
A Game-Theoretic Taxonomy of Visual Concepts in DNNs

Xu Cheng
Shanghai Jiao Tong University
xcheng8@sjtu.edu.cn

Chuntung Chu
University of Toronto
chuntung.chu@mail.utoronto.ca

Yi Zheng
China University of Mining & Technology
zqt1900202051g@student.cumtb.edu.cn

Jie Ren
Shanghai Jiao Tong University
ariesrj@sjtu.edu.cn

Quanshi Zhang*
Shanghai Jiao Tong University
zqs1022@sjtu.edu.cn

Abstract

In this paper, we rethink how a DNN encodes visual concepts of different complexities from a new perspective, *i.e.* the game-theoretic multi-order interactions between pixels in an image. Beyond the categorical taxonomy of objects and the cognitive taxonomy of textures and shapes, we provide a new taxonomy of visual concepts, which helps us interpret the encoding of shapes and textures, in terms of concept complexities. In this way, based on multi-order interactions, we find three distinctive signal-processing behaviors of DNNs encoding textures. Besides, we also discover the flexibility for a DNN to encode shapes is lower than the flexibility of encoding textures. Furthermore, we analyze how DNNs encode outlier samples, and explore the impacts of network architectures on interactions. Additionally, we clarify the crucial role of the multi-order interactions in real-world applications. *The code will be released when the paper is accepted.*

1 Introduction

Although deep neural networks (DNNs) have achieved significant success in various applications, they are still considered as black boxes. Previous post-hoc explanations of visual concepts encoded by DNNs mainly visualize network features [42, 41, 29, 8] or extract correlations between input pixels and network features [2, 23, 11, 19]. Recently, Li et al. [27] proposed a method to measure whether a model mainly encoded textures or shapes for classification. These efforts focused on the distribution of visual concepts encoded over different filters/layers in an experimental manner, by observing correlations between features and the annotated concepts.

Beyond the categorical taxonomy of objects and the cognitive taxonomy of textures and shapes, in this study, we provide a different but more essential perspective to interpret signal-processing behaviors of DNNs encoding visual concepts. Specifically, we classify visual concepts into simple (local) concepts, middle-complex concepts, and complex (global) concepts, and further analyze distinctive properties of encoding each type of concepts. Thus, the new taxonomy of concepts helps us rethink and interpret the encoding of shapes and textures in a more fine-grained manner.

*Correspondence.

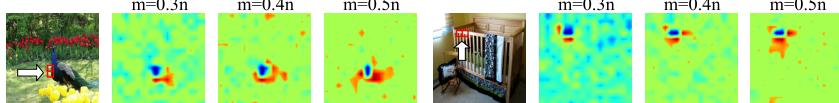


Figure 1: Contexts of the m -order interaction $I^{(m)}(i, j)$. Pixels (i, j) were indicated by white arrows. Pixels which significantly collaborated with (i, j) were regarded as a context, as shown in heatmaps. We simply extended¹ the method in [43] to visualize salient concepts of the multi-order interaction.

To this end, we use multi-order interactions [44] between input pixels in the game theory to represent concepts of different complexities. The interaction between two input pixels measures the marginal attribution from the collaboration between these two pixels, in comparison with the case they work individually. Then, the multi-order interaction represents the additional attribution brought by collaborations among these two pixels and other involved contextual pixels, where the order refers to the number of contextual pixels in the collaboration. As Fig. 1 shows, low-order interactions usually represent relatively simple and local concepts. In comparison, high-order interactions mainly indicate relatively global and complex concepts.

Based on multi-order interactions, we discover that the DNN usually encodes textures in three different ways, *i.e.* local textures, middle-complex textures, and global textures, respectively. 1) Textures encoded in low-order interactions mainly represent the common distribution of local colors, which can be generalized to different patches of the same texture category. 2) The DNN may also encode textures using middle-order interactions, and the encoding of such textures is closer to the memory of specific middle-complex instances. As Fig. 2 shows, in the fine-grained texture classification, low-order interactions usually encode common, and widely-shared local textures. In comparison, middle-order interactions mainly model the subtle difference between similar textures. 3) Textures encoded in high-order interactions mainly represent global textures, which frequently appear in training samples (*e.g.* the sky/ocean/grassland).

Furthermore, the low-order and middle-order interactions reflect the difference in flexibility between encoding textures and encoding shapes. The DNN may flexibly encode a specific texture as either plenty of repetitive local and simple textures corresponding to low-order interactions, or a few middle-complex textures of middle-order interactions. For example, the texture of the marble in Fig. 3 is made up of repetitive local textures. Hence, this texture can be modeled as either the ensemble of massive local textures or the ensemble of a few middle-complex textures. In comparison, the encoding of shapes is less flexible. *I.e.* each shape has a relatively robust distribution of interactions through different orders. For example, the DNN usually encodes a bicycle as the ensemble of middle-complex shapes such as the frame in Fig. 3, instead of massive local shapes. Particularly, when the DNN mainly uses high-order interactions to represent shapes, these interactions mainly reflect the memory of specific large-scale shapes or outliers.

Besides, we analyze how the kernel size and the layer width of ResNets influence multi-order interactions. We find that the ResNet with a large kernel size and a wide ResNet are likely to learn a few discriminative interactions towards a specific category. In comparison, the ResNet with a small kernel size and a narrow ResNet tend to learn plenty of widely-shared, generic, but less discriminative interactions. Furthermore, we clarify the crucial role of multi-order interactions in boosting the classification performance, and explaining and alleviating adversarial effects.

