Revealing Similar Semantics Inside CNNs: An Interpretable Concept-based Comparison of Feature Spaces

Georgii Mikriukov[1,2][0000−0002−2494−6285], Gesina Schwalbe[1][0000−0003−2690−2478], Christian Hellert[1][0000−0002−5781−6575], and Korinna Bade[2][0000−0001−9139−8947]

[1] Continental AG, Germany
{firstname.lastname}@continental-corporation.com
[2] Hochschule Anhalt, Germany
{firstname.lastname}@hs-anhalt.de

Abstract. Safety-critical applications require transparency in artificial intelligence (AI) components, but convolutional neural networks (CNNs) widely used for perception tasks lack inherent interpretability. Hence, insights into what CNNs have learned are primarily based on performance metrics because these allow, e.g., for cross-architecture CNN comparison. However, these neglect how knowledge is stored inside. To tackle this yet unsolved problem, our work proposes two methods for estimating the layer-wise similarity between semantic information inside CNN latent spaces. These allow insights into both the flow and likeness of semantic information within CNN layers, and into the degree of their similarity between different network architectures. As a basis, we use two renowned explainable artificial intelligence (XAI) techniques, which are used to obtain concept activation vectors, i.e., global vector representations in the latent space. These are compared with respect to their activation on test inputs. When applied to three diverse object detectors and two datasets, our methods reveal that (1) similar semantic concepts are learned *regardless of the CNN architecture*, and (2) similar concepts emerge in similar *relative* layer depth, independent of the total number of layers. Finally, our approach poses a promising step toward semantic model comparability and comprehension of how different CNNs process semantic information.

Keywords: Explainable Artificial Intelligence · Network Comparison · Feature Space Comparison · Semantic Concept.

1 Introduction

The emerging use of artificial intelligence (AI) and especially CNNs in safety-critical applications such as automated driving and medicine, has made the interpretability and transparency [3,32] of these models increasingly essential, not least because industrial and legal standards demand sufficient evidence of developed AI modules for safe and ethical use[1,14]. Therefore, it is crucial to develop methods that reveal the model semantics, i.e., what was learned where inside, particularly in relation to other models. Such model comparability at the knowledge level can enhance the general understanding of model knowledge encoding, the influence of architectures, and possibly also datasets. Potential future applications include retrieval of dataset bias, informed model selection, and architecture modification.

One popular method of knowledge representation assessment within the field of XAI is an analysis of semantic concepts, where concepts correspond to real-world objects or notions [5,32,33]. These concepts are associated with vectors in the CNN feature space, the so-called concept activation vectors (CAV) [22,50]. By examining the CAVs and their responses to model inputs, experts can gain valuable insights into model operation.

This research proposes two architecture-agnostic strategies for estimating the similarity of feature spaces and semantic concepts in CNNs. These allow answering for any two CNN layers how similar they are regarding their learned concepts (unsupervised strategy) and any given set of user-defined concepts (supervised strategy). To achieve this, we use the concept analysis methods TCAV [22] (supervised) and ICE [50] (unsupervised) as the basis. Both generate CAVs for concept-related samples during training. The response of these CAVs to test data is then measured to determine the feature space similarity with respect to the given concepts. The contributions and findings of this work are the following:

- We conduct a **concept-based comparison of feature spaces** and show how the same semantic information is processed differently across various CNN backbones;
- We propose **an unsupervised and a supervised layer-wise approaches to compare the semantic information** encoded in CNNs, which are shown to yield intuitive and interpretable results regarding CNN knowledge inspection;
- The main findings of our concept-based comparison of feature spaces are: **same semantic concepts are learned across different CNN architectures** and can be extracted from proper layers, representations of **concepts are located at the same relative depth of the backbone** in feature spaces of different networks.

2 Related Work

Explainable AI. The field of XAI encompasses interpretability techniques [41] to explain the predictions of machine learning functions like neural networks (NNs) to a human. While ante-hoc approaches using interpretable models, e.g., produce human-understandable concept outputs [23,28,7], are preferable [39], we here concentrate on already trained CNNs. For post-hoc explainability, one can distill approximate interpretable surrogate models. Concept-based examples are flow-graphs [18], layer-wise concept hierarchies [47,48,49], decision trees [46,8], and rule sets [34,35]. However, the limited fidelity to the original CNN renders them unsuitable for quantitative CNN comparison. Other post-hoc methods concentrate on explaining the behavior of single samples. Such can be applied in an approximate model-agnostic manner [35,38] or model-specific based on the model internal processing, like prominent saliency methods [51,43,4,2]. Such local approaches, even if aggregated to global information like in [24], only give limited insights into concepts represented in the CNN internals. Instead, this work relies on concept analysis [40], i.e., XAI methods that allow direct insights into the human-understandable concepts learned by a CNN.

Concept Analysis. Early techniques associate single CNN units with concepts [5,33], disregarding the distributed nature of CNN representations. Supervised linear methods like state-of-the-art TCAV [22] associate concepts to latent space vectors. Further extensions to use cases like concept regression [15] and localization [29] also stuck to this principle. There are also non-linear alternatives like clustering [16,21] or NNs [9], which, however, pose additional requirements to the labels. Unsupervised approaches require no concept labels at all, like ICE [50] that applies matrix factorization to the latent space. Alternatives relying on the intelligent choice of concept candidate patches [12,11] lead to less interpretable results [50].

Network Comparison. Existing neural network comparison methods foremostly utilize performance or error-estimation metrics and qualitative manual observation based on visual analytics or XAI. Examples of object detection model analysis are the TIDE [6] metrics and visualizations toolbox, and the framework by Miller et al. [31] to analyze models' ability to handle false negative occurrences. More knowledge-based approaches measure the compliance with constraints like object relations [13,42] or temporal consistency [45].

