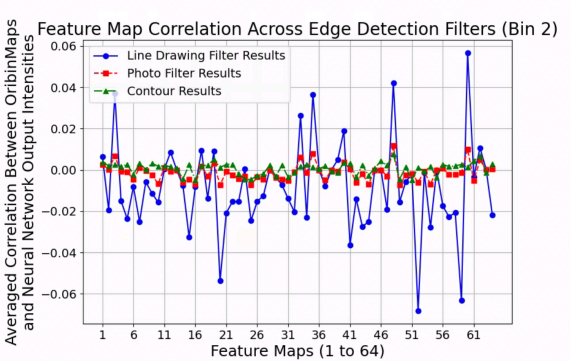


Figure 5: Bin 2 Plot

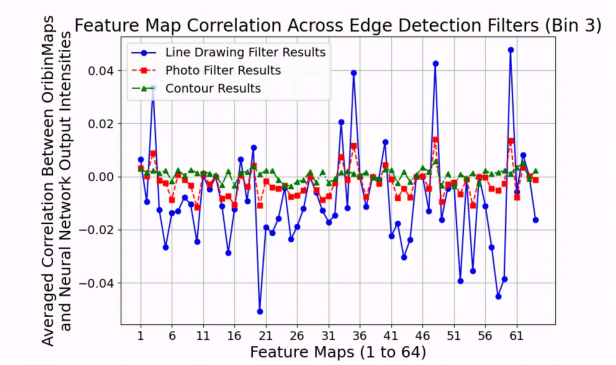


Figure 6: Bin 1 Plot

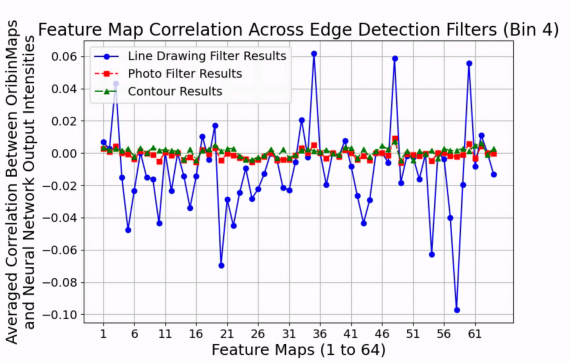


Figure 7: Bin 2 Plot

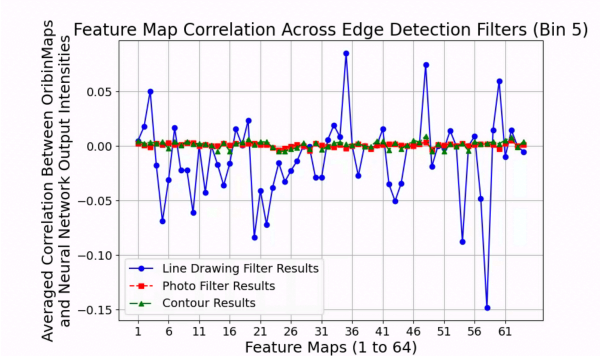


Figure 8: Bin 1 Plot

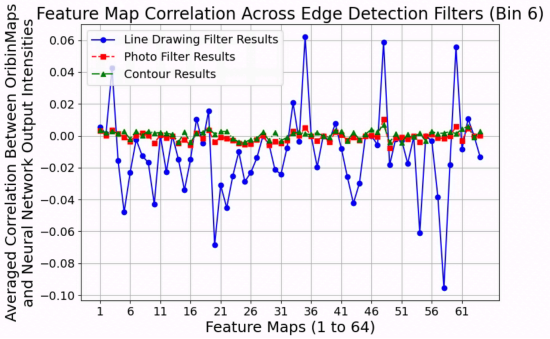


Figure 9: Bin 2 Plot

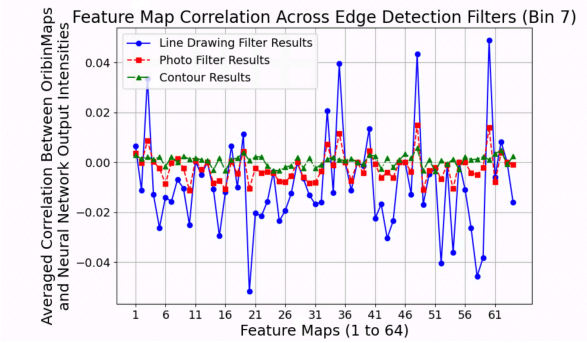


Figure 10: Bin 1 Plot

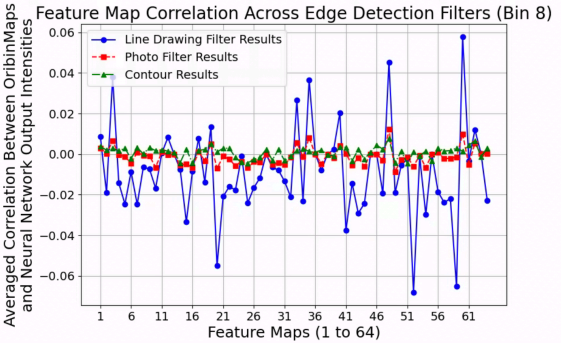


Figure 11: Bin 2 Plot

3.2 Discussion

The results obtained reveal a rich and structured organization in how individual feature maps (FMs) respond to different perceptual properties, such as edge strength and orientation. By ranking the top-3 most correlated feature maps for each experimental condition and comparing their values across different image transformations (line drawing, photo, and contour), we observe both highly specialized feature detectors and general-purpose representations emerging across the network. Some feature maps demonstrate strong, focused responses, while others appear to support multiple perceptual tasks more diffusely. These findings set the stage for a deeper dive into the intensity patterns, correlation gaps, feature map selectivity, and functional specialization throughout the following subsections.

3.2.1 Overview of Correlation Results and Plot Patterns

Two key observations emerge from our analysis of the full correlation heatmaps and plots. First, although the overall structure and pattern of correlation intensities across all eight experimental bins appear strikingly similar across the three visual transformation methods (line drawing, photo, and contour), each method produces peaks and dips at slightly different feature maps. This suggests that while the network responds to edge-like information in a consistent architectural manner, the specific enhancement or suppression of certain feature maps varies subtly depending on the input format. These small shifts may reflect how different pre-processing transformations emphasize or de-emphasize low-level visual features, such as sharpness or gradient continuity.

Second, it is noticeable that the photo and contour transformations tend to yield lower overall intensity values than the line drawing condition. This is likely due to differences in normalization procedures and global contrast scaling between the input images. Despite these discrepancies in absolute value, the relative structure of the plots — including the locations of strong positive or negative peaks — remains largely consistent. For example, even though photo and contour based correlations might range around ±0.3 while line drawing correlations reach up to 0.7, the feature maps showing the strongest responses tend to align across all formats. This alignment suggests a kind of representation invariance: the same feature maps are activated most strongly by specific perceptual properties, regardless of whether those properties appear in the form of a high-contrast line sketch, a real-world photo, or an edge-based contour rendering.

