TU Delft

Delft University of Technology

Document Version
Final published version

Citation (APA)
Shi, X., Khademi, S., & van Gemert, J. (2019). Deep Visual City Recognition Visualization. In *NCCV 2019 – The Netherlands Conference on Computer Vision* (pp. 1-6)

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
In case the licence states “Dutch Copyright Act (Article 25fa)”, this publication was made available Green Open Access via the TU Delft Institutional Repository pursuant to Dutch Copyright Act (Article 25fa, the Taverne amendment). This provision does not affect copyright ownership.
Unless copyright is transferred by contract or statute, it remains with the copyright holder.

Sharing and reuse
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.

This CVPR Workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Deep Visual City Recognition Visualization

Xiangwei Shi, Seyran Khademi, Jan van Gemert
PRB, Computer Vision lab
Delft University of Technology

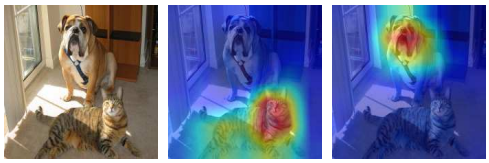
Abstract

Understanding how cities visually differ from each others is interesting for planners, residents, and historians. We investigate the interpretation of deep features learned by convolutional neural networks (CNNs) for city recognition. Given a trained city recognition network, we first generate weighted masks using the known Grad-CAM technique and to select the most discriminate regions in the image. Since the image classification label is the city name, it contains no information of objects that are class-discriminate, we investigate the interpretability of deep representations with two methods. (i) Unsupervised method is used to cluster the objects appearing in the visual explanations. (ii) A pretrained semantic segmentation model is used to label objects in pixel level, and then we introduce statistical measures to quantitatively evaluate the interpretability of discriminate objects. The influence of network architectures and random initializations in training, is studied on the interpretability of CNN features for city recognition. The results suggest that network architectures would affect the interpretability of learned visual representations greater than different initializations.

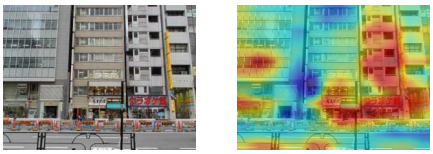
1. Introduction

Understanding how cities visually differ from each others is interesting for planners, residents, and historians. Automatic visual recognition is now making great progress which can help identifying how cities visually differ. Creating interpretable convolutional neural network (CNN) is a fascinating path that may lead us towards trustworthy AI [3, 9, 10, 12, 14, 17, 18]. Understanding CNN filters provides us with valuable insight on decision making criteria for a specific task. Visual features such as objects and parts are examples of high-level semantics that are consistent with how humans understand and analyze images [2, 5, 16]. Accordingly, we investigate and evaluate the interpretability of learned discriminate objects in city recognition CNNs.

Visualization of CNN filters are a popular techniques for analyzing CNNs. In this work, we build on top of gradient-weighted class activation mapping (Grad-CAM) method [9]



(a) cat and dog image and visualizations



(b) Tokyo image and visualization

Figure 1. Visualization examples of image classification (supervised) and city recognition. (a) From left to right: original image with a cat and a dog and the visualization with 'cat'/'dog' information (highlighting cat/dog); [9]. (b) From left to right: original image of Tokyo; visualization with 'Tokyo' information (highlighting, e.g., building, fence and signboard).

to generate class-discriminate visualizations, for our city recognition CNNs. Grad-CAM generates visualizations on the input images with highlight of discriminate regions by analyzing learned convolutional features and taking the information of the fully connected layers into consideration. Grad-CAM does not need to alter structure of the trained CNNs and is model-agnostic.

One common assumption in interpretability analysis of discriminate networks is that the image label matches with a single dominant object. However, interpreting CNNs for city recognition deviates from this assumption as the labels of images for place recognition are places, such as names of nations or cities, which is different from discriminate objects appearing in city images such as certain architecture or vegetation type. The information on these discriminate objects is an unknown priori, including what objects and how many kinds of them are present in the data and even the same kinds of objects could appear in images of different classes. Figure 1 shows an example. Obtaining the information of discriminate objects and how to interpret these visual objects in a dataset are the main stream of our study.

