Catch Your Concepts: A Flexible Concept Locator for Interpretable Visual Recognition

Qiyang Wan[1,2]
qiyang.wan@vipl.ict.ac.cn
Ruiping Wang[1,2] *
wangruiping@ict.ac.cn
Chengzhi Gao[1,2]
chengzhi.gao@vipl.ict.ac.cn
Xilin Chen[1,2]
xlchen@ict.ac.cn

[1] Key Laboratory of AI Safety of CAS, Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing, China

[2] University of Chinese Academy of Sciences, Beijing, China

Abstract

The interpretability of visual recognition models has attracted significant attention, particularly with the advancement of concept-based methods exemplified by Concept Bottleneck Models (CBMs), which decompose categories as a set of representations to human perceivable concepts. While much of the prior research has contributed to extending concept sets to enhance the **semantic interpretability** of the models, the spatially activated region of the concept, or **localization interpretability**, has been routinely neglected. Based on insights from cognitive psychology, both types of interpretability are crucial for constructing explanations that enhance human understanding and trust. This paper introduces a flexible **concept locator**, which serves as a simple yet effective plugin for localizing concepts in CBMs during training, and providing both semantic and localization explanations during inference. A set of dynamic masks, applicable to various backbones and gradually shrinking during the training steps, is employed to constrain the activation region of concept neurons. This allows the concepts used for category recognition to be concentratedly expressed in specific spatial areas. Experiments conducted on multiple datasets demonstrate that the proposed locator effectively locates concepts in CBMs, offering additional localization explanations at a relatively low cost.

1 Introduction

As the demand for reliability in various visual applications increases, the interpretability of visual models is receiving more attention [1, 3, 9, 12]. Beyond the fundamental need for AI model designers to elucidate the mechanisms of the models and clarify their decision-making processes, it is essential for models to provide intuitive, accurate, and comprehensible explanations to enhance the trust of non-expert users for facilitating better application [11]. As a bridge connecting AI and human users, designers of interpretable methods should consider human cognitive preference to provide appropriate explanations.

* Corresponding author.

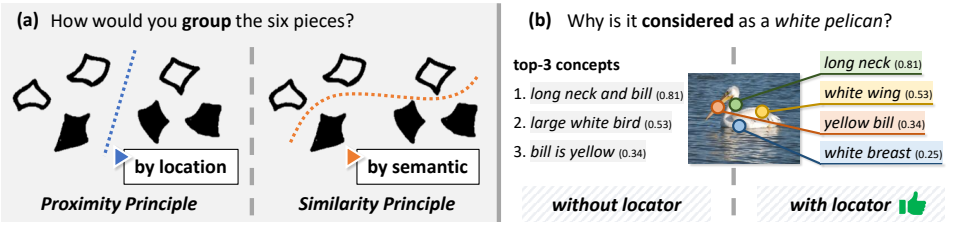

Figure 1: Motivation and illustration of concept localization. (a) The Proximity and Similarity Principle in Gestalt psychology [22]. If asked to divide a set of objects into two groups, how would you subconsciously group them? (b) Illustration of how concept localization enhances the interpretability for concept-based methods.

Gestalt psychology explores how people perceive and organize visual information through principles like **Proximity** and **Similarity** [10, 40]. As depicted in Fig. 1 (a), the Proximity Principle indicates that *elements close together are perceived as a group*; therefore, the objects will subconsciously be divided into two groups, left and right. The Similarity Principle states that *elements with shared visual traits, like color or shape, are seen as related*; so that the objects will be divided into two groups, white and black, according to color. These principles help our minds organize sensory input, and are crucial in user interface design for structuring and presenting visual elements to improve communication and usability.

Similarly, mainstream visual interpretability methods follow these principles to enhance user experience [16]. Specifically, the Similarity Principle indicates that explanations should represent *cross-category, similar visual elements* for object categories, referred to as **semantic interpretability**. The Proximity Principle advises employing visual elements *concentrated in specific spatial regions* to create explanations, known as **localization interpretability**. Concept Bottleneck Models (CBMs) [17] have been extensively studied for semantic interpretability by decomposing recognition into visual concepts combined for classification, and efforts by expanding concept lists [33, 44] or adopting open-vocabulary concepts [26, 39, 43] have advanced semantic interpretability. However, localization interpretability in such models remains overlooked, as they lack intrinsic alignment between concept labels and regions. Even when visualized using post-hoc tools [32], many correspondences are inaccurately represented. If a universally applicable concept locator can be implemented, the interpretability of CBMs will be further enhanced, as shown in Fig. 1 (b). However, the scarce region annotations relative to concepts make concept localization challenging.