3 Background

In contrast to the mentioned methods, our approach involves comparing feature spaces, i.e., the knowledge encoded in CNNs, through semantic concepts and their responses to various inputs. A (visual) semantic concept refers to a feature of an image that can be expressed in natural language (e.g., "head" or "green") [5,10]. Concepts can be associated with a numeric vector in the latent space, known as the concept vector [22,10]. The approaches used in this paper are shortly recapitulated in the following.

TCAV. TCAV [22] is a supervised concept analysis method that utilizes Concept Activation Vectors (CAVs) to represent concepts in the latent space of a NN. Parameters of CAVs correspond to those of a binary linear classifier that separates the feature space of a given layer in a concept-versus-rest manner. The classifier is trained using the activations of concept-related and unrelated samples. Geometrically, a CAV is the normal vector to the separation hyperplane and indicates the direction of the concept in the latent space. The similitude between a sample and concepts is defined by cosine similarity. This feature of CAVs can be employed for ranking of input samples by concept relevance.

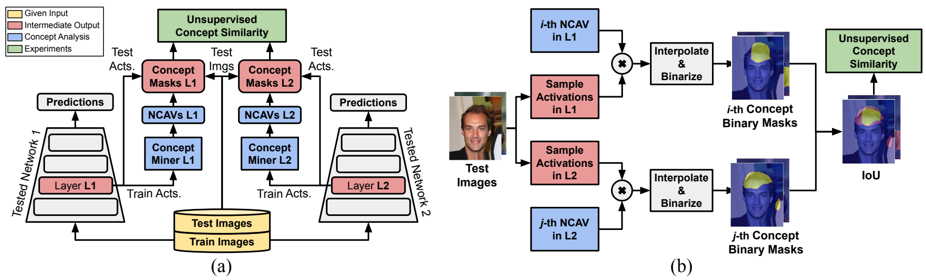


Fig. 1: Unsupervised evaluation of concept similarity (a) and concept similarity scoring (b).

ICE. The unsupervised ICE [50] approach employs Non-Negative Matrix Factorization (NMF) to mine a pre-defined small number of Non-negative CAVs (NCAVs) in the latent space. These NCAVs correspond to the most frequent activation patterns in convolutional filters caused by the training samples. NCAVs are then utilized to map input sample activations of dimensionality $C \times H \times W$ to $C' \times H \times W$ dimensional concept activations, where C, H, W, and C' represent the *channel*, *height*, *width*, and *concept* dimensions, respectively. Each of the C' concept activations of size $1 \times H \times W$ is normalized, interpolated to the original input size, and employed as a saliency map to highlight the concept-related regions. Examples of such binarized masks are presented in Fig. 5.

4 Semantic Comparability Methods

To address the gap in the literature on the comparison of model semantics, we introduce supervised and unsupervised approaches that use concept representations to compare feature spaces of CNN backbones. These methods rely on relative semantic similarity ranking of samples and overlap estimation of concept saliency maps. Section 4.1 and Section 4.2 provide details on the unsupervised and supervised comparison approaches, respectively.

4.1 Unsupervised Concept Similarity

The proposed unsupervised approach addresses two key questions: *"Are there similar concepts in feature spaces of different layers?"* and *"How similar are they?"*. We utilize ICE [50] to identify and extract the most prominent activation patterns, represented by NCAVs, in the feature spaces of different layers. Then, we measure the overlap between binarized concept saliency maps on test data to compare the similarity of extracted concepts in selected layers. Although we use ICE in our work, the general approach is not limited to this specific method and shall only showcase the usage of saliency methods.

Figure 1a depicts the layer-wise unsupervised knowledge comparison process in two trained *Tested Networks*, which may have different architectures. *L1* and

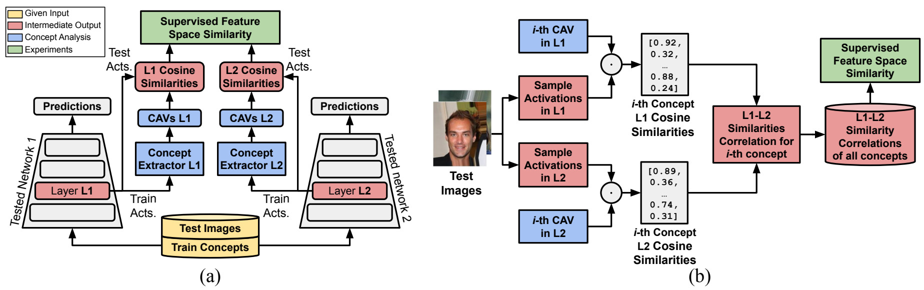


Fig. 2: Supervised concept-based estimation of feature space similarity (a) and feature space similarity scoring (b).

L2 are indices of analyzed layers. The first step involves using the activations of training samples (*Train Acts*) obtained from the *Train Images* to automatically extract concept vectors (*NCAVs*) with the *Concept Miner*. Subsequently, during the testing phase (Fig. 1b), the *NCAVs* are utilized to generate *Concept Masks* for activations (*Test Acts*) of *Test Images*. We process obtained continuous *Concept Masks* to evaluate concept similarity via masks. Each of them is normalized between 0 and 1, bilinearly interpolated to the same size (e.g., size of corresponding *Test Images*), and then binarized by thresholding, where the threshold value is a hyperparameter. After completing the preprocessing step, we calculate the *Unsupervised Concept Similarity* ($UCS_{i,j}$) for any pair of concepts by averaging the pixel-wise Jaccard index, also known as Intersection over Union (IoU), of a set of binary concept masks obtained for test samples:

$$UCS_{i,j} = \frac{1}{N}\sum_{k=1}^{N} \mathrm{IoU}(M_i^k, M_j^k)\ , \qquad \mathrm{IoU}(M_i^k, M_j^k) = \frac{\sum \mathtt{AND}(M_i^k, M_j^k)}{\sum \mathtt{OR}(M_i^k, M_j^k)}\ , \tag{1}$$

where, i and j are concept indices, N is the number of test samples, and $M_i^k, M_j^k \in \{\mathtt{True}, \mathtt{False}\}^{W\times H}$ are binary concept masks binary interpolated to the same fixed size $W \times H$ (defined by user) for the test sample at index k, `AND` and `OR` refer to pixel-wise intersection and union of binary masks, respectively.