This work offers a method to both qualitatively and quantitatively evaluate interpretibility of city recognition

CNNs. While qualitative methods judge the interpretibility of networks directly by human [2,9,17], quantitative methods compute a mathematical expression that reflects the trustworthiness. Examples of the quantitative techniques are [2,16] that compute Intersection over Union (IoU) score to evaluate the interpretability across networks as an objective confidence score. In [3, 9] localization precision of visualizations through Pointing Game [15] is evaluated.

To the best of our knowledge there is no work that quantitatively measures the interpretibility of CNN in a holistic manner. Previous work consider supervised visualization where the the labels of objects that are localized in the image are consistent with the class labels [7–9, 18].

We raise the following research questions in this paper and we try to address them via relevant experiments.

- Are the deep representations learned by the city recognition CNNs interpretable?
- How to measure and evaluate the interpretability of in weakly supervised network?
- Do different architectures or initializations of CNNs affect the interpretability?

2. Methodology

We summarize our proposed interpretability investigations roughly in several steps:

1. Weighted masks are generated in the ultimate layer of any given trained CNNs model that classifies images from different cities, using Grad-CAM that highlights the class-discriminate regions of the test image. A visual explanation is generated using a threshold and weighted mask to cover unimportant regions on test image for classification.
2. Visual explanations are visualized using t-SNE to detect meaningful patterns in an unsupervised manner.
3. A pretrained segmentation model is used to annotate the objects in the test images pixelwise.
4. The normalized distribution of the objects annotated in visual explanations for each class is plotted to see if there is a significant skew towards certain objects.

2.1. Generating Visual Explanations

We adopt Grad-CAM [9] as our visualization technique to generate visual explanations for each test image. Selvaraju et al. [9] proposed Grad-CAM based on the work of [18], to map any class-discriminate activation of last convolutional layers onto input images. In the localization heat-maps ($L^c_{\text{Grad-CAM}}$), the values of significance are calculated in pixel level and the important regions are highlighted on input images. The localization heat-maps can be computed by a linear combination of weighted forward activation maps as proposed in [9]. Note that the weighted masks $mask_norm$ are generated by normalizing localization heat-maps to ensure the values of significance range between $[0, 1]$ for each weighted mask. Additionally, we set a threshold to select important regions (pixels) from the weighted masks to generate visual explanations. See Figure 2 for illustration.

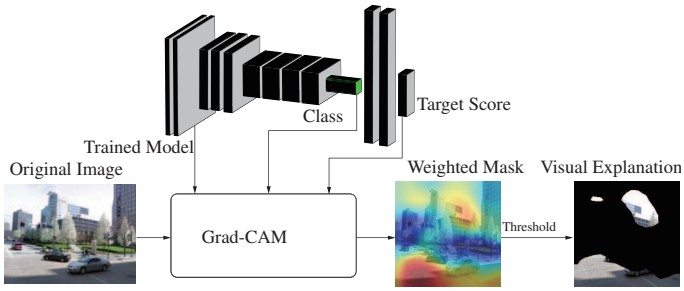


Figure 2. The pipeline of generating weighted masks and visual explanations with Grad-CAM [9] for city recognition CNNs.

2.2. Clustering Weighted Masks

Due to the lack of object labels appearing in visual explanations, we adopt unsupervised method to cluster visual explanations directly to recognize potential patterns. Proper descriptors needs to be extracted to cluster the visual explanations. Instead of extracting descriptors from visual explanations, we take the weighted masks $mask_norm$ as descriptors and cluster them. We use t-distributed stochastic neighbor embedding (t-SNE) [6] for clustering and dimensionality reduction.