In this paper, we propose a weakly supervised, plug-and-play **concept locator** for concept-based models without the requirement of fully-labeled concept annotations. Based on the characteristic that concepts in the real world are most intensively expressed within a specific spatial region, the proposed locator is attached as a plugin between the existing feature extractor and the concept projection layer of concept-based models, constraining the activation of concept neurons within specific spatial areas. Technically, as shown in Fig. 2, this is achieved through a set of dynamically changing Gaussian masks that modify the activation values of the concept activation map. In the training stage, as the epochs increase, the masks shrink from the entire image to the expected small areas. This smooth transition allows the model's convergence speed and peak performance to be almost unaffected while gaining the capability to **catch the concepts** at the end of the training phase. During inference, concentrated activations naturally provide a direct means of generating localization explanations. We directly visualize the constrained concept activation regions by threshold, demonstrating

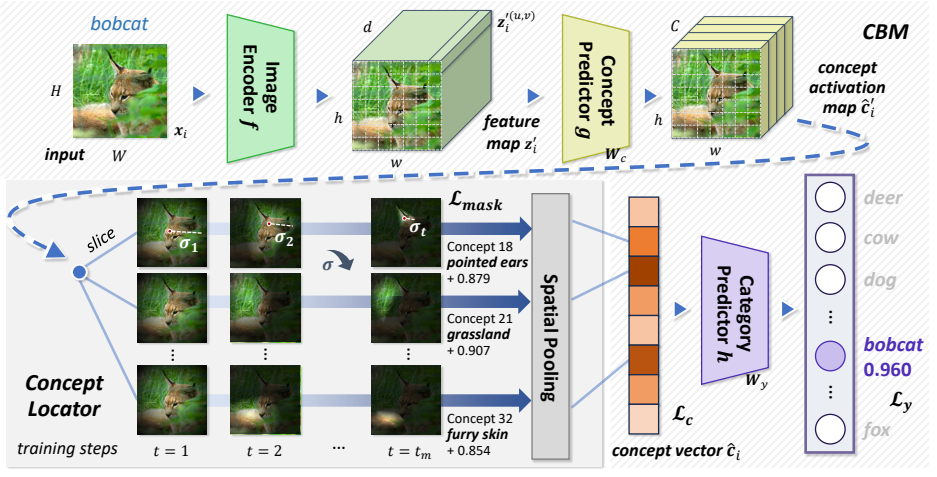

Figure 2: Illustration of the framework of the proposed concept locator on concept-based models. During training, the Gaussian mask used for constraining the concept activation map gradually shrinks (controlled by a dynamic parameter σ), enabling CBMs to locate concepts relying on the implicit concept co-occurrence across the categories.

that they can serve as highly effective localization explanations for concept-based recognition models. In addition, the concept locator can be flexibly attached to the majority of mainstream CBMs, achieving near-lossless performance when constraining the activation regions of concepts. We apply the proposed locator to multiple models and validate its effectiveness on various datasets.

We expect this study can inspire future research in exploring the development of comprehensive interpretable visual recognition models that incorporate both semantic and localization interpretability, so as to build visual applications that are more trustworthy to users.

2 Related Work

2.1 Interpretable Visual Recognition

Post-hoc explanation methods generate explanations by probing models that are fully trained with fixed parameters, including attribution methods [2, 21, 32, 35, 37, 46, 50], and concept probing methods like TCAV [14] and NetDissect [4]. These methods are not able to intervene in the model and can only passively detect the decision process. Another group is **self-interpretable** methods, which primarily achieve interpretable decisions by introducing interpretable structures when designing the model architecture, such as prototype-based methods [6, 24, 25, 30] and concept-based methods [17, 26, 29, 33, 39, 43, 44]. From another perspective outlined in Sec. 1, the objectives of the aforementioned methods can be categorized into two types: **localization interpretability** (including attribution- or prototype-based approaches) and **semantic interpretability** (concept-based approaches). There have been attempts to generate both types of explanation simultaneously, such as SB [20], DeViL [6], and DEAL [19]. They focus, respectively, on enhancing semantic representations in segmentation, generating natural language explanations for network neurons, and providing

concept-level localized explanations for VLMs, which complement the research area of this paper. These studies recognize the significance of diversity in explanations, providing new inspiration for the future work.