Therefore, by comparing the projections of extracted concepts onto the input space, we indirectly measure the similarity between concepts and even describe the similarity of latent spaces. Using different test sets to excite and extract desired concepts in various layers, human experts can gain valuable insights into the knowledge similitude across different models.

4.2 Supervised Feature Space Similarity

The supervised approach aims to answer the question, *"How similar is the arrangement of feature spaces in compared layers with respect to given concepts?"*. In order to answer it, CAVs [22] are utilized as pivot vectors, around which we estimate the behaviuor of feature spaces with activations of test samples.

Figure 2a shows the supervised concept-based layer-wise feature space comparison process for two trained *Test Networks*. *L1* and *L2* are indices of analyzed layers. In the first stage, the *Concept Extractor* is employed to extract *CAVs* for each pair of the compared layers, using the training sample activations (*Train Acts*) obtained from concept-related images (*Train Concepts*). Next (Fig. 2b), to compare the feature spaces with respect to selected concepts, we compute the cosine similarity between the *CAVs* and the activations of test samples (*Test Acts*). Finally, we use the Pearson Correlation Coefficient (PCC) to compare the resulting series of cosine similarities and estimate the *Supervised Feature Space Similarity* ($SFSS_{u,v}$), which takes into account the ranking information of the samples as well as accounts for the relative orientation of sample activations in the feature space:

$$SFSS_{u,v} = \frac{1}{M}\sum_{i=1}^{M}\text{PCC}\left(\{\text{CS}^i_{u,k}\}^N_{k=1}), \{\text{CS}^i_{v,k}\}^N_{k=1}\right) \tag{2}$$

$$\text{CS}^i_{*,k} = \cos(CAV^i_*, x_{*,k})\ ,\ * \in \{u, v\} \tag{3}$$

where, indices u and v represent network layers, M is the total number of test concepts, i is the index of the currently tested concept, N is the total number of test samples, and $\text{CS}_{*,k}$ is a series of cosine similarities between the tested concept's CAV and the activation $x_{*,k}$ of the k-th test sample in layer $*$.

Although we propose using PCC to compute $SFSS_{u,v}$, it can be replaced with a statistical metric that preserves the rank order of values in the series. Spearman's rank correlation coefficient, for example, is a valid alternative.

Hence, by ranking and comparing the similarities between concept representations and test sample activations across multiple layers and models, we can indirectly estimate the generalized similarity and arrangement of their feature spaces.

5 Experimental Setup

Our experiments follow the methodology outlined in the previous section, which involves two main parts: 1) unsupervised layer-wise estimation of semantic similarity with binary concept masks (Sec. 4.1); and 2) supervised layer-wise comparison of model feature spaces with sample semantic similarity rankings (Sec. 4.2). In the subsequent subsections, we provide all details on the experimental setup.

5.1 Experimental Data of Test Images

We assume that the semantic complexity of the test data may affect the performance of the proposed methods. To investigate this, we conduct the evaluation using two datasets with similar knowledge categories but varying semantic complexity: MS COCO 2017 [25] and CelebA [27]. The CelebA is a low semantic diversity dataset, which comprises over 202,599 homogenous images with celebrity faces. In contrast, the MS COCO dataset is an object detection dataset with

(a) Concept: "legs" (b) Concept: "head" (c) Concept: "torso"

Fig. 3: Examples of generated MS COCO synthetic concept training samples.

high semantic diversity, featuring images of various objects in different contexts. This dataset includes images of different shapes with 2D object bounding box annotations. We utilized a subset of more than 2,000 MS randomly selected COCO images containing *person* class objects in various positions and situations. To streamline further visual validation, we only used non-crowd instances with bounding box areas of at least $20,000$ pixels. The resulting subset includes more than 2679 bounding boxes of people in different poses and locations extracted from 1685 images.

5.2 Models

We perform a semantic comparison of three object detectors of different paradigms and generations, which also feature different backbones, to evaluate the applicability of our approach:
- one-stage YOLOv5s[1] [20] with residual (res.) DarkNet [36,17] backbone;
- one-stage SSD[2] [26], which utilizes a VGG [44] backbone;
- two-stage FasterRCNN[3] [37] with inverted res. MobileNetV3 [19] backbone.

All models are trained on the semantically rich MS COCO [25], which is expected to contain semantic concepts relevant to both test datasets (Sec. 5.1). The models above are further referred to as YOLO5, SSD, and RCNN.

5.3 Concept Mining and Synthetic Concept Generation

The effectiveness of supervised concept-based analysis heavily relies on the quality of the concept-related training data. Unfortunately, publicly available datasets with concept labels are scarce, and existing ones may not be suitable for all research domains and tasks To address this issue, we suggest generating synthetic concept samples using concept information automatically extracted from task-specific datasets.

For this, we mine concept-related superpixels (image patches) with ICE [50] from MS COCO bounding boxes of the *person* class with an area of at least

[1] https://github.com/ultralytics/yolov5
[2] https://pytorch.org/vision/stable/models/ssd
[3] https://pytorch.org/vision/stable/models/faster_rcnn

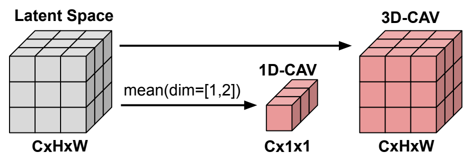


Fig. 4: Generation of 3D- and 1D-CAV representations.

20,000 pixels (Sec. 5.1). For experiments, we selected 3 concepts, each comprising 100 superpixels and semantically corresponding to labels "legs", "head", and "torso". Concepts were extracted from YOLOv5s layers `8.v3.c`, `9.v1.c`, and `10.c` respectively.

