A Peek Into the Reasoning of Neural Networks: Interpreting with Structural Visual Concepts

Yunhao Ge[1,2], Yao Xiao[2], Zhi Xu[2], Meng Zheng[1], Srikrishna Karanam[1],
Terrence Chen[1], Laurent Itti[2], and Ziyan Wu[1]
[1]United Imaging Intelligence, Cambridge MA
[2] University of Southern California, Los Angeles CA
{first.last}@united-imaging.com,yunhaoge@usc.edu,yxiao915@usc.edu,zhix@usc.edu,itti@usc.edu

Abstract

Despite substantial progress in applying neural networks (NN) to a wide variety of areas, they still largely suffer from a lack of transparency and interpretability. While recent developments in explainable artificial intelligence attempt to bridge this gap (e.g., by visualizing the correlation between input pixels and final outputs), these approaches are limited to explaining low-level relationships, and crucially, do not provide insights on error correction. In this work, we propose a framework (VRX) to interpret classification NNs with intuitive structural visual concepts. Given a trained classification model, the proposed VRX extracts relevant class-specific visual concepts and organizes them using structural concept graphs (SCG) based on pairwise concept relationships. By means of knowledge distillation, we show VRX can take a step towards mimicking the reasoning process of NNs and provide logical, concept-level explanations for final model decisions. With extensive experiments, we empirically show VRX can meaningfully answer "why" and "why not" questions about the prediction, providing easy-to-understand insights about the reasoning process. We also show that these insights can potentially provide guidance on improving NN's performance.

1. Introduction

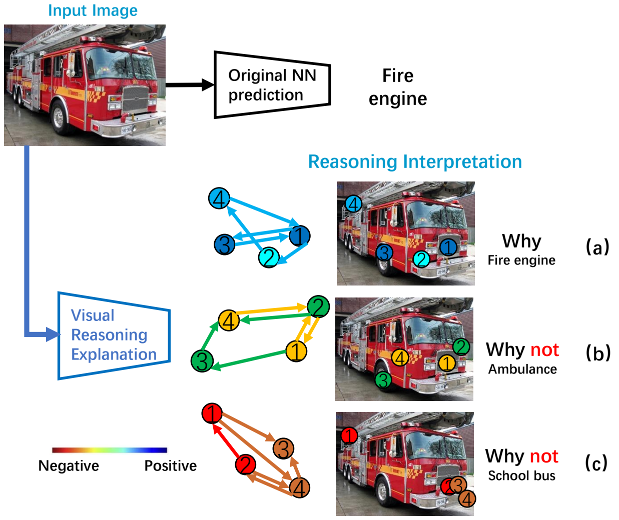


Figure 1. An example result with the proposed VRX. To explain the prediction (i.e., fire engine and not alternatives like ambulance), VRX provides both visual and structural clues. Colors of visual concepts (numbered circles) and structural relationships (arrows) represent the positive or negative contribution computed by VRX to the final decision (see color scale inset). (a): The four detected concepts (1-engine grill, 2-bumper, 3-wheel, 4-ladder) and their relationships provide a positive contribution (blue) for fire engine prediction. (b, c): Unlike (a), the top 4 concepts, and their relationships, for ambulance/school bus are not well matched and contribute negatively to the decision (green/yellow/red colors).

With the use of machine learning increasing dramatically in recent years in areas ranging from security [3] to medicine [31], it is critical that these neural network (NN) models are transparent and explainable as this relates directly to an end-user's trust in the algorithm [12, 1]. Consequently, explainable AI (xAI) has emerged as an important research topic with substantial progress in the past few years. Most recent xAI approaches attempt to explain NN decision reasoning process with visualizations depicting the correlation between input pixels (or low-level features) and the final output [40, 23, 42, 37, 30, 35, 18, 4, 32, 29], with perturbation-based [32, 29] and gradient-based [30, 4] methods receiving particular attention in the community. Despite impressive progress, we identify some key limitations of these methods that motivate our work. First, the resulting explanations are limited to low-level relationships and are insufficient to provide in-depth reasoning for model inference. Second, these methods do not have systematic processes to verify the reliability of the proposed model explanations [17, 9]. Finally, they do not offer guidance on how to correct mistakes made by the original model.

We contend that explaining the underlying decision reasoning process of the NN is critical to addressing the aforementioned issues. In addition to providing in-depth understanding and precise causality of a model's inference process, such a capability can help diagnose errors in the original model and improve performance, thereby helping take a step towards building next-generation human-in-the-loop AI systems. To take a step towards these goals, we propose the visual reasoning explanation framework (VRX) with the following key contributions:

- To understand what an NN pays attention to, given an input image, we use high-level category-specific visual concepts and their pairwise relationships to build structural concepts graphs (SCGs) that help to highlight spatial relationships between visual concepts. Furthermore, our proposed method can in-principle encode higher-order relationships between visual concepts.

- To explain an NN's reasoning process, we propose a GNN-based graph reasoning network (GRN) framework that comprises a distillation-based knowledge transfer algorithm between the original NN and the GRN. With SCGs as input, the GRN helps optimize the underlying structural relationships between concepts that are important for the original NN's final decision, providing a procedure to explain the original NN.

- Our proposed GRN is designed to answer interpretability questions such as `why` and `why not` as they relate to the original NN's inference decisions, helping provide systematic verification techniques to demonstrate the causality between our explanations and the model decision. We provide qualitative and quantitative results to show efficacy and reliability.