2.2 Localization in Visual Explanations

Post-hoc localization methods seek to elucidate how trained models make decisions by identifying the contribution of each input region to the final output. Saliency maps [34, 45] provide visual interpretations by highlighting the regions that are most influential in determining the model's prediction, often using gradient-based techniques to measure importance [23, 32, 35, 37]. As a self-interpretable model, ICNN family [47, 48, 49] is proposed to actively constrain the region of feature activation during training to enhance CNN's localization capability. Recently, ProtoPNet [5] represents a group of novel approaches that integrate the principles of prototype learning into CNNs. By matching patches of an input image to the learned prototypical parts, ProtoPNet [5] offers transparency and insight into the decision-making process. Subsequent methods such as ProtoTree [30], ProtoPool [24], Deformable ProtoPNet [8], and PIP-Net [25] have further optimized the generation quality of prototypical parts to actively provide better localization explanations. However, most of these methods either focus on highlighting the important area within an input image or on visualizing the specific activation region of a neuron in the CNN. The former targets individual images and provides local explanations, thereby lacking a global perspective on the models' decision-making processes. In contrast, the latter is directed at the model's underlying structure, often resulting in explanations devoid of human-understandable semantic meaning. This paper focuses on an appropriate level of granularity, specifically semantic concept layers, to provide comprehensible localization explanations for concepts in CBMs.

2.3 Concept-based Recognition Models

As representative methods of concept-based recognition models, Concept Bottleneck Models (CBMs) [17] introduce an intermediate layer that predicts human-understandable concepts before making final predictions. Subsequent advancements have focused on addressing CBM's limitations, such as the need for extensive concept annotations and potential performance trade-offs. Post-hoc CBM [44] leverages concept activation vectors [14] to learn a concept bank to avoid requirements for massive concept annotations. Label-free CBM [26] and LaBo [43] automatically generate a list of concepts using pre-trained language models and apply cross-modal models to calculate similarity, further reducing the need for annotation, allowing CBMs to scale to large-scale datasets. Another set of methods has been proposed by introducing residual layers [33, 44] or intervening and editing the list of concepts [15, 39] to improve the representation capacity of CBMs. Instead of pre-defined concepts, some works [29, 31] employ unsupervised concept discovery to extract concepts, thus mitigating the influence of prior and achieving a balance between model fidelity and semantic decoupling of concept explanations. These developments collectively advance the utility and applicability of CBMs in various scenarios.

3 Method

3.1 Problem Formulation

In the following sections of this paper, subscript i indicates the index from N samples in the dataset, subscript j indicates the index from C concepts, and subscript k indicates the index of M categories to be recognized.

Given a dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{c}_i, y_i)\}_{i=1}^N$, where $\mathbf{x}_i \in \mathbb{R}^{H\times W\times 3}$ represents the input image of size $H \times W$, $\mathbf{c}_i \in \mathbb{R}^C$ denotes the concept vector , and $\mathbf{y}_i \in \mathbb{R}^M$ is the one-hot category label, the task of concept-based object recognition is to establish a model that interprets a category prediction $\hat{y}_i$ through intermediate concepts $\hat{\mathbf{c}}_i$. Suppose that the d-dimension feature vector $\mathbf{z}_i \in \mathbb{R}^d$ is extracted from an image encoder f. Then the concept-based models are naturally decomposed into two main components:

1. Concept Predictor $g : \mathbb{R}^d \to \mathbb{R}^C$, which maps the input feature $\mathbf{z}_i$ into a space of interpretable concepts $\hat{\mathbf{c}}_i$.
2. Category Predictor $h : \mathbb{R}^C \to \mathbb{R}^M$, which maps these concepts $\hat{\mathbf{c}}_i$ to the final prediction logits $\hat{\mathbf{y}}_i$ of categories.

This sequential recognition process can be represented as

$$\mathbf{z}_i = f\left(\mathbf{x}_i; \theta_f\right),\ \hat{\mathbf{c}}_i = g\left(\mathbf{z}_i; \theta_g\right),\ \hat{\mathbf{y}}_i = h\left(\hat{\mathbf{c}}_i; \theta_h\right). \tag{1}$$

In addition, concept localization requires the model providing an activation map with the same size as the input image $\mathbf{x}_i$ when predicting the j-th concept, indicating the possible location of the j-th concept in the image. As a result of interpretable object recognition, such a concept-localizable model can simultaneously provide the most crucial set of concepts for the decision, the contribution values, and their spatial locations when predicting.

3.2 Baseline Concept Bottleneck Model

We firstly introduce our CBM baseline without localization capability in this section. In CBMs, the image encoder f is similar to a conventional non-interpretable model and can be either trainable or frozen. The obtained d-dimensional feature $\mathbf{z}_i$ is mapped to the predicted vector of concepts $\hat{\mathbf{c}}_i$ through a linear layer (without bias). The concepts $\hat{\mathbf{c}}_i$ are then mapped into the final category logits through another linear layer. Specifically,

$$\hat{\mathbf{c}}_i = g\left(\mathbf{z}_i; \theta_g\right) = \mathbf{W}_c\mathbf{z}_i, \quad \hat{\mathbf{y}}_i = h\left(\hat{\mathbf{c}}_i; \theta_h\right) = \mathbf{W}_y\hat{\mathbf{c}}_i, \tag{2}$$