In order to create a synthetic concept sample, 1 to 5 concept-related superpixels are randomly selected and placed on a background of random noise drawn from a uniform distribution. Figure 3 shows examples of MS COCO synthetic concepts. Additionally, we rescale the superpixels by a factor between 0.9 and 1.1 before placement. The dimensions of the generated samples are set to be 640×480 pixels.

To conduct experiments on the CelebA dataset, we utilized four concepts extracted from the `7.conv` layer of YOLO5 (based on the results of Experiment 1, see Sec. 6.1). These concepts correspond to semantic labels "hairs", "upper face", "lower face", and "neck". Each concept sample contains one concept superpixel. The CelebA concept sample has the same size as the dataset sample, i.e., 178×218 pixels. The top row of Figure 5 displays examples of concept masks, which are utilized for concept superpixel cropping.

5.4 Dimensionality of Concept Activation Vectors

The TCAV [22] method employs 3D-CAVs to represent concepts. However, an alternative approach is to use a 1D-representation, as concept information can be encoded in the linear combination of feature space channels [10,50]. The use of a 1D-CAV offers several benefits [30] over the 3D-CAV: 1) it is more stable and computationally efficient, as it reduces the number of computational parameters, which is particularly important for a layer-to-layer comparison of deep backbones; 2) it is translation invariant since the spatial information of the concept is aggregated and only the presence or absence of the concept affects the channel activation strength. Given the mentioned advantages, we have opted to utilize 1D-CAVs in our experiments for supervised feature space comparison (Section 4.2).

Figure 4 illustrates the process of obtaining 1D- and 3D-CAVs, where `C`, `H`, and `W` represent the *channel*, *height*, and *width* dimensions, respectively. The arrows indicate the concept extraction process (see Sec. 3), where all input representations are aggregated across the *height* and *width* dimensions before computing the 1D-CAV. When dealing with 1D-representations to compute the similarity between the concept and sample the sample activation undergoes the same aggregation across the *height* and *width* dimensions.

5.5 Experiment-specific Settings

Experiment 1: *Unsupervised Concept Similarity.* We carried out experiments on unsupervised concept similarity using datasets of varying semantic diversity to showcase how the input data influence concept mining. To train the NCAVs, we used 100 and 300 random samples from CelebA and MS COCO datasets, respectively, as explained in Section 4.1 and Section 5.1. Afterward, we evaluated the performance using another 100 samples of each dataset to compute our $UCS_{i,j}$ metric from Section 4.1. The experimental results are presented as a heatmap for each pair of layers.

For CelebA, we extracted 5 concepts per layer, while for MS COCO, the number of mined concepts is set to 10. We used a value of $BT = 0.25$ (see Sec. 6.1 for the analysis of the impact of BT values) for binarizing concept masks. Examples of resulting masks for different BT values can be seen in Fig. 8c in Sec. 6.1.

Experiment 2: *Supervised Feature Space Similarity.* We evaluated the layer-wise feature space similarity of neural networks by conducting tests using CAVs trained on synthetic concepts from MS COCO and CelebA (see Section 5.3). To measure the $SFSS_{i,j}$ ranking metric, we used 200 randomly sampled MS COCO images, and the results are presented as a heatmap, with each cell representing a layer combination. To plot the heatmaps, we selected 10 layers uniformly distributed over the backbone depth of the networks under test (see Sec. 6.2), which are listed in Table 1.

Table 1: Shorthands of selected CNN intermediate layers for experiments (b=block, f=features, e=extra, c=conv, v=cv).

NN	Layer id										
	0	1	2	3	4	5	6	7	8	9	10
YOLO5	4.v3.c	5.c	6.v3.c	7.c	8.v3.c	9.v2.c	13.v3.c	17.v3.c	18.c	20.v3.c	21.c
SSD	f.5	f.10	f.14	f.17	f.21	e.0.1	e.0.5	e.1.0	e.2.0	e.3.0	e.4.0
RCNN	5.b.3	7.b.2	8.b.2	9.b.2	10.b.2	11.b.3	12.b.3	13.b.3	14.b.3	15.b.3	16

6 Experimental Results

6.1 Unsupervised Concept Similarity

The experiments were carried out following the methodology outlined in Section 4.1, using the setup described in Section 5.5. Figures 6, 7, 8a, and 8b depict

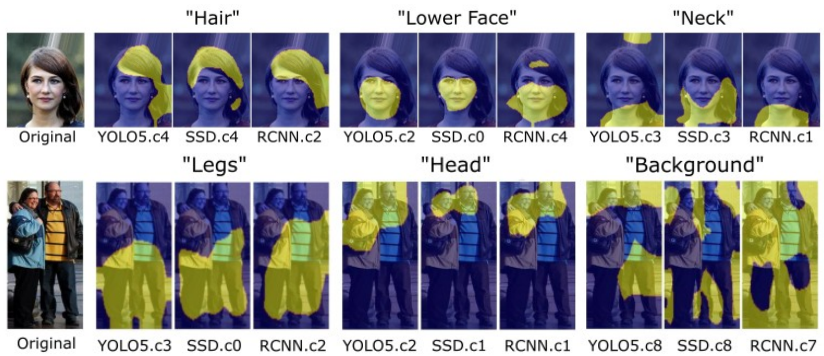


Fig. 5: Examples of binary concept masks obtained unsupervised using ICE for CelebA (top) and MS COCO (bottom) at binarization threshold $BT = 0.25$.

the concept similarity heatmaps of concepts mined in different layers, while Figures 5 and 8c show examples of the produced concept masks.