- As a useful by-product, in addition to visual reasoning explanations, our method can help take a step towards diagnosing reasons for any incorrect predictions and guide the model towards improved performance.

2. Related Work

In this section, we review existing literature relevant to our work interpreting convolutional neural networks, graph neural networks, and knowledge distillation to differentiate our method from others.

Interpreting neural networks. The substantial recent increase in the practical adoption of deep learning has necessitated the development of explainability and interpretability methods for neural networks (NNs), and convolutional neural networks (CNNs) in particular. One line of work focuses on pixel-level interpretation [30, 4, 42, 8, 20, 41, 38], producing attention maps to highlight the relevant image regions contributing to the final model decision. These methods can further be categorized into gradient-based and response-based methods. Response-based approaches use an additional computational unit to calculate the importance score of spatial image locations. For example, CAM [42] utilized an auxiliary fully-connected layer to produce the spatial attention map and highlight image pixels contributing to the network decision. On the other hand, gradient-based methods, e.g., Grad-CAM [30], generate class-specific attention maps based on gradients backpropagated to the last convolutional layer given the model prediction. In addition to pixel-level interpretation, several recent works proposed to extract more human-intuitive concept-level explanations for interpreting neural networks [18, 10]. Specifically, Kim *et al*. [18] proposed TCAV where directional derivatives are used to quantify the sensitivity of the network's prediction with respect to input user-defined concepts. Ghorbani *et al*. proposed an automatic concept selection algorithm [10] based on the TCAV scores to produce meaningful concept-level explanations. While our framework also produces concept explanations automatically, it goes beyond this and learns explicit inter-concept relationships, producing more insightful interpretations.

Graph Networks. Graph neural networks (GNNs) have been successfully applied to tasks ranging from node classification [19, 14, 39], edge classification [26, 13] to graph classification [11, 5]. Based on "message passing", powerful extensions such as GCNs [19], graph attention network (GAT) [36], SAGE [14] and k-GNNs [24] have been proposed. Due to their trackable information-communication properties, GNNs can also be used for reasoning tasks, such as VQA [34, 25] and scene understanding [22]. In this work, we adopt the GCN to learn semantic relationships and interactions between human-interpretable concepts, providing more thorough explanations.

Knowledge distillation. Knowledge distillation can effectively learn a small student model from a large ensembled teacher model [16], which finds broad applications in different areas, like model compression [28] and knowledge transfer [27]. In a similar spirit, in this work, we learn an easy-to-understand graph reasoning network (GRN) that produces the same classification decisions as the original NN model while also learning structural relationships between concepts to generate in-depth explanations for the original NN inference decisions.

3. Visual Reasoning Explanation Framework

Our proposed visual reasoning explanation framework (VRX) to explain the underlying decision reasoning process of a given NN is visually summarized in Fig. 2. VRX

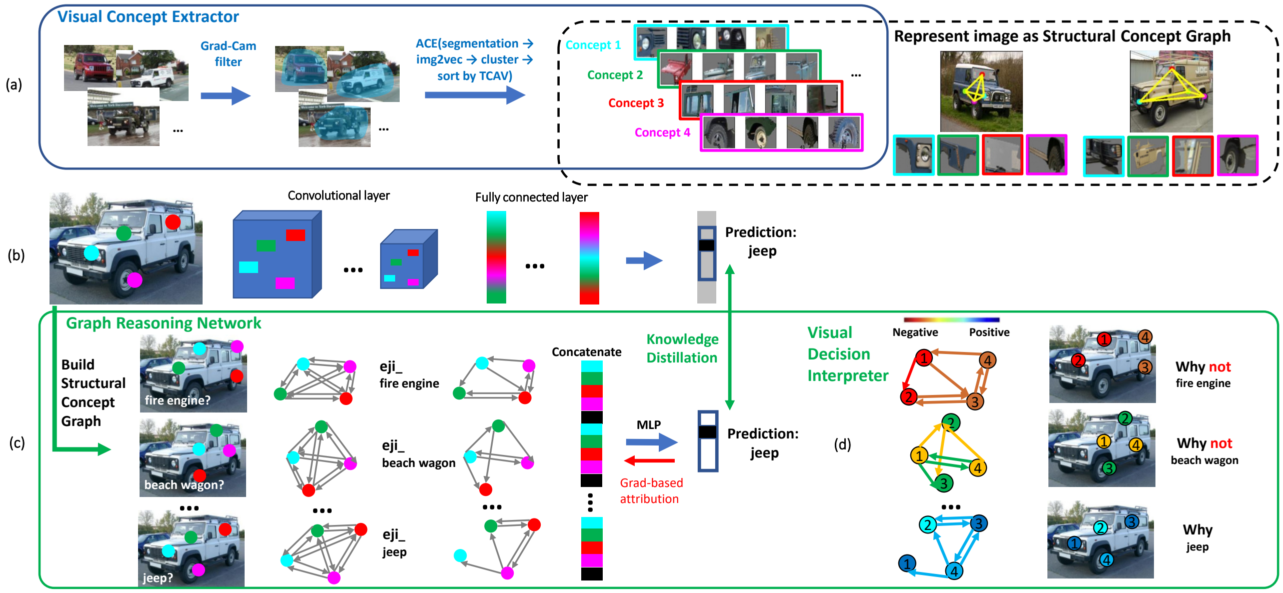


Figure 2. Pipeline for Visual Reasoning Explanation framework. (a) The Visual Concept Extractor (VCE) discovers the class-specific important visual concepts. (b) In original NN, the representation of the top N concepts is distributed throughout the network (colored discs and rectangles). (c) Using Visual Concept Graphs that are specific to each image class, our VRX learns the respective contributions from visual concepts and from their spatial relationships, through distillation, to explain the network's decision. (d) In this example, the concept graphs colored according to contributions from concepts and relations towards each class explain why the network decides that this input is a Jeep and not others.