2.3. Quantifying Interpretability

The aim of this study is to examine the interpretability of deep representations learned from city recognition CNNs, therefore it is necessary to obtain the information of what objects appear as discriminate in the images. In our work, we first use semantic segmentation model to label the objects in pixel level. This pretrained segmentation model should be able to recognize all classes of objects appearing in images. Hence the class information of objects can be used for evaluating the interpretability of deep representations quantitatively.

Some quantitative measurements of interpretability in previous researches, such as IoU [2, 16] and Pointing Game [3, 9, 15], cannot be used for city recognition CNNs, since there is inconsistency between the class information of city images and the class information of objects appearing in city images. Alternatively, we suppose objects appearing in the visual explanations are class-discriminate and their frequent occurrence reflects the interpretability of deep representations. To quantify this metric we calculate the number of pixels for different objects in visual explanations of the test images. To rule out the biases of different classes, we normalize the numbers of pixels of class-discriminate object p in the visual explanations M_P to the pixels of the

same object in all images from that dataset N_P:

$$R_p^c = \frac{\sum_{i=1}^{N^c} M_{p,i}}{\sum_{i=1}^{N^c} N_{p,i}}, \quad (1)$$

where N^c is the total number of city images of class c, indexed by i. For instance, p can be trees where R_p^c reflects the ratio of trees appearing as class discriminate in the class Tokyo to the whole trees appearing in this class. R_p^c is a quantifiable bounded measure of object significance varying between $[0, 1]$, where 0 means non-discriminate with respect to other classes and 1 means very discriminate.

3. Experiments

3.1. Datasets

We use two datasets of city images, which are **Tokyo 24/7** and **Pittsburgh** introduced from [1] to obtain city recognition CNNs.

- **Tokyo 24/7**: This dataset contains 76k dataset images. For the same spot, 12 images were taken from different directions.
- **Pittsburgh**: This dataset contains 250k database images. For the same spot, 24 images were taken from 12 different direction and 2 different angles.

To avoid unbalanced datasets, we only use 76k Pittsburgh images. All images are divided into training, validation and test datasets with the proportions as 6:2:2. These two datasets do not contain any information of objects.

3.2. Experimental Setup

We train four different image classification CNNs models to classify city images. The network architectures include VGG11 [11], ResNet18 [4] and two other shallow networks (as shown below in Table 1), Simple and Simpler. These four image classification networks are used for interpreting deep representations of city recognition CNNs and investigating the influence of network architectures on the interpretability.

All four models are trained with the same training setup. The loss function is cross-entropy function, and Adam optimizer is applied. The initial learning rate is set as 0.0001 and is multiplied by 0.1 every 10 epochs. The accuracies of four models are 99.98%, 99.96%, 99.31% and 98.18%.

3.3. Clustering Weighted masks

To address our first research question on whether the learned representation in the last convolutional layer of our trained CNN are interpratable by human or not, we conduct the following experiment. Using t-SNE, the weighted masks ($mask_norm$) are clustered in an unsupervised manner instead of visual explanations due to the lack of objeclevel labels and the irregular shapes of black regions around visual explanations. We apply PCA to extract 50 dominating features prior the the t-SNE clustering and dimensionality reduction. Figure 3 shows the scatter plots of VGG11 clustering results with label information of city images.

Table 1. Configurations of two shallow networks. In this table, 'convN×N' represents convolutional layer with a N×N filter, and each convolutional layer is followed by a ReLU activation function. The number after hyphen represents the number of channels in the corresponding feature map, and the numbers in the brackets is the size of filter in max pooling layer.