Contributions of this paper are summarized as follows. 1) We provide a new perspective to re-classify visual concepts, which helps us interpret how DNNs encode textures, shapes, and outlier samples in a fine-grained manner. 2) We also analyze the impacts of network architectures on interactions.

2 Related Work

We limit the discussion within the scope of post-hoc explanations of trained DNNs for visual tasks.

Semantic explanations for DNNs. An intuitive way to interpret DNNs is to visualize visual concepts encoded in intermediate layers of DNNs [9, 16, 42, 41, 29, 8, 30, 31, 38, 14, 19]. Moreover, some studies estimated the pixel-wise or regional attribution/importance/saliency of an input image to the network output [46, 33, 37, 24, 12, 34, 4, 28, 25, 6, 22, 18, 7, 36].

¹Technical details of visualizing contexts of multi-order interactions were shown in the supplementary material. Please see the supplementary material for more results.

The dissection of visual concepts is also a classical direction of interpreting DNNs. Bau et al. [2] manually annotated six types of visual concepts (*objects*, *parts*, *scenes*, *textures*, *materials* and *colors*), and associated units of DNN feature maps with these concepts. Furthermore, the first two concepts could be summarized as shapes, and the last four concepts could be summarized as textures [45]. Fong and Vedaldi [11] used a set of filters to represent a single semantic concept, and built up many-to-many projections between concepts and filters. TCAV [23] measured the importance of user-defined concepts to classification results.

In summary, previous semantic dissection was mainly conducted in an experimental manner, *i.e.* people associated filters of a DNN with manually annotated concepts. In contrast, we propose to explain the essence of signal processing for different concepts, instead of simply computing the distribution of concepts via experimental observations.

3 Analyzing and improving visual concepts of different complexities

In this section, we use multi-order interactions to explain the essence of signal processing for concepts of different complexities. This explanation helps us rethink how DNNs encode textures and shapes in a fine-grained manner.

Preliminary: multi-order interactions. Because input pixels of an image do not contribute to the network prediction independently, there often exist interactions between different pixels. Given a trained DNN v and an image x with n pixels, $N = \{1, 2, \dots, n\}$, the Shapley interaction index [13] has been proposed as a standard metric for the interaction between pixels. The Shapley interaction index $I(i, j)$ measures whether the absence/presence of the pixel i can change the attribution of the pixel j . Compared to the Shapley interaction index, in this paper, we introduce multi-order interactions [44], and illustrate that $I(i, j)$ can be decomposed as the sum of multi-order interactions. The multi-order interaction $I^{(m)}(i, j)$ measures the additional attribution brought by collaborations among i, j and other m contextual pixels, in a more fine-grained manner. Please see the supplementary material for more details about the meaning and utility of both the Shapley interaction index and the multi-order interaction.

$$I(i, j) = \frac{1}{n-1} \sum_{m=0}^{n-2} I^{(m)}(i, j), \quad I^{(m)}(i, j) = \mathbb{E}_{S \subseteq N \setminus \{i, j\}, |S|=m} [\Delta v(S, i, j)]. \quad (1)$$

Here, $\Delta v(S, i, j) \stackrel{\text{def}}{=} v(S \cup \{i, j\}) - v(S \cup \{i\}) - v(S \cup \{j\}) + v(S)$, and $v(S)$ represents the network output when we keep pixels in S unchanged and mask pixels in $N \setminus S$ by following [1]. If $I^{(m)}(i, j) > 0$, the presence of j will increase the attribution of i . Thus, we consider pixels i and j have positive interactions. Similarly, If $I^{(m)}(i, j) < 0$, i and j conflict with each other, thereby having negative interactions. Besides, if $I^{(m)}(i, j) \approx 0$, then i and j are independent.

Moreover, as Fig. 1 shows, the order m represents contextual complexity of the interaction. For a low order m , $I^{(m)}(i, j)$ measures the interaction between i and j *w.r.t.* simple contextual collaborations with a few pixels. Whereas, for a high order m , $I^{(m)}(i, j)$ refers to the interaction between i and j *w.r.t.* complex contextual collaborations with massive pixels.

In addition, we have proven that $I^{(m)}(i, j)$ satisfies *linearity*, *nullity*, *symmetry*, *marginal attribution*, *accumulation*, and *efficiency* properties, which reflect the trustworthiness of the m -order interaction².

3.1 Concepts of low/middle-order interactions

Proposition 1 *Low-order interactions mainly reflect simple, common, and elementary visual concepts without encoding rich contexts. Whereas, middle-order interactions mainly represent complex textural/shape information.*

In order to verify Proposition 1, we visualized interaction contexts of different orders. Fig. 1 shows that low-order interactions mainly reflected local edges, local colors, or local textures. In comparison, middle-order interactions mainly represented middle-scale textures/shapes, which were more complex than low-order interactions.

²Please see the supplementary material for more details.

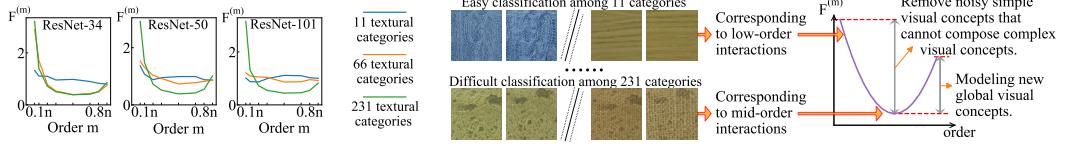


Figure 2: The m -order normalized interaction strength $F^{(m)}$ _{strength} learned for different numbers of texture categories, $0.05n \leq m \leq 0.9n$.