comprises three main components: a visual concept extractor (VCE) to identify primitive category-specific visual concepts from the given neural network; a graph reasoning network (GRN) to organize category-specific visual concepts, represented as structural concept graphs (SCGs), based on their structural relationships, to mimic the decision of the original NN with knowledge transfer and distillation; and a visual decision interpreter (VDI) to visualize the reasoning process of the neural network given a certain prediction. We next explain each of these components in detail.

3.1. Visual Concept Extractor

While most existing neural network explanation techniques focus on producing low-level saliency maps, these results may be suboptimal as they may not be intuitive for human users to understand. Inspired by the concept-based explanations (ACE) technique [10], we propose to use visual concepts to represent an input image given class-specific knowledge of the trained neural network to help interpret its underlying decision-making processes.

While ACE [10] is reasonably effective in extracting class-specific visual concepts, its performance is dependent on the availability of sufficient image samples for the given class of interest. As we show in Figure 3 (left), for a class (ambulance here) with a small number of training images (50), the ACE concepts mostly fall on the background region, presenting challenges for a downstream visual explanation. To alleviate this issue, given an image I, we propose to use top-down gradient attention [30] to first constrain the relevant regions for concept proposals to the foreground segments, thereby helping rule out irrelevant background patterns. Given the class-specific attention map M, we use a threshold τ to binarize M as $\bar{M}$ (pixel values lower than τ set to 0, others set to 1), which is used to generate the masked image $\bar{I} = I \times \bar{M}$ ($\times$ is element-wise multiplication) for further processing. Specifically, following ACE, we extract the top-N visual concepts and their mean feature vectors for each class of interest using the original trained NN. Fig. 3 demonstrates the importance of the proposed gradient attention pre-filtering discussed above using top-3 visual concepts for the ambulance class (concepts with the pre-filtering focus more clearly on the foreground).

3.2. Graph Reasoning Network

3.2.1 Representing Images as SCGs

Given the aforementioned class-specific visual concepts (see Section 3.1), we represent images using structural concept graphs (SCGs), which, as input to our proposed graph reasoning network (GRN), helps learn structural relationships between concepts and produce visual explanations for the original NN. Specifically, given an image, we use multi-resolution segmentation to obtain image patches (also called

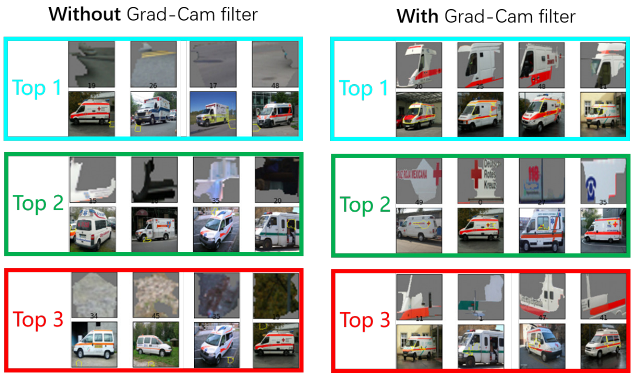


Figure 3. Concept discovery with and without Grad-Cam filter.

concept candidates), as inputs to the original NN to compute patch features, and then match these features to the mean concept feature vectors derived above (from Section 3.1). For each class of interest, we construct an SCG with concepts/patches detected from the input image, based on the Euclidean distance between patch feature and mean concept feature. Specifically, if the Euclidean distance between image patch feature and mean concept feature is larger than a threshold t, we identify this patch as a detected concept. For undetected concepts, we use dummy node feature representation (all feature values equal to a small constant ϵ), to ensure network dimension consistency. Note that we have n SCGs generated for the same input image considering all n classes of interest.

SCG is a fully connected graph (V, E) with bidirectional edges where each node $v_i \in V$ represents one relevant visual concept. Each directed edge $\text{edge}_{ji} = (v_j, v_i) \in E$ has two attributes: 1) a representation of spatial structure relationship between nodes edge_{ji}, initialized with the normalized image locations $[x_j, y_j, x_i, y_i]$ of the two visual concepts it connects and updated in each layer of GRN; 2) a measure of dependency e_{ji} (a trainable scalar) between concepts v_i, v_j (see Fig. 2 (c) and Fig.5 for an overview). Such a design helps our framework not only discover human-interpretable visual concepts contributing to network prediction but also how their underlying interactions (with e_{ji} capturing the dependencies) affect the final decision.

3.2.2 Imitate the Reasoning Process of NN

In addition to learning concept representations and capturing the structural relationship between visual concepts we also need to ensure the proposed GRN follows the same reasoning process as the original NN. Since we represent images as SCGs, this problem comes down to optimizing the GRN, with SCG inputs, so it gives the same output/prediction as the original NN with image inputs. We realize this with a distillation-based training strategy.