In practical terms, this means the CNN has likely developed internally consistent representations of edges and textures that are robust across different levels of abstraction and input modalities. This is an encouraging finding, as it implies that certain internal units in the network have stabilized onto abstract visual principles like contour orientation and edge alignment, which are detectable regardless of superficial changes to the input images. Table 1 was created for the purpose of outputting the ranks of the top 3 feature map performances for each of the three edge detection methods, and provides more insight into the behavior of the feature maps across different bins and filter methods.

3.2.2 Generalist Edge Detection Feature Maps

While some feature maps in the network appear to specialize in a specific visual transformation, others demonstrate broad generalization across different edge detection methods and orientation bins. These

"generalist" maps repeatedly rank in the top three across multiple input styles, suggesting that they encode edge cues that are consistent across rendering formats.

FM48: A Broad-Spectrum Edge Detector
FM 48 stands out as the most dominant and widely reused feature map across all transformations. It appears three times in every single row, meaning it is in the top three ranks for all three filter methods and every single bin.

- Line Drawing: FM48 holds rank 2 in seven bins, and Rank 3 in one bin.
- Photo Filter: FM48 is ranked first in seven bins, and ranked second in one bin.
- Contour Method: FM48 ranks first in all eight bins.

This level of consistency across methods and orientation bins suggests that FM48 serves as a general-purpose edge detector. Its repeated high rankings indicate a sensitivity to orientation features that is robust across input types — even when edge strength, line thickness, and noise characteristics vary substantially, concluding from the fact that many distinct images that produced different levels of noise were used. In this sense, FM48 appears to detect core visual structures, generalizing well regardless of whether edges are clean and binary (contour), sketch-like (line drawing), or soft and photorealistic (photo filter).

FM60: A Strong Generalist with Method-Specific Sensitivity
FM60 is another feature map that frequently reappears across bins and methods, though with slightly less uniformity than FM48.

- Line Drawing: FM60 is ranked first for 5 bins and ranked third for 3 bins, meaning it appears across all bins.
- Photo Filter: It appears in seven bins, ranking second for 6 bins and third for one bin.
- Contour: FM60 does not appear in the top 3 ranks for contour method, but holds a positive average correlation across all eight bins.

Although FM60 is not as dominant as FM48, it still demonstrates strong generalization — especially between line drawings and photo filters. However, its weaker presence in the contour method suggests that its receptive field or filter configuration may be less attuned to hard binary edges and more responsive to textured or gradient-based edge representations.