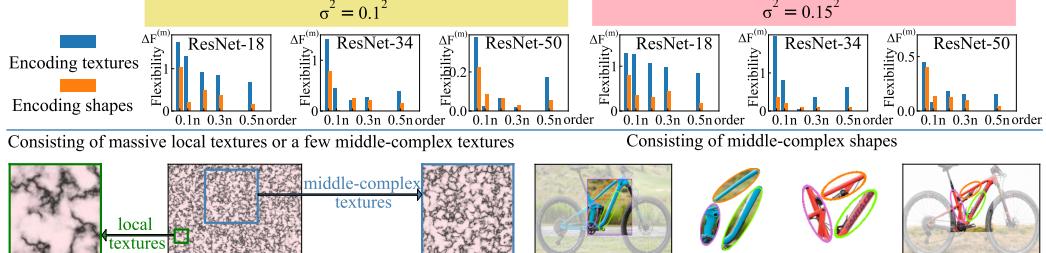


Figure 3: Flexibility of the m -order interaction strength $\Delta F^{(m)}$.

- **Understanding 1-semantics.** The modeling of local textures can be roughly divided into two types without clear boundaries. 1) Textural information encoded in low-order interactions mainly represents the simple and common color distribution, which can be generalized to patches of similar textures. 2) In comparison, textural information encoded in middle-order interactions tends to be complex and hard memorized as specific middle-complex instances by DNNs. Moreover, textures are more likely to be encoded as low-order interactions than middle-order interactions, unless low-order interactions are not discriminative enough for inference.

Verification. Here, we constructed different datasets to force DNNs to learn textures of different complexities. In this way, we could verify the correlation between texture complexity and the interaction order. As Fig. 2 (right) shows, compared to classifying a few textures, the classification of massive fine-grained textures usually forced a DNN to learn more subtle differences between similar textures, and to remove all ambiguous collaborations. Thus, only a few and less diverse middle-complex collaborations subtly distinguishing fine-grained textures could be encoded. To this end, we used an interaction strength metric to verify whether the fine-grained texture classification made the DNN encode fewer middle-order interactions.

Specifically, we constructed three texture datasets. The first was the original KTH-TIPS-2b dataset [3], which contained 11 texture categories. To construct the second dataset, we merged images from each pair of texture categories in the first dataset to generate a new texture category, thereby obtaining $11 + \binom{11}{2} = 66$ texture categories in total. Similarly, to construct the third dataset, we combined images from three texture categories in the first dataset to generate a new texture category, and achieved overall $11 + \binom{11}{2} + \binom{11}{3} = 231$ texture categories. We used these three datasets to train three versions of each ResNet-34/50/101 [21] model. The m -order interaction strength was quantified as

$$I_{\text{strength}}^{(m)} = \mathbb{E}_{x \in \Omega} \left[\mathbb{E}_{i,j} [|I^{(m)}(i,j|x)|] \right], \quad (2)$$

where $\Omega \subseteq \mathbb{R}^n$ represented the set of training samples. To reduce the computational cost, $I^{(m)}(i,j|x)$ was calculated at the grid level by dividing the image into 32×32 grids, thereby $n = 1024$.

Fig. 2 shows the m -order normalized interaction strength $F^{(m)} = I_{\text{strength}}^{(m)} / z'$, where $z' = \mathbb{E}_{m'} [I_{\text{strength}}^{(m')}]$ was used for normalization. The classification of more texture categories made the DNN encode fewer middle-order interactions. *I.e.* the stricter encoding of fine-grained textures usually led to fewer middle-order interactions.

- **Understanding 1-flexibility.** If training samples contain noises, then the strength of low/middle-order interactions learned for shapes is relatively less flexible than the strength of low/middle-order interactions learned for textures. *I.e.* the encoding of textures is more flexible than the encoding of shapes.

This can be understood as follows. Local simple repetitive textures are usually encoded as low-order interactions. The middle-order interaction for textures may be encoded in two different manners. *I.e.* the middle-order interaction can either be modeled as the collaboration of simple/elementary textures

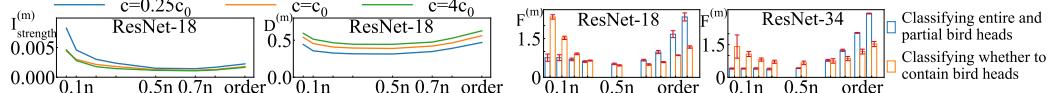


Figure 4: The m -order interaction strength $I_{\text{strength}}^{(m)}$ and disentanglement $D^{(m)}$ for ResNets with different layer widths, where $c_0 = 64$ was the layer width the standard ResNet-18.

Figure 5: $F^{(m)}$ for classifying entire and partial bird heads, and $F^{(m)}$ for classifying whether to contain bird heads, respectively. Red lines refer to standard deviations of results.

corresponding to low-order interactions, or be memorized as a larger-scale middle-complex textural patch in a pixel-wise manner. For example, the texture of the marble in Fig. 3 is composed of repetitive local textures. In this way, this texture can be encoded as either the ensemble of massive local textures or the ensemble of a few middle-complex textures. In comparison, for shapes, low-order interactions often represent local shapes. Middle-order interactions mainly represent collaborations between *middle-level* features of different local shapes, instead of *directly* memorizing the middle-complex shape in a *pixel-wise* manner due to the shape deformation. For example, the DNN encodes a bicycle as the ensemble of middle-complex compositional parts (*e.g.* the frame in Fig. 3), instead of massive local shapes. Thus, the encoding of textures is more flexible.