Specifically, given an input image I and a trained NN classifier $\mathcal{F}(\cdot)$, along with n SCG hypotheses $\mathbf{h} = \{h_1, h_2, ...h_n\}$ extracted from the input image, we seek to learn the GRN $\mathcal{G}$ for $\mathbf{h}$ such that $\mathcal{G}(\mathbf{h}) = \mathcal{F}(I)$, i.e., ensuring prediction consistency between the GRN and the original NN. The proposed $\mathcal{G}(\cdot)$ comprises two modules: 1) a GNN G is applied for all classes with different class-specific e_{ji} to learn the graph representation of SCGs; 2) an embedding network E is used to fuse multi-category SCGs for final class prediction, i.e.:

$$\mathcal{G}(\mathbf{h}) = E(G(\mathbf{h})) = \mathcal{F}(I) \tag{1}$$

Fig. 2(b-c) give an overview of the component relationship between the original NN (b) and the proposed GRN (c), showing how GRN learns the "embedding" for each hypothesis and through knowledge distillation ensures the same prediction as the original NN.

We use GraphConv [24] as G's backbone network and modify the aggregate weights. For each graph convolutional layer, we have:

$$f^i_{k+1} = W_1 f^i_k + \sum_{j \in \mathcal{N}(i)} e^c_{ji} W_2 f^j_k \tag{2}$$

where f^i_k denotes the feature of node v_i (representing a concept) in layer k, W_1 and W_2 denote the shared linear transformation parameters for center node v_i and neighbor node v_j respectively, $\mathcal{N}(i)$ denotes the neighboring node sets connected to node i, and e^c_{ji} denotes the aggregation weight from start node v_j to end node v_i for a certain class c, indicating the inter-dependency of concepts i on j. Instead of using shared edges for all classes of interest, GRN learns class-specific e^c_{ji}, i.e. different aggregation weights for different classes to capture varying structural relationships between class-specific concepts.

In order to better capture inter-concept relationships, we concatenate edge features with neighboring node features, denoted as $\mathcal{C}(e^c_{ji} W_2 f^j_k, \text{edge}^{ji}_k)$, and Equation 2 becomes:

$$f^i_{k+1} = W_1 f^i_k + \sum_{j \in \mathcal{N}(i)} W_3 \mathcal{C}(e^c_{ji} W_2 f^j_k, \text{edge}^{ji}_k) \tag{3}$$

With $\text{edge}^{ji}_{k+1} = W_4 \text{edge}^{ji}_k$, and W_3 and W_4 denoting one layer linear transformation for concatenated message feature and edge feature respectively. Since e^c_{ji} is a trainable parameter by design in our G, it helps learn concept interdependency as measured by the overall training objective (see Fig. 5(b) for a fire engine image example).

The embedding network E concatenates all the feature vectors output from G and maps it into a $n-$dimensional vector with an MLP (n is the number of classes of interest). The GRN is then trained to imitate the original NN (see Fig. 4) by minimizing:

$$\mathcal{L}_\text{d} = ||\sigma(\mathcal{G}(\mathbf{h})) - \sigma(\mathcal{F}(I))||_{l_1} \tag{4}$$

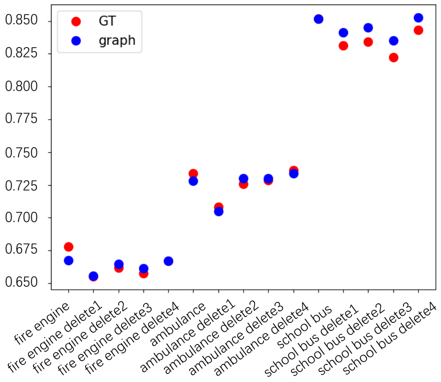
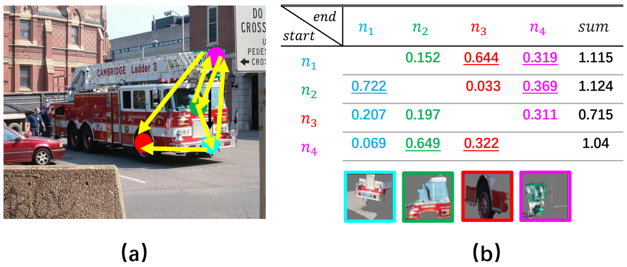


Figure 4. Decision comparison between original NN and proposed GRN.



end / start	n_1	n_2	n_3	n_4	*sum*
n_1		0.152	0.644	0.319	1.115
n_2	0.722		0.033	0.369	1.124
n_3	0.207	0.197		0.311	0.715
n_4	0.069	0.649	0.322		1.04

Figure 5. (a) Class-specific importance weights e_{ji} highlight the important concept relationships for different classes (b) e_{ji} reveals the information transformation between concepts, which shows the dependency between concepts: concept 1 and 2 contribute most information to other concepts, which makes them the 2 most discriminating concepts for a fire engine.

where $\sigma(\cdot)$ is a normalization function (see Supplementary for more implementation details). To imitation robustly, we randomly mask out one of the detected visual concepts on the input image. Fig. 4 demonstrates the prediction comparison between the learned $\mathcal{G}$ and the original NN. $\{$*class name*$\}$_detect$\{N\}$ denotes images from category *class name* with concept N masked out.