Impact of data semantic diversity. After manually inspecting concept masks and UCS-heatmaps generated with CelebA for different layers of the tested CNN backbones, we discovered that YOLO5 `7.c`, SSD `e.0.3`, and RCNN `15.b.3.0` layers (see Tab.1 for shorthands) had the most similar concepts and feature spaces. These are shown in the top row of Fig. 6. Furthermore, we found a one-to-one correspondence between extracted concepts in these layers. For convenience, we arranged the heatmap's horizontal axis by placing the most similar concept pairs on the diagonal. All of the extracted concepts are interpretable and correspond to semantic labels (with optimal layers for them in brackets): "hair" (`YOLO5.c4`, `SSD.c4`, `RCNN.c2`), "upper face" (`YOLO5.c1`, `SSD.c2`, `RCNN.c0`), "lower face" (`YOLO5.c2`, `SSD.c0`, `RCNN.c4`), "neck" (`YOLO5.c3`, `SSD.c3`, `RCNN.c1`), and "background" (`YOLO5.c0`, `SSD.c1`, `RCNN.c3`). Figure 5 demonstrates examples of binary masks generated for "hair", "lower face" and "neck" concepts.

In contrast to CelebA, not all concepts mined from MS COCO are human-interpretable and have counterparts in other models. This is demonstrated in the example of layers `8.v3.c`, `e.0.5`, and `15.b.3.0` of YOLO5, SSD, and RCNN (bottom row of Fig. 6). The higher semantic variability of input samples in MS COCO, where input samples may contain different sets of concepts, makes it more challenging to mine meaningful concepts. However, after a manual inspection of the most similar concept pairs highlighted by our approach, we found concepts corresponding to semantic labels such as "legs" (`YOLO5.cc`, `SSD.c0`, `RCNN.c2`), "head" (`YOLO5.c2`, `SSD.c1`, `RCNN.c1`), and "background" (`YOLO5.c8`, `SSD.c8`, `RCNN.c7`). Examples of their binary masks are depicted in the bottom row of Figure 5.

To summarize our observations, we conclude that datasets with high semantic variability may lead to a lower quality of automatic concept extraction results. Our proposed method can quantify and visualize this issue, assist in finding similar concepts, and identify layers with similar semantic information. Also, for

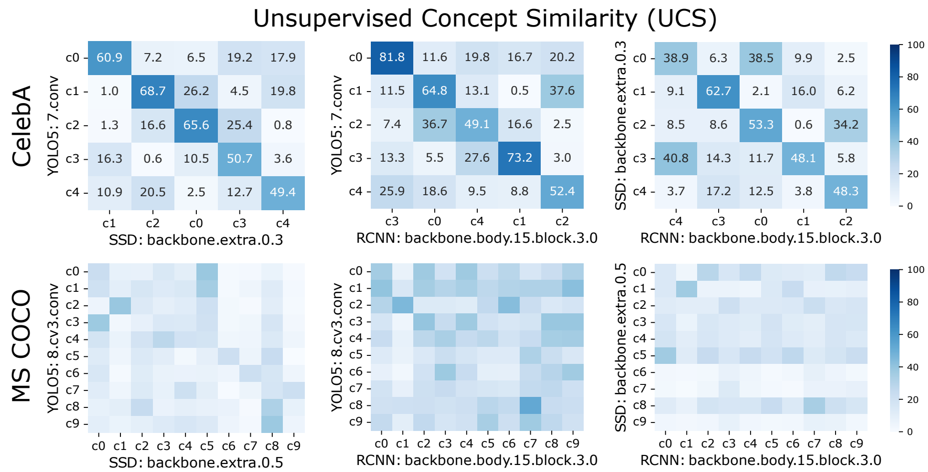
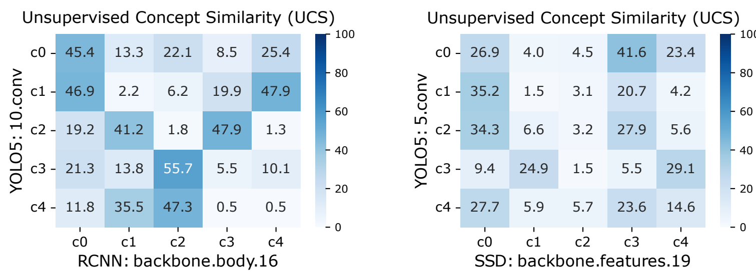


Fig. 6: Unsupervised concept similarity ($UCS_{i,j}$) estimates of different concepts c_i (x-axis) and c_j (y-axis) mined in *optimal layers.*



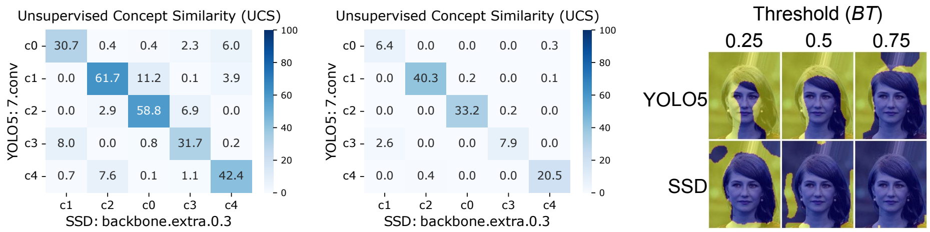
(a) Several concept counterparts. (b) `SSD.c2` has no counterpart.

Fig. 7: Unsupervised concept similarity ($UCS_{i,j}$) estimates of different concepts c_i (x-axis) and c_j (y-axis) mined in *non-optimal layers.*

a comprehensive unsupervised comparison of model concepts and feature spaces, we recommend using datasets with semantically homogeneous samples, similar to those found in CelebA.

Semantically similar layers identification. The proposed method enables the identification of the most similar layers in different networks. For example, the top row diagrams of Figure 6 display the layers with the highest level of feature space correspondence for CelebA, where each concept of one layer has a distinct counterpart in another. Another example in Figure 7 demonstrates non-optimal variations where a concept has multiple possible counterparts (Fig. 7a) or no matches (Fig. 7b).

In our experience, the main factors that influence the identification of similar layers are the number of concepts mined and the semantic complexity of the test dataset, as demonstrated in Figure 6.



(a) $BT = 0.5$ (b) $BT = 0.75$ (c) `YOLO5.c0` & `SSD.c1`

Fig. 8: Influence of concept mask binarization threshold value (BT) on unsupervised concept similarity estimation of concepts c_i (x-axis) and c_j (y-axis) for `7.conv` and `backbone.extra.0.3` layers of YOLO5 and SSD.

