

# A Tour of Visualization Techniques for Computer Vision Datasets

Bilal Alsallakh, Pamela Bhattacharya, Vanessa Feng, Narine Kokhlikyan, Orion Reblitz-Richardson and David Yan

Meta

## Abstract

We survey a number of data visualization techniques for analyzing Computer Vision (CV) datasets. These techniques help us understand properties and latent patterns in such data, by applying dataset-level analysis. We present various examples of how such analysis helps predict the potential impact of the dataset properties on CV models and informs appropriate mitigation of their shortcomings. Finally, we explore avenues for further visualization techniques of different modalities of CV datasets as well as ones that are tailored to support specific CV tasks and analysis needs.

## 1 Introduction

The majority of work on understanding CV algorithms and explaining their behavior and results focuses on analyzing the internals of CV models and revealing the features they learn during training. On the other hand, less attention has been paid to analyzing CV datasets and their properties, despite the impact of these properties on the model behavior and on the learned features. Existing work tends to rely on quantitative analysis [58] of these datasets, with limited focus on exploratory visual analysis.

Torralba and Efros [50] assessed the quality of various CV datasets. Their comparative analysis illustrates different types of bias in these datasets which limit their generalizability and representativeness. Likewise, various studies examined label imbalance issues in various CV datasets and their impact on different tasks, focusing on solutions to mitigate the bias associated with them. Fabbrizzi et al. [15] provide an excellent overview of these studies. For example, Oksuz et al. [38] studied different types of imbalance in object detection stemming from the interaction between foreground and background objects, their sizes, and locations. Nevertheless, as noted by Wang et al. [54] there is a lack of reusable tools and techniques to systematically surface different types of bias in CV datasets. We also note that most techniques rely on quantitative analysis, and rarely leverage data visualization as a powerful means to analyze CV datasets.

We present visualization techniques that help analyze fundamental properties of CV datasets. The goal of this analysis is to improve our understanding of these properties and how they can potentially impact CV models and algorithms. Our contributions include:

- Providing an overview of generally applicable visualization techniques for CV datasets.
- Demonstrating how the insights these techniques provide help expose potential shortcomings of CV datasets and explain CV model behavior.

In Section 3 we explore further analysis opportunities that are not well supported by existing techniques.

## 2 A Tour of Visualization Techniques

This section surveys a number of visualization techniques that can be applied to CV datasets. For each technique, we demonstrate example insights in popular datasets, as well as possible implications of these insights.

### 2.1 Pixel-Level Component Analysis

Principal Component Analysis (PCA) is widely used as a nonparametric method for dimensionality reduction. When used for analysis and visualization purposes, oftentimes, the data points are projected along the top-2 or top-3 principal components (PCs) and visualized using 2D or 3D plots. Such plots are rather limited when PCA is applied in the flattened pixel space, which is usually very high-dimensional. Furthermore, a relatively small number of data points can be intelligibly visualized in 2D plots, especially when image icons are displayed.

Fortunately, it is possible to visualize the PCs of a flattened pixel space directly as visual entities, without projecting data points on them. This is because each PC is an eigenvector defined as a linear combination of the dimensions of the mean-centered pixel space. Each value in this vector corresponds to one color channel of a unique pixel and can be either positive or negative. A pictorial representation of these values can reveal the image features to which the respective PC is sensitive.

We propose visualizing each PC by means of two images, one consisting of the positive values of its eigenvector (with the negative ones treated as zero), and the other consisting of the inverted negative values (with the

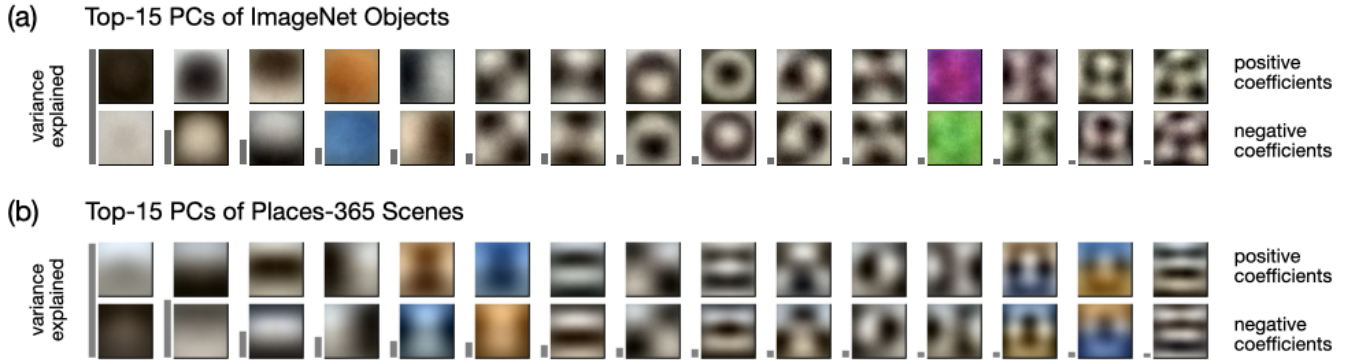


Figure 1: The top-15 principal components in two CV datasets ImageNet and Places365. The images were resized to  $40 \times 40$ . The analysis is applied to (a) the bounding boxes of ImageNet objects (b) to full scenes in Places365.