Simple	Simpler
Input images:224×224×3(RGB)	
conv5×5-20	conv9×9-20
max pooling(2×2)	
conv7×7-64	conv9×9-64
max pooling(2×2)	
conv5×5-96	conv9×9-96
max pooling(2×2)	
conv7×7-128	
max pooling(2×2)	
fully connected-4096	
fully connected-100	
fully connected-number of classes:2	

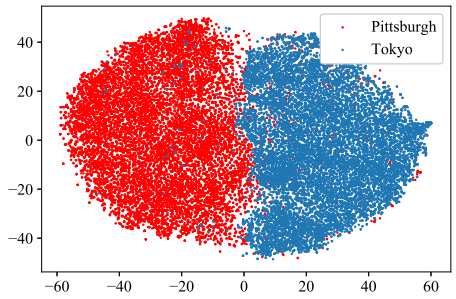


Figure 3. Scatter plots of clustering results of VGG11 with city information of images. Each point represents a weighted mask generated from each test image. Most of weighted masks from different datasets are separable in terms of city label information.

From the Figure 3, the clustering result that the weighted masks of test images are separable, is consistent with the high accuracy of VGG11. To visually exhibit the objects information in visual explanations that is related with the interpretability of deep representations, we next replace the points with visual explanations and demonstrate the relation between clustering result and class-discriminate objects intuitively. Due to the considerable number of test images, we randomly select around 500 visual explanations generated from VGG11 model to exhibit, as shown in Figure 4.

Based on the data visualization results shown in Figure 4, we can see that the result of our clustering leads to a collection of visually similar objects in a 2D map, which indicates that the VGG11 model learns semantically meaningful discriminate objects in the last convolutional layer. Although

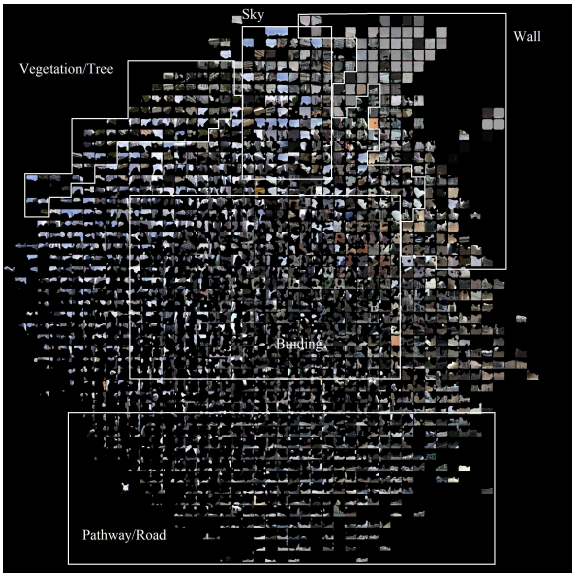


Figure 4. Exhibiting t-SNE results with visual explanations of VGG11. After replacing clustering reslut with visual explanations of test images, similar class-discriminate objects in visual explanations are clustered together. The information of these objects is obtained directly by human.

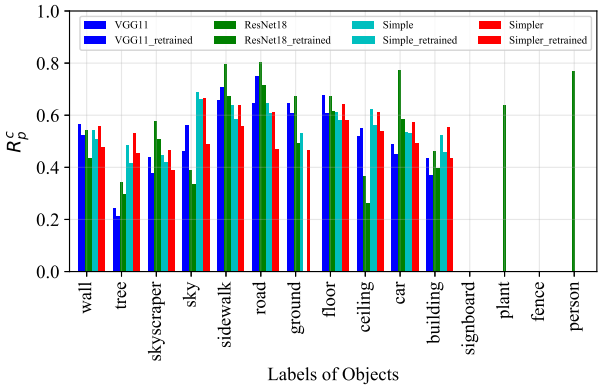
these patterns of objects reveal the interpretability of deep representations learned for a city recognition CNN, to some degree, it is still necessary to evaluate the interpretability in a quantitative manner.