Verification. Here, we considered the flexibility of encoding textures and shapes when we learned DNNs under various noisy conditions. If DNNs learned under different noisy conditions had similar distributions of the multi-order interaction strength, we considered the learning of concepts was not flexible; otherwise, the learning of concepts was flexible. To this end, we aimed to verify the encoding of shapes was usually less flexible than the encoding of textures.

Thus, for encoding shapes, we trained DNNs using 30 categories with the largest numbers of samples in the Caltech 101 dataset [10] for object classification. Meanwhile, we used the aforementioned dataset with 66 texture categories, which was constructed based on the KTH-TIPS-2b dataset, to train DNNs for texture classification. Moreover, we added Gaussian noises with zero mean and variance $\sigma^2 \in \{0.1^2, 0.15^2\}$ to each training sample to train ResNet-18/34/50 models for comparisons.

In Fig. 3, we computed $\Delta F^{(m)} \stackrel{\text{def}}{=} |F^{(m),\text{noise}} - F^{(m)}|$ to reflect the flexibility of multi-order interactions. $F^{(m),\text{noise}}$ and $F^{(m)}$ denoted the interaction strength of the DNN learned with noises and the DNN learned without noise, respectively. Note that $F^{(m),\text{noise}}$ and $F^{(m)}$ had been normalized for fair comparisons as aforementioned. In this way, a large $\Delta F^{(m)}$ value indicated the m -order interaction strength was flexible under noisy conditions. Fig. 3 shows the encoding of textures was more flexible than the encoding of shapes.

- **Understanding 1-generalization.** *Compared to middle-order interactions, low-order interactions extracted from the same image have similar utilities in classification, i.e. low-order interactions push the influence towards the same category. Therefore, low-order interactions are usually learned to encode simple and common features of specific categories, just like bag-of-words models [15]. For example, a local texture of the rape blossom in Fig. 8 (IV) is a common discriminative pattern for inference. In comparison, middle-order interactions extracted from the same image tend to represent more diverse, less common, and more complex concepts of various categories.*

Verification. We proposed the disentanglement metric of the m -order interaction to check whether interactions extracted from the same image had similar utilities in classification.

$$D^{(m)} = \frac{\mathbb{E}_{x \in \Omega} \mathbb{E}_{i,j \in N, i \neq j} \left[\left| \sum_{S \subseteq N \setminus \{i,j\}, |S|=m} \Delta v(i, j, S|x) \right| \right]}{\mathbb{E}_{x \in \Omega} \mathbb{E}_{i,j \in N, i \neq j} \left[\sum_{S \subseteq N \setminus \{i,j\}, |S|=m} |\Delta v(i, j, S|x)| \right]}, \quad (3)$$

where the numerator equalled to $I_{\text{strength}}^{(m)}$. The disentanglement $D^{(m)}$ measured whether m -order interactions under different contexts had similar (either positive or negative) effects on the prediction. In this way, a high disentanglement $D^{(m)}$ reflected that interactions between pixels (i, j) under different contexts $\{S\}$ consistently had positive effects $\Delta v(S, i, j|x)$ (or consistently had negative effects) on the inference of a specific category. Therefore, we could consider a large value of $D^{(m)}$ indicated that the m -order interaction mainly had similar classification utilities towards specific categories. Let us consider a toy example. When the pair of pixels (i, j) consistently had a positive interaction towards a specific category, *i.e.* $\forall S \in \{S \subseteq N \setminus \{i, j\} | |S|=m\}, \Delta v(S, i, j|x) > 0$, then we had $D^{(m)} = 1$. This indicated the m -order interaction between i and j stably promoted the output probability of this category. In contrast, a small value of $D^{(m)}$ reflected interactions between (i, j)



Figure 6: Positive training samples with entire bird heads and negative training samples with partial bird heads, which were used to verify *understanding 2-semantics*.

referred to sophisticated concepts of various categories. *I.e.* given different contexts, the interaction sometimes had positive effects on a specific category, and sometimes had negative effects.

Fig. 4 shows low-order interactions encoded in the ResNet models with standard convolutional filters (3×3) were more disentangled than middle-order interactions, which indicated low-order interactions were more likely to stably have similar classification utilities than middle-order interactions. Experiments were introduced in Sec. 3.4, and supplementary materials also included more results computed using standard filters.

3.2 Global concepts encoded as high-order interactions

Proposition 2 *High-order interactions mainly represent the memory of specific large-scale concepts.*

- **Understanding 2-semantics.** *Unlike low/middle-order interactions mainly representing local/middle-complex concepts, high-order interactions mainly reflect specific large-scale visual concepts for inference. The large-scale (global) texture is a typical case. When a large-scale texture is encoded as a high-order interaction, it usually satisfies two requirements. 1) This large-scale textural concept frequently appears in images, e.g. the sky or the ocean. 2) When local textures are not discriminative enough, the DNN has to use high-order interactions over the entire image for inference. For example, in Fig. 8 (II), the background ocean interacts with the foreground red-breasted merganser for the classification of the red-breasted merganser. The absence of either the background or the foreground will hurt the inference.*

Verification 1. The basic idea was as follows. We constructed two datasets. One dataset forced the DNN to hard memorize the entire large-scale appearances of specific input instances for inference, while the other dataset did not. Thus, if the DNN had memorized these specific instances, it was supposed to encode more high-order interactions.