3.3. Visual Decision Interpreter

Once our GRN is trained to be a structural-concept-level representation of the original neural network, we can then interpret the original model decisions with our visual decision interpreter (VDI) module. As shown in Fig. 2(c-d), after feeding an image to both the original NN and the GRN, we obtain the final prediction y representing the probability of all class of interest, $y = E(G(\mathbf{h})) = E(\mathcal{C}_{i=1}^{m}(G^i(h_i)))$. where G^i represents the shared G equipped with class i's aggregate weight e_{ji}^{i} and $G^i(h_i)$ is the graph embedding for the ith hypothesis SCG composed of the extracted concept node and edge feature representations; $\mathcal{C}$ denotes concatenation operation. For each interested class c, we have a class prediction score y^c and compute gradients of y^c with respect to the graph embeddings from m hypothesis as:

$$\boldsymbol{\alpha}_i = \frac{\partial y^c}{\partial G^i(h_i)}, i = 1, ..., m \tag{5}$$

where $\boldsymbol{\alpha}_i$ denotes the contribution weight vector of hypothesis h_i. The contribution score s_i for each hypothesis h_i w.r.t the prediction of y^c is computed as the weighted sum of $\boldsymbol{\alpha}_i$ and $G(h_i)$:

$$s_i = \boldsymbol{\alpha}_i^T G^i(h_i), i = 1, ..., m \tag{6}$$

We then use the contribution score s_i computed from Eq. 6 to indicate the positive or negative contribution (contribution score) of each node (concept) or edge (spatial and dependency conceptual relationship) to the decision made by the neural network (positive contribution score means positive contribution and vice versa).

4. Experiments and results

We conduct four different experiments to demonstrate the effectiveness of our proposed VRX in interpreting the underlying reasoning logic of neural network's decision, guiding network diagnosis and improving the performance of the original neural network. In our experiments, we use Xception [6] and GoogLeNet models [33] pre-trained on the ILSVRC2012 dataset (ImageNet) [7] as the target neural networks.

4.1. Visual Reasoning Explanation Experiment

Fig. 6 (a-b) shows two examples (one correct and one incorrect prediction) of how our VRX can help to explain the decision behind neural networks by performing experiments on GoogLeNet and Xception, respectively.

Given a pre-trained GoogLeNet on ImageNet, we develop a VRX as introduced in Sec. 3 to explain the reasoning logic. As shown in Fig. 6 (a), for the input school bus image, both GoogLeNet and our VRX correctly predict the input as a school bus, with VRX outputs nearly identical prediction vector as original GoogLeNet which aligns with our expectation that our VRX ideally should imitate the behavior of original NN. We then use our proposed VRX to compute the contribution score for each concept node and edge to analyze how the detected human-interpretable concepts along with their structural relationships contributing to the network's decision. In this case, we ask 'why school bus?' (why the original NN predict this image as a school bus?): from a visual/conceptual perspective, all detected top 4 important concepts have high positive contribution (blue) to the prediction probability of school bus (Row 3 of Fig. 6 (a)), indicating the network is able to discover meaningful visual regions contributive to the correct

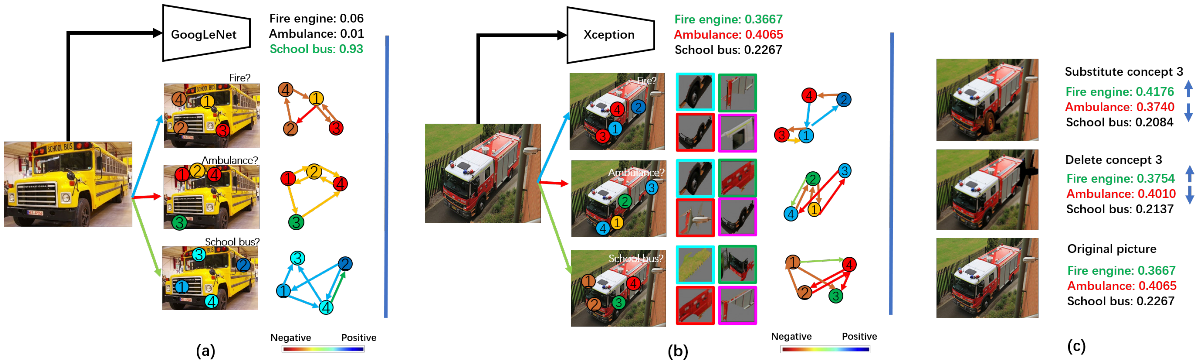


Figure 6. Visual Reasoning Explanation and logic consistency experiment example.

prediction; from a structural perspective, the spatial location and relationship between concepts represented by edge arrows also contribute positively (light or dark blue), meaning the network identifies correct spatial correlations between detected visual concepts. Similarly, to answer 'why not fire engine?' and 'why not ambulance?', VRX identifies nearly all detected concepts negatively contribute to the corresponding prediction class, and all structure relationships between concepts have negative contributions to the class prediction as well. Based on the explanation above, VRX can give a systematically in-depth and easy-to-understand interpretation of the decision-making logic of GoogLeNet, from the visual and structural perspectives respectively.

The second example is shown in Fig. 6 (b) for Xception network. Given an image of a fire engine, both the original Xception and our VRX wrongly predict ambulance as output. To understand why original Xception makes the incorrect prediction, our VRX is able to provide both visual and structural clues as well. From Fig. 6 (b) Row 1, we can see that the detected visual concepts 3 (wheels of the vehicle) and 4 have negative contribution to the prediction of fire engine class, indicating that the wheel region of the input image is not consistent with the model's knowledge of fire engine (with negative contribution). To answer "why ambulance", concept 3 and 4 have positive contribution to ambulance prediction, which explains why the original Xception network incorrectly predicts the input image as an ambulance.