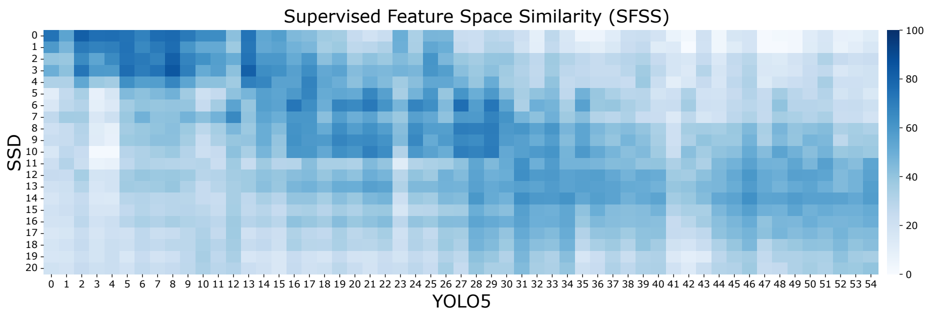


Fig. 9: Supervised feature space comparison of SSD and YOLO5 layers (all convolutional layers indexed from 0 to 20 resp. 54).

Concept robustness with regards to BT. The parameters of binary masks generated by ICE on test samples depend on the binarization threshold BT: higher BT values may reduce mask size and, hence, impact concept similarity, as illustrated in Fig. 8c. By leveraging this finding, we can also quantify the relative robustness of different concepts. As illustrated in Figures 8a and 8b, concepts like `YOLO5.c1` and `SSD.c2`, as well as `YOLO5.c2` and `SSD.c0`, exhibit the most resilience to changes in BT, making them the most robust ones.

6.2 Supervised Feature Space Similarity

Semantic information flow. Figures 9 and 10 display the layer-wise similarity between the feature spaces of models with respect to given concepts. Notably, the diagonal values in the heatmap of Figure 9 are more intense, indicating that semantic similarity is primarily influenced by the layer's relative depth in the backbone. Therefore, we can compare entire networks by evaluating a selected set of N layers (like in Table 1) evenly distributed throughout the backbone. Such an approach helps save processing power and time while preserving the global picture.

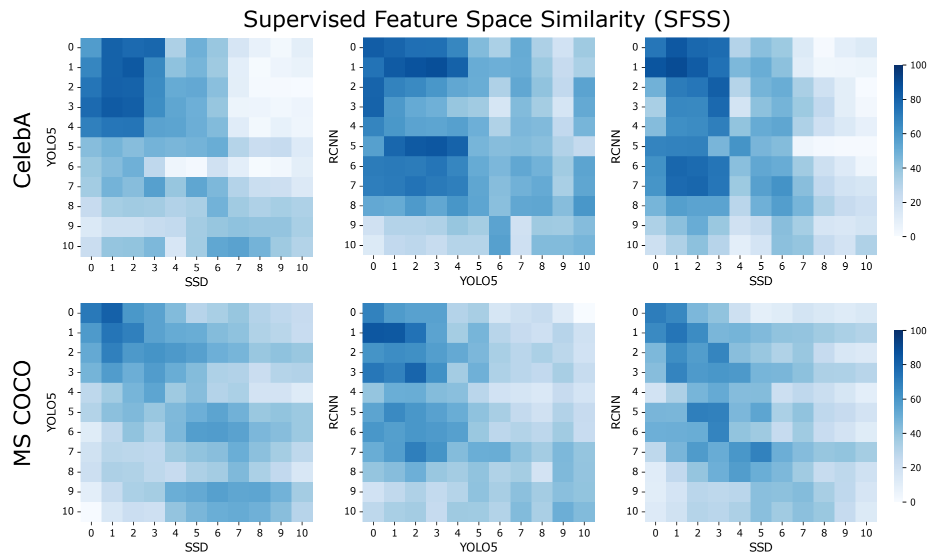


Fig. 10: Supervised feature space comparison of selected model layers (cf. Tab. 1).

Concept complexity. By examining Figure 10, we can observe that concepts derived from the CelebA dataset, which are lower in abstraction level and pertain to different parts of the face, lead to the greater similarity between layers in identical model pairs compared to the more complex body part concepts extracted from MS COCO. Additionally, these concepts are more clearly defined across a broader range of layers, resulting in distinguishable clusters (more prominent, darker regions) on the heatmaps. These observations imply that these concepts are more effectively represented in the feature spaces of the compared models.

Network architecture differences. Among the tested backbones, the MobileNetV3 backbone of RCNN exhibits a remarkably distinct behavior. Specifically, MobileNetV3 captures the same semantic information in two distinct regions within the network: at the beginning and in the middle. This can be observed in the middle and right columns of Figure 10, where we see a pattern with two distinct clusters (darker areas) along the vertical axis, between layers 0 and 3, and layers 5 and 8. This pattern is not observed in the direct comparison of the DarkNet and VGG backbones of YOLO5 and SSD, and, thus, only typical to MobileNetV3. We attribute this peculiarity to the distinctive network-building technique employed in inverted residual blocks of MobileNetV3, which propagates the semantics of tested concepts across the network efficiently. This, in turn, leads to a comparatively lesser decrease in semantic similarity of deeper layers in RCNN than in SSD and YOLO5.

Thereby, the proposed method for supervised feature space comparison also allows us to identify significant variations in the feature spaces and semantic representation learning across different models, and also can be used to judge optimal model architectures concerning interpretability.

6.3 Limitations and Future Work

Our approaches for concept analysis naturally inherit all limitations of data-driven methods, like dependence on high-quality data. Thus, as done here, manual visual validation of used CAVs and NCAVs remains inevitable. Our proposed semi-automatic data generation can be used to reduce labeling costs. Moreover, we found that the semantic diversity of the test data strongly affects the quality of the extracted concepts, and hence recommend using semantically homogeneous sets for testing.