Specifically, these two datasets were constructed using images from the Tiny ImageNet [26]. To construct the first dataset, we pasted entire bird heads onto images as positive samples, and pasted partial bird heads to images as negative samples², as Fig. 6 shows. In this way, the DNN had to memorize the entire shape of the bird head for inference, which required complex collaborations with massive pixels. In comparison, to construct the second dataset, we pasted bird heads onto images as positive samples, and considered images without bird heads as negative samples². In this case, partial information of bird heads, *e.g.* beaks and eyes, was already discriminative enough to classify a bird head. It was not necessary for the DNN to memorize collaborations over the *entire* bird head for classification. We repetitively trained multiple ResNet-18/34 models for classification with different parameter initialization states. Fig. 5 shows the normalized interaction strength $F^{(m)}$ of all these DNNs. It was found that the DNN trained on the first dataset encoded more high-order interactions than the DNN trained on the second dataset. Moreover, results in Fig. 5 had small standard deviations, which proved the trustworthiness of our conclusion.

Verification 2. In the supplementary material, we conducted the second experiment. This experiment verified *understanding 2-semantics* by the phenomenon that style classification (the style could be regarded as a large-scale texture) required more large-scale collaborations than object classification for inference.

- **Understanding 2-outlier.** *Like understanding 2-semantics, the second typical case for high-order interactions is outlier samples.*

Verification 1. Because there was no standard way to define outlier samples in classification, we constructed different datasets with synthetic outliers. Specifically, we added negligible noises to some training samples, and assigned these noisy images with random labels to generate outliers. For verification, we proposed the purity metric *w.r.t.* the m -order interaction, which examined whether most m -order interactions of different contexts contributed to the inference positively, with few m -order interactions having negative effects. Based on this purity metric, we found that high-order interactions were more useful for the classification of outliers than that of normal samples.

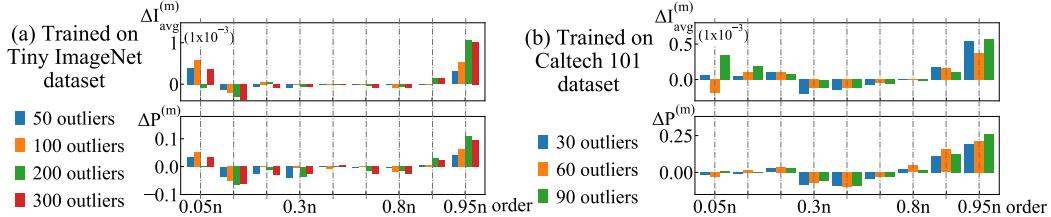


Figure 7: Differences between outlier samples and normal samples in the m -order average interaction $\Delta I_{\text{avg}}^{(m)}$ and the purity metric $\Delta P^{(m)}$, $m \in \{0.05n, 0.1n, 0.2n, 0.3n, 0.5n, 0.7n, 0.8n, 0.9n, 0.95n\}$.

We constructed datasets with synthetic outlier samples based on the Tiny ImageNet dataset and the Caltech 101 dataset², respectively. For the Tiny ImageNet, we added Gaussian noises with zero mean and variance $\sigma^2 = 0.05^2$ to 50/100/200/300 randomly chosen samples, and assigned these samples with random labels. Similarly, for the Caltech 101 dataset, we added noises to randomly selected 30/60/90 training samples ($\sigma^2 = 0.1^2$) and assigned them with random labels to construct three synthetic datasets. In this way, we trained ResNet-18 models on these datasets. Each DNN with high classification accuracy was supposed to memorize specific features of outlier samples, which required complex collaborations with massive pixels.

For verification, we used two metrics to measure whether the classification of outliers more depended on high-order interactions than normal samples. The first metric was the average interaction $I_{\text{avg}}^{(m)} = \mathbb{E}_{x \in \Omega} [\mathbb{E}_{i,j \in N} [I^{(m)}(i,j|x)]]$, which reflected the average effect of m -order interactions for inference. We had proven that a large value of $I_{\text{avg}}^{(m)}$ indicated that the m -order interaction made a significant contribution to the classification (proofs in the supplementary material). Another metric was the purity $P^{(m)}$, which measured the percentage of m -order interactions having positive effects among all m -order interactions, *i.e.* $P^{(m)} = \mathbb{E}_{x \in \Omega} \mathbb{E}_{i,j \in N} [\max(I^{(m)}(i,j|x), 0)] / I_{\text{strength}}^{(m)}$. A large value of $P^{(m)}$ indicated that more m -order interactions contributed to the classification positively, *i.e.* being more useful. In this way, we computed the difference between outliers and normal samples, $\Delta P^{(m)} \stackrel{\text{def}}{=} P^{(m), \text{outliers}} - P^{(m), \text{normal samples}}$, $\Delta I_{\text{avg}}^{(m)} \stackrel{\text{def}}{=} I_{\text{avg}}^{(m), \text{outliers}} - I_{\text{avg}}^{(m), \text{normal samples}}$.

Fig. 7 shows $\Delta P^{(m)} > 0$ and $\Delta I_{\text{avg}}^{(m)} > 0$, when $m > 0.8n$. It indicated that the classification of outliers mainly depended on high-order interactions. In the supplementary material, given an LSTM model [17] learned for sentiment classification using the SST-2 dataset [39], we also found the phenomenon that outliers relied on high-order interactions for inference.

Verification 2. In the second experiment, *understanding 2-outlier* was verified by the phenomenon that object classification of more categories (containing more hard samples, similar to outliers) encoded more high-order interactions for inference than object classification of fewer categories (see the supplementary material).