where $\mathbf{W}_c$ and $\mathbf{W}_y$ are the projection matrix of the concept predictor g and the category predictor h respectively. Therefore, the learning objective of a supervised CBM can be expressed mathematically as

$$\mathcal{L}_{CBM} = \frac{1}{N}\sum_{i=1}^{N}\left(\lambda_c\mathcal{L}_c(\mathbf{c}_i, \hat{\mathbf{c}}_i) + \mathcal{L}_y(\mathbf{y}_i, \hat{\mathbf{y}}_i)\right), \tag{3}$$

where $\mathcal{L}_c(\cdot,\cdot)$ measures the discrepancy between the predicted and true concepts, usually with binary cross entropy, and $\mathcal{L}_y(\cdot,\cdot)$ assesses the prediction accuracy using the standard classification loss. The hyperparameter λ_c controls the balance between the two aspects.

If the dataset provides concept annotations at category or image level, they can be used directly as supervision for the concept vectors. When annotations are not available, previous works [26, 33, 39, 43, 44] have explored how to automatically obtain concept pseudo-labels. We utilize a pre-trained LLM and a multi-modal alignment model to generate the concept list and corresponding annotations for each category as Label-free CBM [26] does. In addition, we implement steps to filter out concepts that are low in quality and lack distinctiveness. Further details regarding the generated concepts are provided in the supplementary material.

3.3 Concept Locator

In subsequent paragraphs, we discuss how to implement localization capabilities through weak supervision. For the feature extractor f, the output $\mathbf{z}_i$ of the existing image encoder is expected to be modified from $\mathbb{R}^d$ to $\mathbb{R}^{h\times w\times d}$, denoted as $\mathbf{z}'_i$, where h and w represent the height and width of the feature map respectively. Each position of $\mathbf{z}'_i$, i.e. $\mathbf{z}'^{(u,v)}_i \in \mathbb{R}^d$, contains the local information at the coordinate (u,v), which can be regarded as the local feature. The modification differs according to the backbone structure of the image encoder. For the CNN backbone, such as ResNet, we remove the final fully connected and pooling layers, ensuring that the output features are naturally represented as the desired feature map. Each vector at a given coordinate corresponds to the local feature at that location.

For the concept predictor g accepting a modified feature $\mathbf{z}'_i \in \mathbb{R}^{h\times w\times d}$ as input, the original projection matrix $\mathbf{W}_c$ can function normally, forming the **concept activation map** $\hat{\mathbf{c}}'_i \in \mathbb{R}^{h\times w\times C}$, which can be regarded as C slices of $h\times w$ activations. To restrict each concept to be activated only within a concentrated area in spatial positions, it is necessary to impose constraints on the concept activation map $\hat{\mathbf{c}}'_i$. Firstly, analyze C slices separately, finding the coordinates of the maximum value for each, resulting in a $2\times C$ tensor $\mathbf{p}_i$ containing the positions of these maxima:

$$p_{ij} := (\mu_x, \mu_y) = \arg\max_{(u,v)} c^{(u,v)}_{ij}, j \in \{1\cdots C\}, \tag{4}$$

where p_{ij} refers to the coordinate of the j-th maximum, and c_{ij} refers to the j-th slice of $\hat{\mathbf{c}}'_i$. Subsequently, use $\mathbf{p}_i$ as the peak points to generate Gaussian masks $\mathbf{m}_i \in \mathbb{R}^{h\times w\times C}$, where

$$m^{(u,v)}_{ij} = e^{-\frac{(u-\mu_x)^2+(v-\mu_y)^2}{2\sigma^2}}, j \in \{1\cdots C\}. \tag{5}$$

Here σ controls the size of the mask, decreasing dynamically during the training epochs:

$$\sigma = \sigma_{ed} + \frac{1}{2}(\sigma_{st} - \sigma_{ed})\left(1 + \cos\left(\frac{t - t_w}{t_m - t_w}\cdot\pi\right)\right), \tag{6}$$

where σ_{st} and σ_{ed} denote the initial and expected final standard deviation of Gaussian masks, which is determined by datasets and the size of feature map. t represents the current step, and t_w, t_m represent the number of warm-up steps and the total number of steps respectively. Eq. 6 causes σ to decrease according to a cosine annealing schedule. The mask is dynamically computed in real-time during each forward pass, determined by the location of peak points in the current concept activation map. Specifically, the mask is not applied during the warm-up phase to ensure that the model converges quickly.