positive ones treated as zero). This decomposes the visual representation into two simpler parts, each showing which pixels tend to have high value at one end of the PC direction in the data space. This reveals subtle details in both images that are not always visible when visualizing the eigenvector values in a single image. For example, this reveals that the variance explained by the 4-th PC in Figure 1a is between orange-dominated vs. blue-dominated images. To the side of the two images, we depict a gray bar whose length represents the eigenvalue of each PC. Accordingly, the visualization of the pixel space hence consists of the two extremes of each PC, in addition to a bar chart of the eigenvalues.

We next present two different modes to apply PCA to image data based on the unit of analysis, and elaborate on the insights they provide in the dataset. The first mode operates on whole images while the second operates on small patches in these images.

### PCA on Whole Images

To apply PCA to a dataset whose items are images, these images must be of the same size and color depth. This allows treating the dataset as a high-dimensional table. Furthermore, PCA requires the number of data points  $n$  to be larger than the number of dimensions  $p$  in the table to allow consistent estimation of the subspace of maximal variance. Depending on the dataset and the goal of analysis, recommendations for the ratio between sample size and number of dimensions vary between 1 and 100, with higher ratios usually recommended for very high-dimensional datasets.

The above requirements are generally straightforward to meet for the purpose of analyzing and understanding vision datasets by applying suitable crop and resize operations. While downscaling the images can compromise fine-grained details, applying PCA to downscaled images is useful to analyze global factors of variations among them. Furthermore, with small images the computation overhead is significantly smaller than with large ones<sup>1</sup>. Accordingly, we recommend starting the analysis with

<sup>1</sup>The time complexity of standard PCA is  $O(p^3 + p^2 \cdot n)$ .

relatively small images, and validating the axes found by repeating the analysis on larger resolutions while avoiding overfitting.

In Figure 1a, the PCs were computed using the ImageNet validation set, which contains  $n = 50,000$  data points. We first crop each image to the bounding box of the target object and then resize the image further to  $40 \times 40$  pixels, resulting in  $k = 4800$  dimensions and hence a ratio of  $n/k = 10.4$ . As evident in Figure 1a, the top-5 PCs correspond to low-frequency features that together explain about 40% of the variance in the dataset.

- The 1st PC differentiates between dark and bright object images, explaining about 24% of the variance.
- The 2nd PC differentiates between dark objects on a bright background, and the opposite case.
- The 3rd PC is sensitive to whether the brightness is concentrated in the upper or bottom parts of the image.
- The 4th PC differentiates based on orange and sky-blue tints, two opposing colors in the RGB cube.
- The 5th PC is sensitive to whether the brightness is concentrated in the left or right parts of the image.

The remaining PCs correspond to increasingly higher-frequency features in the spatial dimensions. One exception is the 12th PC, which differentiates based on violet and green tints, which are also opposing colors in the RGB cube. Interestingly, this PC explains only 1.24% of the variance, while the orange-blue PC explains 3.3% of it. This suggests that perturbations along these dimensions might have varying impact on the learned models, as we demonstrate in Section 2.1. In Figure 1b, the same analysis is applied to the Places365 validation set. Notice how:

- Horizontal bands dominate the top PCs, such as the 3rd and 7th PCs, while vertical bands are less salient and are often mixed with the blue-orange dimension. This reflects the nature of scene-centric datasets, in contrast with the object-centric ImageNet.

- The green-violet axis is not among the top-15 components of Place-365, while the blue-orange dimension is correlated with four of them. This is because scene images are rarely dominated by violet color, which limits the variance explained by the green-violet axis.

### PCA on Image Patches

Instead of treating images as the unit of analysis, it is possible to apply PCA to image patches. This mode of analysis has been used in the literature to compare images [47], denoise them [11], and to extract features for recognition tasks [26]. Furthermore, it has been used to understand the statistics of natural images and to link them with certain mechanisms of human vision [13, 21, 23, 45].

Applying PCA to image patches eliminates the need for resizing the images. When used to understand the influence of the data on the model learned, the patch size can be selected to match how the model processes the images. For example, when convolutional neural networks are used, the size can be selected to match the filter size of the first convolutional layer.



Figure 2: The top-15 principal components in 3.1 million image patches of size  $11 \times 11 \times 3$ , sampled from ImageNet.

Figure 2 shows the PCA results computed for 3.1 million patches of size  $11 \times 11$ , randomly sampled from the ImageNet validation dataset at random locations spanning the entire image area. The size corresponds to the filter size of the first convolutional layer in AlexNet [30]. When trained to classify ImageNet, the color-blob filters in the first layer of AlexNet largely follow the two color dimensions we found among the top PCs.