A limitation inherent to using ICE concept masks is the differing and low mask resolution resulting from the different activation map dimensions. Choosing the scaling factors individually for any pair of layers may mitigate this, however, at the cost of comparability. Another issue is the dependence on the binarization threshold BT. Therefore, an interesting future direction could be to compare the non-binary concept masks directly.

In general, it will be interesting to apply our approach to further large NN architectures, e.g., transformers, and other visual tasks than object detection.

7 Conclusion and Outlook

In this research, we presented architecture-agnostic supervised and unsupervised methods for estimating the similarity of feature spaces in CNN backbones. Proposed methods help to reveal how the same semantic information is processed across various model backbones and enable identification of the semantically similar layers. We use semantic concept vectors, namely CAVs, and NCAVs, to assess the behavior of the latent space through the concept's response to the test data. Experiments on two datasets and three different backbone architectures trained on the same data revealed that regardless of the NN architecture, layers with similar semantic information can be found, as we found network layers with one-to-one concept correspondence. We also discovered that the feature space semantic information depends on the relative depth of the layer in the network backbone. Therefore, to compare different CNN backbones, it seems sufficient to compare only a subset of layers of uniform depth-distance in the backbone. Finally, our method provides valuable insights, which may be useful for applications like informed model selection, meta-analysis of network architectures, or dataset bias retrieval.

Acknowledgments

The research leading to these results is funded by the German Federal Ministry for Economic Affairs and Climate Action within the project "KI Wissen – Entwicklung von Methoden für die Einbindung von Wissen in maschinelles Lernen". The authors would like to thank the consortium for the successful cooperation.

References

1. 32, I.S.: ISO 26262-1:2018(En): Road Vehicles – Functional Safety – Part 1: Vocabulary (2018), https://www.iso.org/standard/68383.html
2. Achtibat, R., Dreyer, M., Eisenbraun, I., Bosse, S., Wiegand, T., Samek, W., Lapuschkin, S.: From" where" to" what": Towards human-understandable explanations through concept relevance propagation. arXiv preprint arXiv:2206.03208 (2022)
3. Arrieta, A.B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al.: Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. Information fusion **58**, 82–115 (2020)
4. Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one **10**(7), e0130140 (2015)
5. Bau, D., Zhou, B., Khosla, A., Oliva, A., Torralba, A.: Network dissection: Quantifying interpretability of deep visual representations. In: Proc. IEEE conf. computer vision and pattern recognition. pp. 6541–6549 (2017)
6. Bolya, D., Foley, S., Hays, J., Hoffman, J.: Tide: A general toolbox for identifying object detection errors. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16. pp. 558–573. Springer (2020)
7. Chen, C., Li, O., Tao, D., Barnett, A., Rudin, C., Su, J.K.: This looks like that: deep learning for interpretable image recognition. Advances in neural information processing systems **32** (2019)
8. Chyung, C., Tsang, M., Liu, Y.: Extracting interpretable concept-based decision trees from CNNs. In: Proc. 2019 ICML Workshop Human in the Loop Learning. vol. 1906.04664. CoRR (Jun 2019)
9. Esser, P., Rombach, R., Ommer, B.: A disentangling invertible interpretation network for explaining latent representations. In: Proc. 2020 IEEE Conf. Comput. Vision and Pattern Recognition. pp. 9220–9229. IEEE (Jun 2020). https://doi.org/10.1109/CVPR42600.2020.00924
10. Fong, R., Vedaldi, A.: Net2vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. In: Proc. IEEE conf. computer vision and pattern recognition. pp. 8730–8738 (2018)
11. Ge, Y., Xiao, Y., Xu, Z., Zheng, M., Karanam, S., Chen, T., Itti, L., Wu, Z.: A peek into the reasoning of neural networks: Interpreting with structural visual concepts. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2195–2204 (2021)
12. Ghorbani, A., Wexler, J., Zou, J.Y., Kim, B.: Towards automatic concept-based explanations. Advances in Neural Information Processing Systems **32** (2019)
13. Giunchiglia, E., Stoian, M., Khan, S., Cuzzolin, F., Lukasiewicz, T.: ROAD-R: The Autonomous Driving Dataset with Logical Requirements. In: IJCLR 2022 Workshops (Jun 2022)
14. Goodman, B., Flaxman, S.: European union regulations on algorithmic decision-making and a "right to explanation". AI Magazine **38**(3), 50–57 (Oct 2017). https://doi.org/10.1609/aimag.v38i3.2741, https://ojs.aaai.org/index.php/aimagazine/article/view/2741
15. Graziani, M., Andrearczyk, V., Marchand-Maillet, S., Müller, H.: Concept attribution: Explaining CNN decisions to physicians. Computers in Biology and Medicine **123**, 103865 (Aug 2020). https://doi.org/10.1016/j.compbiomed.2020.103865