We multiply the mask $\mathbf{m}_i$ element-wise with the activation slices to obtain a region-constrained activation map $\hat{\mathbf{a}}_i$:

$$\hat{\mathbf{a}}_i = \mathbf{m}_i \odot \hat{\mathbf{c}}'_i. \tag{7}$$

Method	CIF-10	CIF-100	CUB	IN-1k	Places
Vanilla ResNet [13]	*93.7*	*74.0*	*76.7*	*76.4*	*53.4*
PCBM-h [14]	87.1	68.0	61.0	N/A	N/A
LFCBM [26]	86.4	65.1	74.3	67.5	49.0
LaBo [43]	87.9	69.1	71.8	68.9	N/A
Res-CBM [33]	88.0	67.9	62.4	N/A	N/A
DN-CBM [29]	87.6	67.5	N/A	72.9	53.5
VLG-CBM [36]	88.6	66.5	75.8	62.7	N/A
Loc-CBM (Ours)	**92.9**	**73.0**	**76.1**	**73.6**	**54.2**

Table 1: Recognition performance compared with the state-of-the-art CBMs on various datasets. Only a few models are capable of scaling to large-scale datasets (ImageNet and Places). *Vanilla* refers to the standard uninterpretable models.

Method	CUB
Vanilla ResNet [13]	*84.5*
ProtoPNet [5]	79.4
ProtoTree[24]	82.2
ProtoPool [30]	85.5
PIP-Net [25]	82.0
SPANet [42]	81.7
Loc-CBM (Ours)	85.1

Table 2: Recognition performance compared with prototype-based methods on CUB using the same backbone, ResNet-50.

Then we apply global average pooling to $\hat{\mathbf{a}}_i$ and obtain the concept vector $\hat{\mathbf{c}}_i \in \mathbb{R}^C$ consistent in shape within the baseline CBM, allowing for normal subsequent classification operations. An alignment loss $\mathcal{L}_{mask}$ is introduced to constrain the concept activation map $\hat{\mathbf{c}}'_i$:

$$\mathcal{L}_{mask} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_{MSE}\left(\hat{\mathbf{c}}'_i, \hat{\mathbf{a}}_i\right), \tag{8}$$

where $\mathcal{L}_{MSE}$ denotes the mean squared error. Thus, the overall optimization objective $\mathcal{L}$ is

$$\mathcal{L} = \mathcal{L}_{CBM} + \lambda_m \mathcal{L}_{mask}, \tag{9}$$

where λ_m is the hyperparameter.

In summary, the design of the proposed concept locator integrates a parameter-free mask operation following the concept activation map, making it compatible not only with CNN-type models but also with a variety of architectures such as ViT. More attempts are available in the supplementary material. In the experiment, incorporating the concept locator introduces negligible computational overhead in both the training and inference processes compared to the baseline model.

4 Experiment

In this section, we demonstrate the effectiveness of the proposed concept locator through extensive experiments. Due to the space limitation, we present only the key recognition and localization performance; additional results are available in the supplementary material.

4.1 Dataset and Model Implementation

Similar to previous work of concept-based self-interpretable recognition methods, we primarily conduct the experiments on a group of wide and representative datasets of visual recognition scenarios, ranging from generic object and fine-grained object recognition to scene classification. Experiments on CIFAR-10 [18], CIFAR-100 [18], and CUB-200-2011 [41] validate the effectiveness of the proposed plugin. In addition, large-scale datasets, i.e.

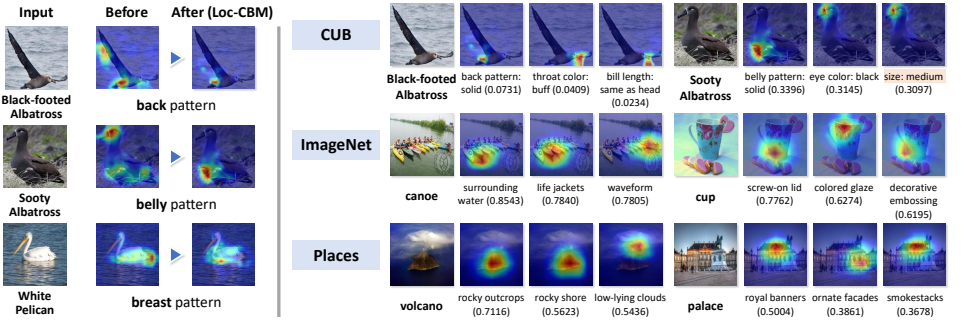

Figure 3: Qualitative results of the proposed Loc-CBM. Left: Comparison of concept activation regions before and after implementing the concept locator. Right: Examples of concept localization on CUB, ImageNet, and Places. Best viewed in color.

ImageNet [7] and Places-365 [51] are considered to demonstrate the proposed method's generalizability on generic datasets. Unless specifically mentioned, the concept annotations in the datasets (if any) will not be used; only the image samples and their category labels are required.