3.4. Object-level Interpretability

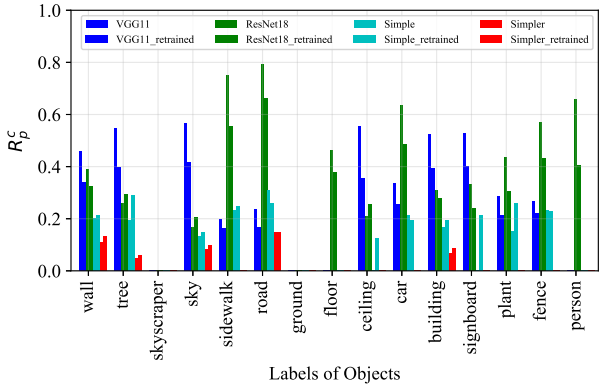
We address the second research question in this section by quantifying the object level information that are extracted using visualization method. The lack of classes information of objects appearing in city images from **Pittsburgh** and **Tokyo 24/7** datasets makes it difficult to quantify the interpretibility of the deep representations learned from a city recognition CNN. Therefore, we apply semantic segmentation models to obtain the objects classes information before evaluating interpretability. The semantic segmentation model used in our experiment is pre-trained on MIT ADE20K scene parsing dataset [13, 19, 20] and is built on ResNet50 [4]. The segmentation model is able to classify 150 different categories of objects, including all classes of objects appearing in city images.

To evaluate interpretability of deep representations quantitatively and avoid missing any possible information of objects in visual explanations, we calculate R_p^c for different objects and datasets (classes), as shown in Figure 5. The objects are selected by the criterion that the average number of pixels exceeds a certain threshold (set as 100).

Comparing the class-discriminate objects shown in Figure 5, dissimilar objects for different datasets are learned by city recognition CNNs. Skycraper and ground are the unique class-discriminate objects learned from **Pittsburgh**



(a) Histogram of R_p^c over class-discriminate objects appearing in **Pittsburgh**



(b) Histogram of R_p^c over class-discriminate objects appearing in **Tokyo 24/7**

Figure 5. Histograms of R_p^c regarding different architectures of CNNs, initializations and datasets. Different values of R_p^c of different objects are learned from different datasets. Some unique objects can only be learned from certain dataset.

dataset, while signboard and fence are the unique ones from **Tokyo 24/7**. The values of the ratios of pixels R_p^c indicate the selectivity of city recognition CNNs from specific dataset. The larger value of R_p^c is, the stronger the class-discriminate attributes. E.g., a uniform histogram over different objects means city recognition CNNs take any object in the image as class-discriminate, which is meaningless in this case. Different non-uniform distributions over objects from different classes reveal city recognition CNNs learn distinct combinations of class-discriminate objects from different datasets, which is interpretable for city recognition CNNs.

3.4.1 Do Different Models Learn Similar discriminate Objects?

Besides the histograms used in Figure 5, we also apply another quantitative method to investigate the influence of network architectures and initializations on the interpretabil-

ity of city recognition CNNs. Figure 6 shows some examples of weighted masks learned by different city recognition CNNs. The difference among the weighted masks learned by different city recognition models reflects the influence of network architectures.

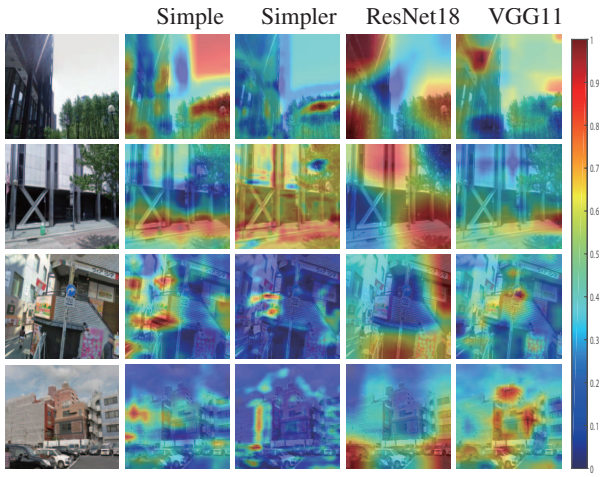


Figure 6. Different city recognition CNNs generate different weighted masks for the same image. The first two rows present two test images from **Pittsburgh** and the last two rows show the test images from **Tokyo 24/7**. From the second column to the fifth column, the weighted masks are shown for Simple, Simpler, ResNet18 and VGG11 city recognition CNNs, respectively. Note that a shallow net triggers on the sky or on disjoint regions in the image. The ResNet18 focuses on wider regions and VGG11 is more selective.