- **Understanding 2-complexity.** *High-order interactions can be regarded as both complex and simple collaborations from different perspectives. 1) According to Eqn. (1), high-order interactions are computed conditionally dependent on specific collaborations among most pixels in images. Such collaborations are complex and represent a specific input instance. 2) However, from another perspective, we can also consider high-order interactions as simple collaborations, which correspond to low-order interactions with the absence of a few pixels. I.e. high-order interactions measure how the absence of a few pixels destroys the global feature of an image. In this way, the DNN learns more “simple” high-order interactions than middle-order interactions.*

Verification. Fig. 4 and Fig. 9 verify that the strength of high-order interactions was larger than the strength of middle-order interactions. Experiments were introduced in Sec. 3.4.

3.3 Combining taxonomy of concept complexities and taxonomy of textures and shapes

Based on Propositions 1 and 2, the new taxonomy of concept complexities provides a new perspective to re-classify textures and shapes encoded in DNNs into global (large-scale) textures, local textures, global (large-scale) shapes, and local shapes. This helps people interpret the signal-processing behaviors of shapes and textures in a fine-grained manner.

To this end, we use the metric $\eta = \mathbb{E}_{i,j} [|I^{(m=0.9n)}(i,j|x)|] / z$ to describe high-order interactions, where $z = |v(N) - v(\emptyset)|$ is used for normalization. We separate images in the ImageNet validation

dataset [35] into images with large η values and images with small η values. Among all images with small η values, texture-biased images usually represent local textures, and shape-biased images often reflect local shapes. Similarly, among all images with large η values, images containing large-scale texture-biased backgrounds are usually regarded as global textures, and images including large-scale rigid objects often represent global shapes. We use the method proposed by Li et al. [27] to classify whether images are shape-biased or texture-biased.

Type 1: Hourglasses and irons in Fig. 8 (I) usually exhibit strong high-order interactions, and are considered as global shapes. The common property of these images is that they all contain similar large-scale shapes without significant deformation. For example, samples of the hourglass category in the ImageNet dataset resemble each other. In this way, the global shape without significant deformation makes the DNN encode complex interactions between massive pixels as a specific feature for inference.

Type 2: The *red-breasted merganser* and the *black grouse* with large texture-biased backgrounds in Fig. 8 (II) also exhibit strong high-order interactions, and are regarded as global textures. The common property of these images is that the classification depends on interactions between the foreground and the background, instead of exclusively relying on the foreground. For example, the absence of the *ocean* background will hurt the inference of the *red-breasted merganser*. Thus, the requirement for interactions between the foreground and background is the common cause for the DNN to encode global textures as high-order interactions.

Type 3: Mountain bikes and rocking chairs shown in Fig. 8 (III) usually exhibit weak high-order interactions, and refer to local shapes. The common property for these images is that the classification mainly depends on local shapes of the foreground object. Although the background (*e.g.* the *grassland* or the *lake* for *mountain bikes*) can be regarded as textures to some extent, these textures are not discriminative for classification, thereby not being modeled by the DNN.

Type 4: Rape blossoms and stone walls in Fig. 8 (IV) with small η values are encoded as local repetitive textures. The common property of such local textures is that they usually represent common simple repetitive textures. The local texture within larger contexts is already discriminative enough for inference. For example, simple collaborations among a local texture of *rape blossoms* are sufficient for classification. Thus, it is not necessary for the DNN to encode complex collaborations with massive pixels for inference. In other words, the efficiency of local textures is a common reason for the DNN to encode them as low-order interactions.

Li et al. [27] proposed a method to evaluate whether the DNN used textural information or shape information for inference. They used the classification confidence ξ of a specific texture-biased model to examine whether textures were encoded for inference. Similarly, the classification confidence ξ of a shape-biased model was used to check whether shapes were modeled for classification. In this way, we used ξ values to distinguish texture-biased and shape-biased images, and used the normalized high-order interaction η to distinguish global concepts and local concepts².

Fig. 8 compares the frequency histogram of η values and ξ values over images in ImageNet, which were computed based on ResNet-50 models. We localized the selected images on each histogram, which illustrated the significance of the globality/locality and the shape/texture bias of images².

3.4 Impacts of network architectures on interactions

In this section, we further explored how the kernel size and the layer width of a ResNet influenced multi-order interactions. Given a ResNet, we constructed two baseline models with different kernel sizes. In order to simplify comparative studies, for each ResNet, we changed the kernel size of all 3×3 (7×7) filters in convolutional layers to $\kappa \times \kappa$ filters, $\kappa \in \{2, 7, 11\}$. We used the Tiny ImageNet dataset to train baseline ResNet-18/50 models for object classification². Fig. 9 shows the ResNet with a larger kernel size often exhibited a smaller interaction strength $I_{\text{strength}}^{(m)}$ and a higher disentanglement $D^{(m)}$. It indicated the DNN with a large kernel size was more likely to learn a few interactions of similar classification utilities (*i.e.* learning more discriminative and more reliable interactions, see *understanding 1-generalization*), which encoded global collaborations with more pixels. This could be considered as the memory of a few complex but reliable interactions. In comparison, the DNN with a small kernel size tended to encode plenty of widely-shared, generic, but less discriminative interactions for classification.