To fairly evaluate performance, following the previous works [26, 39], we conduct major experiments using a pre-trained CLIP-RN50 [28] as backbone for CIFAR, ImageNet, and Places, and a pre-trained ResNet-18 [38] for CUB when comparing with CBMs. When comparing with prototype-based methods [5, 24, 25, 30, 42] on CUB, a pre-trained ResNet-50 [27] is used. Refer to the supplementary material for more details[1].

4.2 Recognition Performance

We refer to the localization-capable model constructed in Sec. 3 as the Localizable CBM, abbreviated as Loc-CBM in the following sections and tables. The recognition results for accuracy compared with the state-of-the-art CBMs [26, 33, 39, 43, 44] on CIFAR-10 (CIF-10), CIFAR-100 (CIF-100), CUB-200-2011 (CUB), ImageNet (IN-1k), and Places-365 (Places) are shown in Tab. 1. The results indicate that the proposed Loc-CBM with the concept locator performs well on both small-scale and large-scale datasets, particularly achieving performance closer to standard black-box models.

Besides, in Tab. 2 we compare Loc-CBM with some prototype-based SoTAs with localization capability on CUB. The results indicate that the proposed method is effective even when compared to prototype-based methods. It should be noted that prototype-based methods are inherently well-suited to fine-grained datasets because their mechanism relies on comparing the similarity of local regions in images, as evidenced by the generally high accuracy of previous works on CUB in Tab. 2. However, due to the design limitation, most of these methods can only provide explanations of local similarities in images but cannot identify the concepts in the regions. The proposed Loc-CBM achieves comparable performance to these methods while providing the additional semantic explanation. This demonstrates the advantage of the proposed concept locator.

[1] The code is available at https://github.com/VIPL-VSU/Loc-CBM.

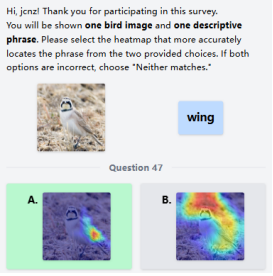

Figure 4: Interface for user study.

Localization Metric	Base CBM	Loc-CBM
Keypoint Alignment Error ↓	6.892	**0.959**
Concept Activation L1-norm ↓	3.902	**2.604**
User Satisfaction ↑	0.153	**0.317**

Table 3: Localization performance of Loc-CBM on CUB. The first two metrics respectively assess the accuracy and the concentration of activations within the concept activation map. *User Satisfaction* reflects users' subjective satisfaction score with the explanations.

4.3 Localization Performance

In addition to the recognition accuracy, we further evaluate the localization performance of Loc-CBM. As an intuitive comparison, Fig. 3 (left) illustrates the concept activation areas for the same set of concepts during training, without and with the concept locator. It can be observed that without constraints, the model's learned concepts may not activate in the target areas. When the concept locator is employed, the activated regions of the concepts are smaller and more precise. Fig. 3 (right) presents more examples from CUB, ImageNet, and Places, and more examples are shown in the supplementary material. Loc-CBM can decompose the category recognition into a combination of concepts while providing the specific activation location of each concept.

For quantitative results, it is difficult to evaluate without the ground-truth annotations of concept location, but can be indirectly measured using the keypoint annotations of parts available in CUB. Out of the 312 official attributes labeled in the dataset, 278 are related to these parts. Therefore, we measure the localization results of these concepts by the location of the corresponding parts. For each present part, the corresponding concept activation slices are extracted from the complete concept activation map. Then we measure the discrepancy between the activation slices and the target activations obtained by masking with the ground-truth maxima, which is referred to as *Keypoint Alignment Error*. In addition, we report the average sparsity of the concept activation, referred to as the *Concept Activation L1-norm*, to demonstrate that only necessary regions are activated with Loc-CBM. The results compared with the unconstrained CBM (referred to as Base CBM) are shown in Tab. 3.

To intuitively verify the impact of localization assistance for end users, we conduct a subjective evaluation through a user study to assess user satisfaction. 20 participants from diverse professional backgrounds – including bird enthusiasts, computer science undergraduates, and machine learning engineers – are asked to answer questions related to localization, with options presented in random order (Fig. 4). The results are presented in the last row of Tab. 3, in which the higher user satisfaction suggests that the explanations generated by Loc-CBM provide better interpretability compared to unconstrained CBMs.

Various metrics demonstrate that Loc-CBM significantly enhances localization accuracy compared to previous CBMs lacking localization capabilities. This improvement is attributed to the effectiveness of the proposed concept locator during training. More details of the evaluation, including ablation study, are available in the supplementary material.