3.2.3. Orientation-Biased Feature Maps

Certain feature maps also show signs of orientation-specific tuning, responding more strongly to particular bins than others. A key example is FM35, which appears to be especially sensitive to near-horizontal edge orientations under the line drawing filter.

- Line Drawing: FM35 is present in Bins 4, 5, 6, and 7, with Rank 1 placement in Bins 4–6. These bins correspond to horizontal and near-horizontal edge orientations.
    - Bin 4: Rank 1 (0.0618)
    - Bin 5: Rank 1 (0.0851 — highest correlation in the entire table)
    - Bin 6: Rank 1 (0.0621)
    - Bin 7: Rank 3 (0.0359) — small difference from rank 1, FM60 with correlation (0.048)
- Other Methods: FM35 appears sporadically in photo filter bins, and only once (Bin 8) in contour. It is completely absent from the top three in Contour for bins 1–7.

This suggests that FM35 does not function as a generalist. Instead, it shows tuned selectivity for horizontal structures within the sketch-like, sparse lines of the line drawing method. Its absence from other bins and methods supports the idea that certain feature maps develop directional tuning, similar to how biological visual systems may develop neurons selective for particular edge orientations.

Similarly, FM3 only appears in the top three ranks for the line drawing filter method for bins 1, 2, and 8. These are the bins containing the near-vertical edges. Although FM3 is only ranked second or third for these bins, with their correlation values significantly lower than the feature maps ranking first (ie., FM60 for Bin 1 has correlation 0.0701 and FM3 has correlation 0.0444), the appearance of FM3 across these suggest they are weak horizontal edge detectors.

Together, these results imply that certain feature maps in CNNs may act as orientation-specific edge detectors, showing reliable activation patterns for edges aligned in particular directions. Rather than serving as general-purpose filters, maps like FM35 and FM3 appear to be selectively engaged by horizontal and vertical edge structures, respectively — especially within the sparse and stylized conditions of the line drawing method. This behavior suggests that, even in the absence of explicit supervision for edge orientation, DNNs can spontaneously develop filters with directional tuning, functionally similar to orientation-selective neurons observed in early stages of biological vision.

3.2.4 Method-Specific Detectors

A particularly striking pattern emerges when analyzing the Contour-based edge detection results across the eight orientation bins. The same three feature maps — FM48, FM62, and FM19 — consistently appear in the top three ranks across nearly all bins: FM48 appears in the Rank 1 position for all 8 bins, showing a high and stable correlation with contour-based edge information. FM62 is ranked second in every bin. FM19 holds Rank 3 across all bins without exception. This level of consistency is remarkable and suggests that these three feature maps are highly specialized for detecting contour orientation, and moreover, they form a dedicated submodule within the CNN for representing edge orientation features in contour renderings.

The correlation values, though relatively modest in absolute terms (ranging from ~0.006 to ~0.009), show a tight and regular structure across bins:

- FM48 (correlation range: 0.0074 – 0.0092)
- FM62 (correlation range: 0.0048 – 0.0092)
- FM19 (correlation range: 0.0039 – 0.0072)

This consistency is unique to the contour method. Neither FM62 nor FM19 dominates to the same extent in the line drawing or photo filter conditions. While they occasionally appear in those transformations, they do not exhibit the same ranking stability or specialization, with FM19 not appearing anywhere else and FM62 only ranked in the top three for Bin 1 and Bin 5 for photo filter method. Furthermore, looking at Figures 4–11, FM62 shows consistently low, though still positive, correlations across all eight bins for both the line drawing method. In fact, FM62's correlation is lower than FM19’s in some bins for these methods. This suggests a reversal in performance ranking between FM19 and FM62 for the photo and line drawing conditions, and indicates that their ability to detect edges across orientations is not as prominent outside of the contour condition.

The emergence of this cluster may point to method-specific tuning, where FM48, FM62, and FM19 are particularly well-matched to the clean, high-contrast structure of the contour renderings. Unlike photo and line drawing methods, which introduce texture, shading, or stylization, the contour filter emphasizes sharp binary transitions, potentially aligning more closely with the representational preferences of these feature maps. Moreover, the fact that the order of ranks remains constant across bins hints at a graded tuning mechanism, where FM48 may respond to the most salient contour edges, with FM62 and FM19 capturing progressively subtler features.

This kind of structured, tiered representation strongly supports the idea that deep CNNs can develop orientation-specific processing units tuned to particular low-level representations, especially under consistent visual styles like those found in contour renderings — even without explicit training signals for orientation.