4.2. Logic Consistency between VRX and NN

To verify that the explanation of VRX is logically consistent with the reasoning of Xception, we present two experiments as follows. First, as shown in Fig. 6 (c), for the wrong prediction example same as Fig. 6 (b), we substitute the flawed fire engine concept 3, which has a negative contribution (low contribution score), with a good concept 3 (high contribution score) from another fire engine image and form a new modified image. Then, we use Xception to re-predict the class of the modified image, it corrects the error and predicts the input as a fire engine correctly. To show a causal relationship between VRX's explanation and the reasoning logic of Xception, we perform two additional contrastive experiments: a) Random substitute: if we substitute concept 3 with random patches, Xception does not achieve a correct prediction; b) Substitute good: if we substitute concepts 1 or 2 with other equivalently good patches from other images of fire engines, Xception also does not produce a correct decision. Thus, we conclude that VRX has correctly diagnosed the cause of Xception's error (here, a bad concept 3). Below, we show how this can be used to further guide improved training of original NN without manually modifying the image.

For the wrongly predicted class, ambulance, if we delete a concept patch with a high contribution to ambulance probability, the prediction of Xception shows a decreased probability prediction of ambulance class and a higher probability prediction of fire engine. In total, we applied this experiment to 119 images that were initially wrongly predicted by Xception (Table. 1). The results show that with the guidance of VRX (confusing/bad concept detected), most wrong prediction cases can be corrected through learning-based modifications of the images.

	Cause of error			
Error type	total	concept	structure	both
Before correction	119	5	6	108
Substitute with Random patches	117	5	6	106
Change good concepts	115	5	6	104
VRX guided correction	**5**	**1**	**2**	**2**

Table 1. VRX model helps correction. Out of 119 images initially misclassified by Xception, only 5 remain misclassified after VRX-guided image editing. Over 30% of the samples have missing concepts and over 95% of them have been correctly explained. In contrast, 117 and 115 images remain misclassified after substituting bad concepts with random image patches, or substituting good concepts with other good concepts from other images from the same class.

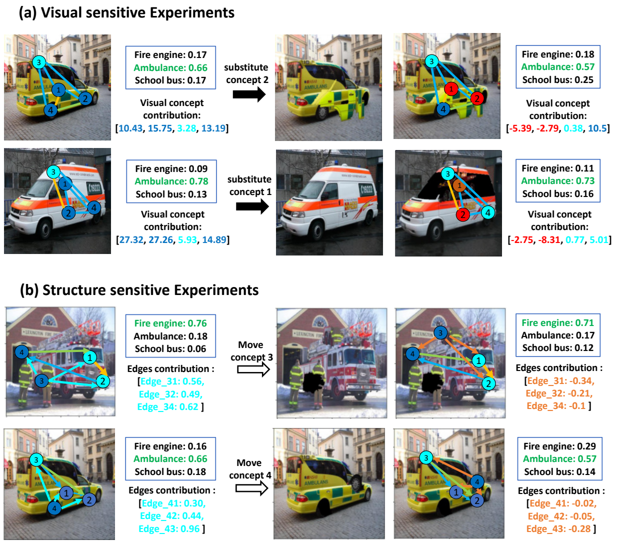


Figure 7. Interpretation from VRX is sensitive to visual and structure aspects. (a) visual sensitive (b) structure sensitive.

4.3. Interpretation Sensitive of Visual and Structure

We have demonstrated that VRX can help explain why and why not the model makes the decision, and shows a causal relationship between VRX's explanation and the original NN's decision. In this section, we focus on the sensitivity analysis of VRX's explanation from visual and structural aspects, respectively. We design two experiments accordingly: first, when we substitute a relatively good concept (with high positive contribution scores to corresponding class prediction) patch with a relatively bad concept (with lower positive or even negative contribution score) patch in an image, we want to see if VRX can capture the difference and precisely locate the correct modification, which shows the sensitivity of VRX to visual explanation. Second, when we move one concept's location from a reasonable place to an abnormal location, we want to make sure if VRX can precisely capture the structural abnormality and produce a corresponding explanation that correctly matches our modification.

Fig. 7(a) demonstrates two visual sensitivity experiment examples. In the top row, given an ambulance image with a correct prediction from a trained Xception (Fig. 7(a) left), VRX explains that all detected concepts and relative structure relationship have positive contributions to the prediction of ambulance class. We then substitute the original good concept 2 with relatively bad concept 2 from another ambulance image and form a modified ambulance image (Fig. 7(a) right), to check the sensitivity of our VRX with respect to visual perturbation. From Fig. 7(a), we can see that after the substitution, the class prediction score from both VRX and original Xception decrease as expected. While VRX gives a clear explanation for this performance decrease due to: less contributive concept 1 and 2 (negative contribution to the ambulance prediction), and invariant structure contributions, which correctly matches our modification in the original image. This proves the sensitivity of our VRX to visual perturbations. The second row of Fig. 7(a) shows an additional example of visual sensitivity test.