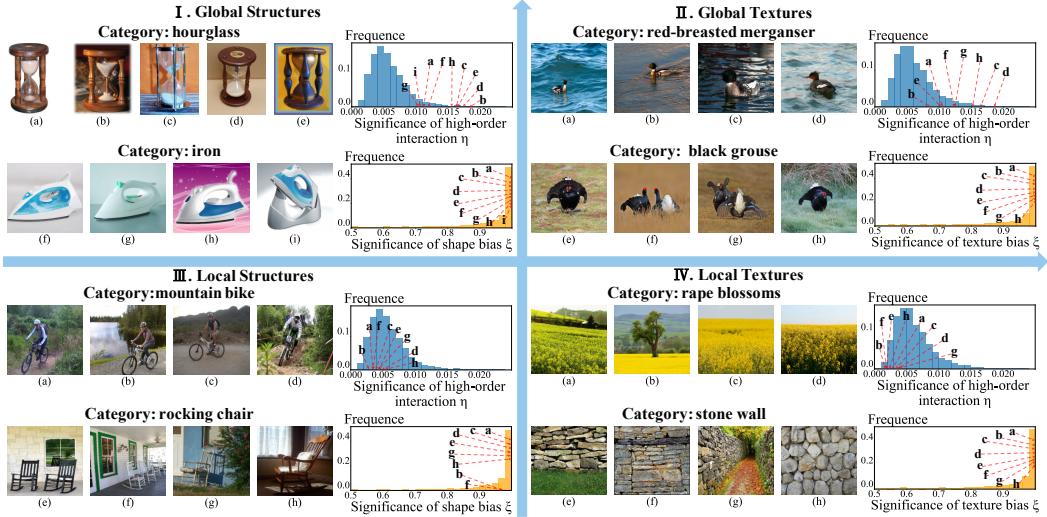


Figure 8: Typical samples of local textures, local shapes, global (large-scale) textures, and global (large-scale) shapes.

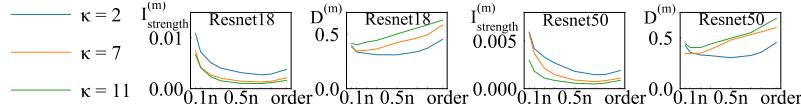


Figure 9: The m -order interaction strength $I_{\text{strength}}^{(m)}$ and the disentanglement $D^{(m)}$ for ResNets with different filter sizes $\kappa \times \kappa$.

Similarly, given a ResNet, we constructed baseline models with different layer widths (channel numbers). We used convolutional layers of widths $[c, 2c, 4c, 8c]$ for $c \in \{0.25c_0, c_0, 4c_0\}$, where $c_0 = 64$ corresponded to the standard ResNet-18. We also used the Tiny ImageNet dataset to train baseline ResNet-18 models for object classification². Fig. 4 shows a wider ResNet usually exhibited a smaller interaction strength and a higher disentanglement. It indicated a wide ResNet tended to encode a few discriminative interactions towards a specific category. Whereas, a narrow ResNet did not have enough filters, so it had to learn generic interactions shared by various categories. Nevertheless, we will conduct experiments on DNNs with more diverse architectures in the future.

Note that the phenomenon, *i.e.* the middle order interaction had a smaller disentanglement than the low-order interaction, only existed when the DNN used standard (small) filters (shown in Fig. 4). It was because small filters usually encouraged the feature sharing of low-order interactions. However, when the DNN used extremely large filters (*e.g.* 11×11 or 7×7), higher-order interactions were more disentangled (shown in Fig. 9). It was because these large filters naturally encouraged the memory of specific instances.

3.5 Clarifying practical values of multi-order interactions

In this study, we have proven that the multi-order interaction can successfully reflect the complexity of visual concepts encoded in a DNN. In this section, we further discuss recent achievements of using multi-order interactions to improve the network performance. In this way, we can associate visual concepts of different complexities with deep-learning utilities.

- **Penalizing high-order interactions to improve the classification performance.** Both *understanding 2-outlier* and findings in [44] had well shown that high-order interactions usually were responsible for the overfitting problem of a DNN. Zhang et al. [44] also proposed a method to improve the classification performance by penalizing the interaction strength.

- **Explaining and alleviating adversarial effects.** Ren et al. [32] discovered that adversarial perturbations usually affected high-order interactions. Then, the study [32] also explained the detection of adversarial samples [40] as the detection of salient high-order interactions. Besides, many defense methods [5, 20, 40] were explained by [32] as the removal of high-order interactions.

4 Conclusion

In this paper, we use multi-order interactions to rethink how a DNN encodes visual concepts of different complexities and explain the signal-processing behaviors of shapes and textures in a fine-grained manner. In this way, we classify textures into three types based on the multi-order interaction. The modeling of textures and that of shapes are distinguishable from the perspective of interactions. Besides, we use multi-order interactions to explain how a DNN encodes outliers. We also analyze impacts of network architectures on interactions modeled by the DNN, and clarify the crucial role of multi-order interactions in real-world applications. In conclusion, our study provides a new perspective to classify and analyze concepts of different complexities, beyond the categorical taxonomy of objects and the cognitive taxonomy of textures and shapes.