16. Gu, J., Tresp, V.: Semantics for global and local interpretation of deep neural networks. CoRR **abs/1910.09085** (Oct 2019)
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. IEEE onf. computer vision and pattern recognition. pp. 770–778 (2016)
18. Hohman, F., Park, H., Robinson, C., Polo Chau, D.H.: Summit: Scaling Deep Learning Interpretability by Visualizing Activation and Attribution Summarizations. IEEE Transactions on Visualization and Computer Graphics **26**(1), 1096–1106 (Jan 2020). https://doi.org/10.1109/TVCG.2019.2934659
19. Howard, A., Sandler, M., Chu, G., Chen, L.C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., et al.: Searching for mobilenetv3. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1314–1324 (2019)
20. Jocher, G.: YOLOv5 in PyTorch, ONNX, CoreML, TFLite. https://github.com/ultralytics/yolov5 (Oct 2020). https://doi.org/10.5281/zenodo.4154370, https://doi.org/10.5281/zenodo.4154370
21. Kazhdan, D., Dimanov, B., Jamnik, M., Liò, P., Weller, A.: Now you see me (CME): Concept-based model extraction. In: Proc. 29th ACM Int. Conf. Information and Knowledge Management Workshops. CEUR Workshop Proceedings, vol. 2699. CEUR-WS.org (2020)
22. Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., et al.: Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In: Int. conf. machine learning. pp. 2668–2677. PMLR (2018)
23. Koh, P.W., Nguyen, T., Tang, Y.S., Mussmann, S., Pierson, E., Kim, B., Liang, P.: Concept bottleneck models. In: Int. conf. Machine Learning. pp. 5338–5348. PMLR (2020)
24. Lapuschkin, S., Wäldchen, S., Binder, A., Montavon, G., Samek, W., Müller, K.R.: Unmasking Clever Hans predictors and assessing what machines really learn. Nature Communications **10**(1), 1096 (Mar 2019). https://doi.org/10.1038/s41467-019-08987-4
25. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European onf. computer vision. pp. 740–755. Springer (2014)
26. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. pp. 21–37. Springer (2016)
27. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (ICCV) (December 2015)
28. Losch, M., Fritz, M., Schiele, B.: Interpretability beyond classification output: Semantic bottleneck networks. In: Proc. 3rd ACM Computer Science in Cars Symp. Extended Abstracts (Oct 2019)
29. Lucieri, A., Bajwa, M.N., Dengel, A., Ahmed, S.: Explaining AI-based decision support systems using concept localization maps. In: Neural Information Processing. pp. 185–193. Communications in Computer and Information Science, Springer International Publishing (2020). https://doi.org/10.1007/978-3-030-63820-7_21
30. Mikriukov, G., Schwalbe, G., Hellert, C., Bade, K.: Evaluating the stability of semantic concept representations in cnns for robust explainability. arXiv preprint arXiv:2304.14864 (2023)
31. Miller, D., Moghadam, P., Cox, M., Wildie, M., Jurdak, R.: What's in the black box? the false negative mechanisms inside object detectors. IEEE Robotics and Automation Letters **7**(3), 8510–8517 (2022)

32. Mittelstadt, B., Russell, C., Wachter, S.: Explaining explanations in ai. In: Proc. conf. fairness, accountability, and transparency. pp. 279–288 (2019)
33. Nguyen, A., Yosinski, J., Clune, J.: Understanding Neural Networks via Feature Visualization: A Survey. In: Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, pp. 55–76. Lecture Notes in Computer Science, Springer International Publishing (2019). https://doi.org/10.1007/978-3-030-28954-6_4
34. Rabold, J., Schwalbe, G., Schmid, U.: Expressive explanations of dnns by combining concept analysis with ilp. In: KI 2020: Advances in Artificial Intelligence. pp. 148–162. Lecture Notes in Computer Science, Springer International Publishing (2020). https://doi.org/10.1007/978-3-030-58285-2_11, https://arxiv.org/abs/2105.07371
35. Rabold, J., Siebers, M., Schmid, U.: Explaining black-box classifiers with ILP – empowering LIME with Aleph to approximate non-linear decisions with relational rules. In: Proc. Int. Conf. Inductive Logic Programming. pp. 105–117. Lecture Notes in Computer Science, Springer International Publishing (2018). https://doi.org/10.1007/978-3-319-99960-9_7
36. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
37. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015)
38. Ribeiro, M.T., Singh, S., Guestrin, C.: Model-agnostic interpretability of machine learning. arXiv preprint arXiv:1606.05386 (2016)
39. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence **1**(5), 206–215 (May 2019). https://doi.org/10.1038/s42256-019-0048-x
40. Schwalbe, G.: Concept Embedding Analysis: A Review. arXiv:2203.13909 [cs, stat] (Mar 2022)
41. Schwalbe, G., Finzel, B.: A comprehensive taxonomy for explainable artificial intelligence: A systematic survey of surveys on methods and concepts. arXiv e-prints pp. arXiv–2105 (2021)
42. Schwalbe, G., Wirth, C., Schmid, U.: Enabling verification of deep neural networks in perception tasks using fuzzy logic and concept embeddings (Mar 2022)
43. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proc. IEEE int. conf. computer vision. pp. 618–626 (2017)
44. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
45. Varghese, S., Gujamagadi, S., Klingner, M., Kapoor, N., Bär, A., Schneider, J.D., Maag, K., Schlicht, P., Hüger, F., Fingscheidt, T.: An unsupervised temporal consistency (TC) loss to improve the performance of semantic segmentation networks. In: 2021 IEEE/CVF Conf. Comput. Vision and Pattern Recognition Workshops. pp. 12–20 (Jun 2021). https://doi.org/10.1109/CVPRW53098.2021.00010
46. Wan, A., Dunlap, L., Ho, D., Yin, J., Lee, S., Petryk, S., Bargal, S.A., Gonzalez, J.E.: NBDT: Neural-backed decision tree. In: Posters 2021 Int. Conf. Learning Representations (Sep 2020)
47. Wang, A., Lee, W.N., Qi, X.: Hint: Hierarchical neuron concept explainer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10254–10264 (2022)

48. Wang, D., Cui, X., Wang, Z.J.: CHAIN: Concept-harmonized hierarchical inference interpretation of deep convolutional neural networks. CoRR **abs/2002.01660** (2020)
49. Zhang, Q., Cao, R., Shi, F., Wu, Y.N., Zhu, S.C.: Interpreting CNN knowledge via an explanatory graph. In: Proc. 32nd AAAI Conf. Artificial Intelligence. pp. 4454–4463. AAAI Press (2018)
50. Zhang, R., Madumal, P., Miller, T., Ehinger, K.A., Rubinstein, B.I.: Invertible concept-based explanations for cnn models with non-negative concept activation vectors. In: Proc. AAAI Conf. Artificial Intelligence. pp. 11682–11690 (2021)
51. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proc. IEEE conf. computer vision and pattern recognition. pp. 2921–2929 (2016)