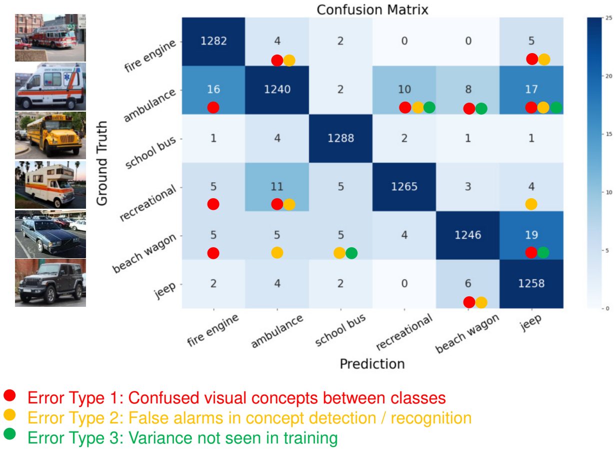


Figure 8. Model diagnosis and improving performance

Fig. 7(b) illustrates two structure sensitivity experiments. Given a fire engine image with a correct prediction from trained Xception, VRX shows that concept 3 and the structural relationships of concept 3 to all adjacent concepts are positively contributive for class prediction. We then move concept 3 from the original location to an abnormal location (we move the wheels from the bottom to the sky) and form a modified fire engine image (Fig. 7(b) right) to test the structural sensitivity of our VRX. Similarly, VRX produces consistent explanation with respect to structure perturbation as well, where the spatial relationship importance score between concept 3 to all adjacent concepts decrease after the substitution, which demonstrates the good sensitivity of our VRX to structural information. A second example in Fig. 7(b) shows similar results.

4.4. Model Diagnosis with VRX

With the explainability of VRX, reasoning results generated by VRX can be further utilized to guide improving the performance and generalizability of the original NN. Fig. 8 shows a 6-class confusion matrix with Xception. With VRX, the type of error Xception makes can be categorized as the following:

(1) Confused visual concepts between classes. The top k concepts of different classes may share certain overlaps. For instance, most vehicles have concepts related to 'wheels'. Hence judging only by this concept, the neural network may confuse one type of vehicle with another. There are existing

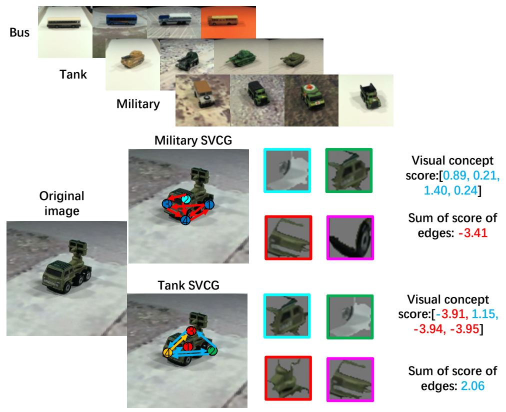

Bus
Tank
Military
Military SVCG
Original image
Visual concept score:[0.89, 0.21, 1.40, 0.24]
Sum of score of edges: -3.41
Tank SVCG
Visual concept score:[-3.91, 1.15, -3.94, -3.95]
Sum of score of edges: 2.06

Figure 9. Diagnosis and improvement experiment on iLab-20M.

approaches [21] which can guide the network in growing its attentive region and alleviating the impact from biases in training data.

(2) False alarms in concept detection/recognition. To VRX this usually means one or more patches are incorrectly labeled, which means either the neural network's feature extraction can be improved, or the most important visual concepts for specific classes are not discriminative enough.

(3) Variance not seen in training. For instance, the distribution of viewpoints of a class of interest is biased in the training set of the NN. When the same object with an unseen viewpoint is presented to the NN, it may fail to recognize it. In these cases, in VRX's decision reasoning, it may appear that most of the detected concepts are very close matches. However, the edge features seem off, suggesting the structural or spatial relationships between concepts are the cause for the NN to make incorrect predictions. Augmenting the training images with more diversity in viewpoints may solve the problem, as the further experiment shown below with the iLab-20M [2] dataset.

To further demonstrate the capability of NN diagnosis, we design an experiment on iLab-20M. iLab-20M is an attributed dataset with images of toy vehicles on a turntable captured with 11 cameras from different viewpoints. We sampled a subset from iLab-20M with similar identity and pose: we focus on three classes of vehicles: bus, military, and tank. In the training set, each class has 1000 images. We manually introduce biases with the pose of each class: all buses are with pose 1, all military are with pose 2 and all tanks are with pose 3 (Fig. 9). We designed an unbiased test set where each kind of vehicle has all the 3 poses.

We train a Resnet-18 [15] to classify the 3 types of vehicles with the training set and test the accuracy on the test set (Table. 2). To explain the reasoning logic of the trained network, we trained a GRN with VRX and explained the logic of common mistakes made by the Resnet-18 (Details in supplementary). For most incorrectly classified samples in the test set, given the input image (in Fig. 9, the military is wrongly predicted as tank), VRX's interpretation shows that most of the detected visual concepts had a positive contribution to the correct class while the structure relationship between concepts contributed mostly negatively, which leads to the incorrect prediction. To verify the "diagnosis", we designed a follow-up experiment, focusing on improving performance for the military class. Setting 1: we add images of additional poses (150 for each of the three poses) for the military in the training set and test the performance on the test set; setting 2: we add the same amount of images (450) as setting 1 but with images of the same pose as in the original training set. Table 2 shows that the accuracy with the augmented training set using setting 1 obtains much higher performance compared to the initial experiment and the follow-up experiment with setting 2 which does not bring any improvement. This suggests that VRX can help to diagnose the root cause of mistakes a neural network made, and potentially provide useful suggestions to improve the original NN's performance.

	original	setting 1	setting 2
Average accuracy	50	60	50

Table 2. Testing set accuracy comparison for VRX boost original model performance. All numbers are in %.