5 Discussion

Active and Passive Localization. Many *passive* (post-hoc) localization explanation methods have been proposed for feature visualization, such as GradCAM [32], SmoothGrad [35], IG [37], and LRP [23]. These techniques are model-agnostic and can designate activation regions for each input sample. However, as outlined in Sec. 2.1, most regular CBMs are *active* interpretability techniques designed to develop self-interpretable recognition models. In contrast, *passive* localization explanation methods cannot directly affect the activation regions of concepts. While they facilitate model diagnosis, integrating effective constraints during training to enhance explanation accuracy remains challenging. In other words, merely knowing which regions the model activates is not sufficient. For intuitive comparison, examples of acquiring localization explanations through post-hoc methods are provided in the supplementary material. An *active* constraint is expected to be imposed to make the model function along the intended path.

Limitations. The proposed method shares a common issue with concept-based methods, where the automatically generated concept labels cannot precisely correspond to the patterns truly relied upon for model decisions. This can be observed by visualizing concept activation regions, a point that previous work has rarely addressed. The default setting of weakly supervised semantic-region alignment lacks adequate supervision, so the locator may struggle to focus on accurate and discriminative semantic features in some cases, especially when applied to fine-grained data or when spurious correlations exist. In addition, this supervision relies on the assumption that concepts are localized within small regions. While this holds true for the vast majority of concepts (see supplementary material for statistical data), it may not apply to certain concepts, such as background. These issues can be mitigated with more comprehensive annotations of concept location and more flexible mask size; however, this could involve substantial costs. Further discussion and examples are available in the supplementary material, and we look forward to future work continuing to explore the approach.

6 Conclusion

In conclusion, this paper contributes to the field of visual recognition by enhancing the interpretability of concept-based models through concept localization. By emphasizing both **semantic interpretability** – depicting how categories are represented by a set of concepts – and – **localization interpretability** – illustrating where concepts activate within an image – we underscore the importance of these dual aspects for building human trust in AI systems. We propose a flexible concept locator to offer a simple yet powerful solution to integrate the capability of concept localization into CBMs. The experiments on various datasets illustrate that the proposed method keeps recognition performance while favorably enriching the interpretability of visual recognition. This advancement is expected to provide the community with inspiration for delivering comprehensive explanations and enhance our understanding and trust in model predictions.

Acknowledgement

This work is partially supported by National Key R&D Program of China No. 2023YFF1105104, 2021ZD0111901, and Natural Science Foundation of China under contract No. U21B2025.

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.

[2] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLOS ONE*, 10(7):1–46, 2015.

[3] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador Garcia, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, Raja Chatila, and Francisco Herrera. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82–115, 2020.

[4] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[5] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: Deep learning for interpretable image recognition. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[6] Meghal Dani, Isabel Rio-Torto, Stephan Alaniz, and Zeynep Akata. Devil: Decoding vision features into language. In *Pattern Recognition*, pages 363–377, 2024.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[8] Jon Donnelly, Alina Jade Barnett, and Chaofan Chen. Deformable protopnet: An interpretable image classifier using deformable prototypes. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10255–10265, 2022.

[9] Rudresh Dwivedi, Devam Dave, Het Naik, Smiti Singhal, Rana Omer, Pankesh Patel, Bin Qian, Zhenyu Wen, Tejal Shah, Graham Morgan, and Rajiv Ranjan. Explainable ai (xai): Core ideas, techniques, and solutions. *ACM Computing Surveys*, 55(9), 2023.

[10] Michael W Eysenck and Marc Brysbaert. *Fundamentals of Cognition*. Routledge, 2018.

[11] William Ferguson, Dhruv Batra, Raymond Mooney, Devi Parikh, Antonio Torralba, David Bau, David Diller, Josh Fasching, Jaden Fiotto-Kaufman, Yash Goyal, Jeff Miller, Kerry Moffitt, Alex Montes de Oca, Ramprasaath R. Selvaraju, Ayush Shrivastava, Jialin Wu, and Stefan Lee. Reframing explanation as an interactive medium: The equas (explainable question answering system) project. *Applied AI Letters*, 2(4), 2022.

[12] David Gunning. Darpa's explainable artificial intelligence (xai) program. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, page ii, 2019.

[13] 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 (CVPR)*, 2016.

[14] 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). In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 2668–2677, 2018.

[15] Jeeyung Kim, Ze Wang, and Qiang Qiu. Constructing concept-based models to mitigate spurious correlations with minimal human effort. In *Computer Vision – ECCV 2024*, pages 137–153, 2025.

[16] Sunnie S. Y. Kim, Elizabeth Anne Watkins, Olga Russakovsky, Ruth Fong, and Andrés Monroy-Hernández. "help me help the ai": Understanding how explainability can support human-ai interaction. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, 2023.

[17] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pages 5338–5348, 2020.

[18] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

[19] Tang Li, Mengmeng Ma, and Xi Peng. Deal: Disentangle and localize concept-level explanations for vlms. In *Computer Vision – ECCV 2024*, pages 383–401, 2025.