3.2.5 Drawbacks on Ranks

Another important insight from Table 1 is that rank alone can be a little misleading, unless paired with the actual correlation values. In some cases, the Rank 1 feature map is significantly more predictive than Rank 2, suggesting a high degree of selectivity. In others, the difference is negligible, indicating redundancy or shared responsibility across multiple units. A compelling example is FM60 in Line Drawing Bin 1, which shows a correlation of 0.0701, far outpacing the second-best feature map (0.0444) and third-best feature map (0.0433). This wide margin suggests that FM60 is not just relevant, but dominant — likely encoding a highly specific and reliable representation for this task. These examples highlight that interpretation must go beyond ordinal ranking: the magnitude of difference between ranks can reveal whether the network relies heavily on a single specialized feature map or distributes its representation across several.

3.2.6 Conclusion

The results provide strong evidence in support of the research question: different filter methods *do* reveal orientation-specific tuning in certain feature maps within the first layer of VGG16. While the absolute strength of activation varies depending on the filter type, the identity and selectivity of the top-performing feature maps remain largely consistent. This indicates that the network has developed stable representations for edge orientation, which can be exposed more clearly through methodical input transformations. Some feature maps, like FM48 and FM60, act as general-purpose edge detectors, maintaining high rankings across all or most conditions. Others, such as FM35 and FM3, exhibit clear orientation-specific tuning, responding preferentially to horizontal or vertical edges, respectively — especially within sparse or stylized renderings like line drawings. The emergence of filter-method-specific clusters, such as FM62 and FM19 in the contour condition, further highlights how different input styles can surface specialized components of the network.

Taken together, these findings echo core principles from biological vision, where early visual neurons exhibit orientation selectivity shaped by natural image statistics. Similarly, the first-layer feature maps in VGG16 appear to self-organize into detectors tuned to fundamental edge properties, guided not by supervision but by structural biases in both the data and architecture. This convergence suggests that CNNs may not just approximate biological vision in function, but may also arrive at similar internal representations through analogous constraints.

3.3 Extension

Future explorations could extend this analysis beyond the first layer of VGG16 to deeper layers or even entirely different neural network architectures. By examining how mid-level vision—such as curvature, symmetry, or more complex edge interactions—is represented across state-of-the-art (SOTA) models, we may begin to uncover how different architectures succeed or fail in capturing specific visual properties. If certain models demonstrate consistent interpretability or alignment with human perceptual principles, this could provide valuable insights into why they perform better in certain tasks, particularly those requiring nuanced visual understanding. These findings could guide more targeted fine-tuning efforts and inspire the design of new architectures that are more aligned with the human visual system.

Moreover, this type of correlation-based analysis could be applied at scale across networks trained on different datasets or under varied supervision (e.g., self-supervised vs. supervised), to identify whether orientation-selective behavior generalizes across training regimes. Another promising extension could involve treating the correlation heatmaps as structured signals and inputting them into a secondary network trained to classify filter methods or detect orientation bins, offering a quantitative way to verify the distinctiveness of the network's edge-related representations. Ultimately, a better understanding of how different networks process the world around us at multiple levels of abstraction — from sharp binary contours to photorealistic gradients — could bring us closer to models that not only perform well but also see more like we do.

4 Bibliography

Bau, D., Zhou, B., Das, A., & Olah, C. (2017). *Network Dissection: Quantifying Interpretability of Deep Visual Representations*. arXiv preprint arXiv:1704.05710. https://arxiv.org/abs/1704.05710

Biscione, P., & Bowers, A. (2023). *Investigating the Limits of Deep Neural Networks in Modeling Gestalt Principles*. *Journal of Neural Networks*, 49(3), 102-115. https://doi.org/10.1016/j.jneural.2023.01.004

BwLab, University of Toronto. (2023). *MLV Toolbox (Mid-level Vision Toolbox)* [Computer software]. GitHub repository. https://github.com/bwlabToronto/MLV_toolbox

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

Kanizsa, G. (1979). *Grammatica del vedere*. *Il Mulino*.

Khaligh-Razavi, S. M., & Kriegeskorte, N. (2014). *Deep Supervised, but Not Unsupervised, Models May Explain IT Cortical Representation*. *PLOS Computational Biology*, 10(11), e1003915. https://doi.org/10.1371/journal.pcbi.1003915

Nakayama, K., Shimojo, S., & Hochberg, J. (1995). *Humans perceive illusory contours as surfaces*. *Science*, 257(5070), 1041-1043. https://doi.org/10.1126/science.7574489

Wallach, H. (1935). *The perception of motion and the effect of motion on form perception*. *Psychological Review*, 42(6), 528-543. https://doi.org/10.1037/h0061313

Wagemans, J., van Lier, R., & van der Helm, P. (2012). *A quantitative analysis of the structure of visual motion and depth perception*. *Vision Research*, 63, 1-16. https://doi.org/10.1016/j.visres.2012.06.001

Harvard Vision Lab. (n.d.). *The Human Visual System*. Retrieved from https://vision.harvard.edu