5. Conclusion

We considered the challenging problem of interpreting the decision process of a neural network for better transparency and explainability. We proposed a visual reasoning explanation framework (VRX) which can extract category-specific primitive visual concepts from a given neural network, and imitate the neural network's decision-making process. Our experiments showed that the VRX can visualize the reasoning process behind neural network's predictions at the concept level, which is intuitive for human users. Furthermore, with the interpretation from VRX, we demonstrated that it can provide diagnostic analysis and insights on the neural network, potentially providing guidance on its performance improvement. We believe that this is a small but important step forward towards better transparency and interpretability for deep neural networks.

Acknowledgments

This paper is based on the work done during Yunhao Ge's internship with United Imaging Intelligence. The research was partially supported by C-BRIC (one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA), the Army Research Office (W911NF2020053), and the Intel and CISCO Corporations. The authors affirm that the views expressed herein are solely their own, and do not represent the views of the United States government or any agency thereof.

References

[1] Amina Adadi and Mohammed Berrada. Peeking inside the black-box: A survey on explainable artificial intelligence (xai). *IEEE Access*, 6:52138–52160, 2018. 1

[2] Ali Borji, Saeed Izadi, and Laurent Itti. ilab-20m: A large-scale controlled object dataset to investigate deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2221–2230, 2016. 8

[3] Anna L Buczak and Erhan Guven. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications surveys & tutorials*, 18(2):1153–1176, 2015. 1

[4] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N. Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *WACV*, 2018. 1, 2

[5] Hongming Chen, Ola Engkvist, Yinhai Wang, Marcus Olivecrona, and Thomas Blaschke. The rise of deep learning in drug discovery. *Drug discovery today*, 23(6):1241–1250, 2018. 2

[6] F. Chollet. Xception: Deep learning with depthwise separable convolutions. In *CVPR*, pages 1800–1807, 2017. 5

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 5

[8] Hiroshi Fukui, Tsubasa Hirakawa, Takayoshi Yamashita, and Hironobu Fujiyoshi. Attention branch network: Learning of attention mechanism for visual explanation. In *CVPR*, 2019. 2

[9] Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3681–3688, 2019. 2

[10] Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. Towards automatic concept-based explanations. In *Advances in Neural Information Processing Systems*, 2019. 2, 3

[11] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. *arXiv preprint arXiv:1704.01212*, 2017. 2

[12] Bryce Goodman and Seth Flaxman. European union regulations on algorithmic decision-making and a "right to explanation". *AI magazine*, 38(3):50–57, 2017. 1

[13] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016. 2

[14] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in neural information processing systems*, pages 1024–1034, 2017. 2

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 8

[16] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 2

[17] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR, 2018. 2

[18] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie J. Cai, James Wexler, Fernanda B. Viégas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *ICML*, 2018. 1, 2

[19] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016. 2

[20] Kunpeng Li, Ziyan Wu, Kuan-Chuan Peng, Jan Ernst, and Yun Fu. Tell me where to look: Guided attention inference network. In *CVPR*, 2018. 2

[21] Kunpeng Li, Ziyan Wu, Kuan-Chuan Peng, Jan Ernst, and Yun Fu. Tell me where to look: Guided attention inference network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9215–9223, 2018. 8

[22] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1261–1270, 2017. 2

[23] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *CVPR*, 2015. 1

[24] Christopher Morris, Martin Ritzert, Matthias Fey, William L Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4602–4609, 2019. 2, 4

[25] Will Norcliffe-Brown, Stathis Vafeias, and Sarah Parisot. Learning conditioned graph structures for interpretable visual question answering. In *Advances in neural information processing systems*, pages 8334–8343, 2018. 2

[26] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, 2014. 2

[27] Mary Phuong and Christoph Lampert. Towards understanding knowledge distillation. In *International Conference on Machine Learning*, pages 5142–5151, 2019. 2

[28] Antonio Polino, Razvan Pascanu, and Dan Alistarh. Model compression via distillation and quantization. *arXiv preprint arXiv:1802.05668*, 2018. 2

[29] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386*, 2016. 1

[30] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Ba-

tra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017. 1, 2, 3

[31] Dinggang Shen, Guorong Wu, and Heung-Il Suk. Deep learning in medical image analysis. *Annual review of biomedical engineering*, 19:221–248, 2017. 1

[32] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. *arXiv preprint arXiv:1703.01365*, 2017. 1

[33] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 5

[34] Damien Teney, Lingqiao Liu, and Anton van Den Hengel. Graph-structured representations for visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2017. 2

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017. 1

[36] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017. 2

[37] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *CVPR*, 2017. 1

[38] Lezi Wang, Ziyan Wu, Srikrishna Karanam, Kuan-Chuan Peng, Rajat Vikram Singh, Bo Liu, and Dimitris Metaxas. Sharpen focus: Learning with attention separability and consistency. In *ICCV*, 2019. 2

[39] Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. Representation learning on graphs with jumping knowledge networks. *arXiv preprint arXiv:1806.03536*, 2018. 2

[40] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014. 1

[41] Meng Zheng, Srikrishna Karanam, Ziyan Wu, and Richard J Radke. Re-identification with consistent attentive siamese networks. In *CVPR*, 2019. 2

[42] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 1, 2