To quantify the divergence between different models, we calculate the average residual AR of each city image between any two models to investigate the consistency quantitatively:

$$AR_{m_1,m_2} = \frac{\left|mask_norm^c_{m_1} - mask_norm^c_{m_2}\right|}{H \times W}, \quad (2)$$

where H and W are the height and width of weighted masks, and m_1 and m_2 represent CNN models. The value of AR_{m_1,m_2} ranges from 0 to 1, where 0 means two models learn exactly same weighted mask for this image and 1 means totally different weighted masks have been learned by two models. Due to the considerable images, we calculate the average AR_{m_1,m_2} over all test images. All values of average AR_{m_1,m_2} between different network architectures and initialziations are listed in Table 2.

Comparing the values in Table 2, we can find that ARs between different architectures are larger than the ones between different initializations in general, which means network architectures affect the deep representations learned from city recognition CNNs greater than different training initializations. This is also consistent with the results from Figure 5.

Table 2. The average AR_{m_1,m_2} between different network architectures and initializations. The values of AR_{m_1,m_2} between different network architectures are all larger than the ones between different initializations.

Models (m_1-m_2)	Average AR_{m_1,m_2}
VGG11-ResNet18	0.4349
VGG11-Simple	0.4303
VGG11-Simpler	0.4502
ResNet18-Simple	0.4118
ResNet18-Simpler	0.4136
Simple-Simpler	0.3149
VGG11-VGG11_retrained	0.2679
ResNet18-ResNet18_retrained	0.2265
Simple-Simple_retrained	0.2411
Simpler-Simpler_retrained	0.2460

Besides calculating ARs between different city recognition models, we can also use R^c_p to get the consistent results. In Figure 5 (a) and (b), different CNNs architectures learn dissimilar histograms over class-discriminate objects, however, similar values of R^c_p over class-discriminate objects are learned due to different initializations. In Figure 5 (b), we can also find the values R^c_p of VGG11 and ResNet18 are larger than the ones of shallow networks over all class-discriminate objects, which also reflects that convolutional features learned by deep network architectures are more semantically interpretative than the shallow ones. Therefore, the influence of network architectures on the interpretability of CNN features is stronger than the one of different initializations.

4. Conclusion

In this work, we provided a framework to investigate the emergence of semantic objects as discriminate attributes in the ultimate layer of network. This is consistent with the way human understand city images. We applied our proposed framework to investigate the influence of network architectures and different initializations on the interpretability. We conclude that network architectures would affect the learned visual representations greater than different initializations.

References

[1] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5297–5307, 2016.

[2] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba. Network dissection: Quantifying interpretability of deep visual representations. *arXiv preprint arXiv:1704.05796*, 2017.

[3] R. C. Fong and A. Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *2017 IEEE international conference on computer vision (ICCV)*, pages 3449–3457. IEEE, 2017.

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[5] L.-J. Li, H. Su, L. Fei-Fei, and E. P. Xing. Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *Advances in neural information processing systems*, pages 1378–1386, 2010.

[6] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[7] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1717–1724, 2014.

[8] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free?-weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 685–694, 2015.

[9] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626. IEEE, 2017.

[10] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[11] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[12] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[13] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 418–434, 2018.

[14] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[15] J. Zhang, S. A. Bargal, Z. Lin, J. Brandt, X. Shen, and S. Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10):1084–1102, 2018.

[16] Q. Zhang, Y. Nian Wu, and S.-C. Zhu. Interpretable convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8827–8836, 2018.

[17] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object detectors emerge in deep scene cnns. *arXiv preprint arXiv:1412.6856*, 2014.

[18] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2929, 2016.

[19] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, page 4. IEEE, 2017.

[20] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, pages 1–20, 2016.