[20] Max Losch, Mario Fritz, and Bernt Schiele. Semantic bottlenecks: Quantifying and improving inspectability of deep representations. *International Journal of Computer Vision*, 129(11):3136–3153, 2021.

[21] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[22] Wolfgang Metzger. *Laws of Seeing*. Mit Press, 2009.

[23] Grégoire Montavon, Alexander Binder, Sebastian Lapuschkin, Wojciech Samek, and Klaus-Robert Müller. *Layer-Wise Relevance Propagation: An Overview*, pages 193–209. Springer International Publishing, 2019.

[24] Meike Nauta, Ron van Bree, and Christin Seifert. Neural prototype trees for interpretable fine-grained image recognition. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14928–14938, 2021.

[25] Meike Nauta, Jörg Schlötterer, Maurice van Keulen, and Christin Seifert. Pipnet: Patch-based intuitive prototypes for interpretable image classification. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2744–2753, 2023.

[26] Tuomas Oikarinen, Subhro Das, Lam M. Nguyen, and Tsui-Wei Weng. Label-free concept bottleneck models. In *The Eleventh International Conference on Learning Representations*, 2023.

[27] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[28] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pages 8748–8763, 2021.

[29] Sukrut Rao, Sweta Mahajan, Moritz Böhle, and Bernt Schiele. Discover-then-name: Task-agnostic concept bottlenecks via automated concept discovery. In *Computer Vision – ECCV 2024*, pages 444–461, 2024.

[30] Dawid Rymarczyk, Łukasz Struski, Michał Górszczak, Koryna Lewandowska, Jacek Tabor, and Bartosz Zieliński. Interpretable image classification with differentiable prototypes assignment. In *Computer Vision – ECCV 2022*, pages 351–368, 2022.

[31] Simon Schrodi, Julian Schur, Max Argus, and Thomas Brox. Concept bottleneck models without predefined concepts. *arXiv preprint arXiv:2407.03921*, 2024.

[32] 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 *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017.

[33] Chenming Shang, Shiji Zhou, Hengyuan Zhang, Xinzhe Ni, Yujiu Yang, and Yuwang Wang. Incremental residual concept bottleneck models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11030–11040, 2024.

[34] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[35] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[36] Divyansh Srivastava, Ge Yan, and Tsui-Wei Weng. Vlg-cbm: Training concept bottleneck models with vision-language guidance. In *Advances in Neural Information Processing Systems*, volume 37, pages 79057–79094, 2024.

[37] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 3319–3328, 2017.

[38] Oleg Sémery. Sandbox for training deep learning networks. https://github.com/osmr/imgclsmob, 2019.

[39] Andong Tan, Fengtao Zhou, and Hao Chen. Explain via any concept: Concept bottleneck model with open vocabulary concepts. In *Computer Vision – ECCV 2024*, pages 123–138, 2025.

[40] Johan Wagemans, James H Elder, Michael Kubovy, Stephen E Palmer, Mary A Peterson, Manish Singh, and Rüdiger von der Heydt. A century of gestalt psychology in visual perception: I. perceptual grouping and figure-ground organization. *Psychological bulletin*, 138(6):1172—1217, 2012.

[41] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. Technical report, California Institute of Technology, 2011.

[42] Qiyang Wan, Ruiping Wang, and Xilin Chen. Interpretable object recognition by semantic prototype analysis. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 800–809, 2024.

[43] Yue Yang, Artemis Panagopoulou, Shenghao Zhou, Daniel Jin, Chris Callison-Burch, and Mark Yatskar. Language in a bottle: Language model guided concept bottlenecks for interpretable image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19187–19197, 2023.

[44] Mert Yuksekgonul, Maggie Wang, and James Zou. Post-hoc concept bottleneck models. In *ICLR 2022 Workshop on PAIR^2Struct: Privacy, Accountability, Interpretability, Robustness, Reasoning on Structured Data*, 2022.

[45] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Computer Vision – ECCV 2014*, pages 818–833, 2014.

[46] Hao Zhang, Yichen Xie, Longjie Zheng, Die Zhang, and Quanshi Zhang. Interpreting multivariate shapley interactions in dnns. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12):10877–10886, 2021.

[47] Quanshi Zhang, Ruiming Cao, Feng Shi, Ying Nian Wu, and Song-Chun Zhu. Interpreting cnn knowledge via an explanatory graph. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), 2018.

[48] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[49] Quanshi Zhang, Xin Wang, Ying Nian Wu, Huilin Zhou, and Song-Chun Zhu. Interpretable cnns for object classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3416–3431, 2021.

[50] 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 (CVPR)*, 2016.

[51] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464, 2018.