References

- [1] Marco Ancona, Cengiz Öztieli, and Markus Gross. Explaining deep neural networks with a polynomial time algorithm for shapley values approximation. *arXiv preprint arXiv:1903.10992*, 2019.
- [2] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 3319–3327. IEEE Computer Society, 2017. doi: 10.1109/CVPR.2017.354. URL <https://doi.org/10.1109/CVPR.2017.354>.
- [3] Barbara Caputo, Eric Hayman, and P Mallikarjuna. Class-specific material categorisation. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pages 1597–1604. IEEE, 2005.
- [4] Aditya Chattpadhyay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847. IEEE, 2018.
- [5] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.
- [6] Amit Dhurandhar, Pin-Yu Chen, Ronny Luss, Chun-Chen Tu, Paishun Ting, Karthikeyan Shanmugam, and Payel Das. Explanations based on the missing: Towards contrastive explanations with pertinent negatives. In *Advances in Neural Information Processing Systems*, pages 592–603, 2018.
- [7] Ann-Kathrin Dombrowski, Maximillian Alber, Christopher Anders, Marcel Ackermann, Klaus-Robert Müller, and Pan Kessel. Explanations can be manipulated and geometry is to blame. In *Advances in Neural Information Processing Systems*, pages 13589–13600, 2019.
- [8] Alexey Dosovitskiy and Thomas Brox. Inverting visual representations with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4829–4837, 2016.
- [9] Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009.
- [10] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004.
- [11] Ruth Fong and Andrea Vedaldi. Net2vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8730–8738, 2018.
- [12] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3429–3437, 2017.

- [13] Michel Grabisch and Marc Roubens. An axiomatic approach to the concept of interaction among players in cooperative games. *International Journal of Game Theory*, 28:547–565, 1999.
- [14] Samuel Greydanus, Anurag Koul, Jonathan Dodge, and Alan Fern. Visualizing and understanding atari agents. In *International Conference on Machine Learning*, pages 1792–1801, 2018.
- [15] Zellig S Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954.
- [16] Geoffrey E Hinton. A practical guide to training restricted boltzmann machines. In *Neural networks: Tricks of the trade*, pages 599–619. Springer, 2012.
- [17] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [18] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *Advances in Neural Information Processing Systems*, pages 9737–9748, 2019.
- [19] Md Amirul Islam, Matthew Kowal, Patrick Esser, Sen Jia, Björn Ommer, Konstantinos G. Derpanis, and Neil Bruce. Shape or texture: Understanding discriminative features in {cnn}s. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=NcFEZ0i-rLa>.
- [20] Malhar Jere, Maghav Kumar, and Farinaz Koushanfar. A singular value perspective on model robustness. *arXiv preprint arXiv:2012.03516*, 2020.
- [21] Shaoqing Ren Kaiming He, Xiangyu Zhang and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [22] Andrei Kapishnikov, Tolga Bolukbasi, Fernanda Viegas, and Michael Terry. Xrai: Better attributions through regions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.
- [23] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). *arXiv preprint arXiv:1711.11279*, 2017.
- [24] Pieter-Jan Kindermans, Kristof T Schütt, Maximilian Alber, Klaus-Robert Müller, Dumitru Erhan, Been Kim, and Sven Dähne. Learning how to explain neural networks: Patternnet and patternattribution. *arXiv preprint arXiv:1705.05598*, 2017.
- [25] Pieter-Jan Kindermans, Kristof T. Schütt, Maximilian Alber, Klaus-Robert Müller, Dumitru Erhan, Been Kim, and Sven Dähne. Learning how to explain neural networks: Patternnet and patternattribution. In *International Conference on Learning Representations*, 2018. URL <https://openreview.net/forum?id=Hkn7CBaTW>.
- [26] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7, 2015.
- [27] Yingwei Li, Qihang Yu, Mingxing Tan, Jieru Mei, Peng Tang, Wei Shen, Alan Yuille, and cihang xie. Shape-texture debiased neural network training. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=Db4yerZTYkz>.
- [28] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in neural information processing systems*, pages 4765–4774, 2017.
- [29] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5188–5196, 2015.
- [30] Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *Advances in neural information processing systems*, pages 3387–3395, 2016.

- [31] Aaron Van Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. volume 48 of *Proceedings of Machine Learning Research*, pages 1747–1756, New York, New York, USA, 20–22 Jun 2016. PMLR.
- [32] Jie Ren, Die Zhang, Yisen Wang, Lu Chen, Zhanpeng Zhou, Xu Cheng, Xin Wang, Yiting Chen, Jie Shi, and Quanshi Zhang. Game-theoretic understanding of adversarially learned features. *arXiv preprint arXiv:2103.07364*, 2021.
- [33] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.
- [34] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In *AAAI*, volume 18, pages 1527–1535, 2018.
- [35] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.
- [36] Karl Schulz, Leon Sixt, Federico Tombari, and Tim Landgraf. Restricting the flow: Information bottlenecks for attribution. *arXiv preprint arXiv:2001.00396*, 2020.
- [37] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.
- [38] K Simonyan, A Vedaldi, and A Zisserman. Deep inside convolutional networks: visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2017.
- [39] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.
- [40] Puyudi Yang, Jianbo Chen, Cho-Jui Hsieh, Jane-Ling Wang, and Michael Jordan. Ml-loo: Detecting adversarial examples with feature attribution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6639–6647, 2020.
- [41] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.
- [42] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.
- [43] Hao Zhang, Yichen Xie, Longjie Zheng, Die Zhang, and Quanshi Zhang. Interpreting multivariate interactions in dnns. *arXiv preprint arXiv:2010.05045*, 2020.
- [44] Hao Zhang, Sen Li, YinChao Ma, Mingjie Li, Yichen Xie, and Quanshi Zhang. Interpreting and boosting dropout from a game-theoretic view. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=Jacdvfjicf7>.
- [45] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18–22, 2018*, pages 8827–8836. IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00920. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Zhang_Interpretable_Convolutional_Neural_CVPR_2018_paper.html.
- [46] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.