### Independent Component Analysis (ICA)

PCA is designed to maximize the variance explained by the top PCs it computes. When applied to images, these PCs hence tend to involve all pixels in order to find directions that maximize this variance. Accordingly, the visual features that correspond to these directions are rather global, as evident in Figure 1. Moreover, these directions can be correlated.

ICA [24] offers an alternative component analysis that aims to maximize the *independence* between these components, instead of the variance they explain. When applied to images, the independent components (ICs) found can hence focus on localized features that are uncorrelated. Figure 3 shows examples of ICs computed for two datasets, VGG Face [41] and the CASIA Chinese Offline Handwritten Characters [33]. The ICs were computed using Fast ICA [22], applied to images of size  $40 \times 40$ . Notice how the IC components are localized and highly disentangled from each other.

Selected ICs of VGGFace



Selected ICs of CASIA's Handwritten Chinese Characters



Figure 3: Selected independent components in VGG Face and in CASIA’s Handwritten Chinese Characters. Notice how ICA can identify localized features.

When applied to image patches sampled from natural images, ICA was shown to learn features that resemble Gabor functions [23]. The gray-scale filters in AlexNet also resemble Gabor functions. This suggests that these filters aim to maximize the independence between their features. It is useful to reduce the dimensionality of the data space using PCA, before applying ICA, in order to discourage ICA from overfitting noise patterns.

### How useful is component analysis?

Exploring the top PCs in the pixel space is helpful to understand which image features are behind significant variations in the dataset, and accordingly predict their potential importance for the model. For example, it is evident in Figure 1a and in Figure 2 that the blue-orange direction explains more variance than the green-violet direction in the ImageNet pixel space. Accordingly, we suspect that ablating the blue color channel will likely have more impact on ImageNet classifiers than ablating the green channel. To verify this assumption, we ablate each of the three color channels in the input of ResNet-18 and report its top-1 accuracy on ImageNet in Table 1. We perform ablation either by replacing the color channel with the mean of the other two channels, or with the mean of all three channels (i.e. a basic gray image).

Table 1: Top-1 accuracy of ResNet-18 trained on ImageNet when different color channels are masked in the validation set. The baseline accuracy is 69.68%

Mask channel w.:	Mean of other channels	Gray image
Red	56.50%	63.70%
Green	62.12%	66.57%
Blue	60.56%	64.93%

As evident in Table 1, ablating the blue channel indeed has a higher impact on performance, compared with the green channel. Moreover, ablating the red channel has the highest impact, since it is strongly involved both in the violet and in the orange PC extremes. These results generalize to a variety of models as well as to CIFAR-10. This illustrates the value of understanding the major dimensions of variation in the pixel space.

## 2.2 Spatial Analysis

CV datasets usually contain annotations of various objects in their images, typically in the form of bounding boxes or segmentation masks. Visualizing these boxes and masks for an entire dataset is helpful to understand how the corresponding objects are spatially distributed. Figure 4 depicts the spatial distribution of three object



Figure 4: The spatial distribution of three object categories in the MS COCO dataset, obtained by aggregating their masks.

categories in the MS COCO dataset [32]. Each plot aggregates the mask images of dataset samples that contain the corresponding object. The aggregation is performed by resizing the mask images to  $640 \times 640$  pixels, summing up the object pixels in these mask into an aggregate image, and normalizing the resulting image to the range  $[0, 255]$ . It is also possible to analyze further spatial relationships in CV datasets, such as co-occurrence and adjacency relationships between various categories in semantic segmentation [54].

### How useful is this analysis?

Analyzing the spatial distribution of object categories helps uncover potential shortcomings of a CV dataset. Figure 5 illustrates how the spatial distribution of *caravan* category in CityScapes [9] varies significantly between the training and validation sets. Likewise, Gauhen et al [17] show differences in how person images are spatially distributed in seven popular datasets, which helps evaluate transferability between them. The analysis can also help assessing whether popular data augmentation methods are suited to mitigate any skewness in the spatial distribution. For example, the commonly used horizontal flipping helps create training sample that contain a computer mouse both on the left side and on the right side of the image. However, as evident in Figure 4, the samples remain predominantly in the lower part of the image. Moreover, horizontal flipping might negatively impact images with orientation-sensitive content such as written text [50].

To appreciate the impact of spatial distribution, we present an example of a traffic light detector that was shown to learn and overfit the spatial distribution of traffic lights in the training set [2]. This distribution is depicted in Figure 6a. Figure 6b depicts the detection score the the model computes a traffic light stimulus placed in a blank image. The score is evidently higher if the stimulus is present in a position that has a high



Figure 5: The spatial distribution of the *caravan* category in two splits of the CityScapes dataset.

density of training samples. Such overfitting is possible as convolutional networks are able to encode position information [25], with padding being a major source of this information [27]. This insight helps inform the choices of data augmentation and of the detection architecture.

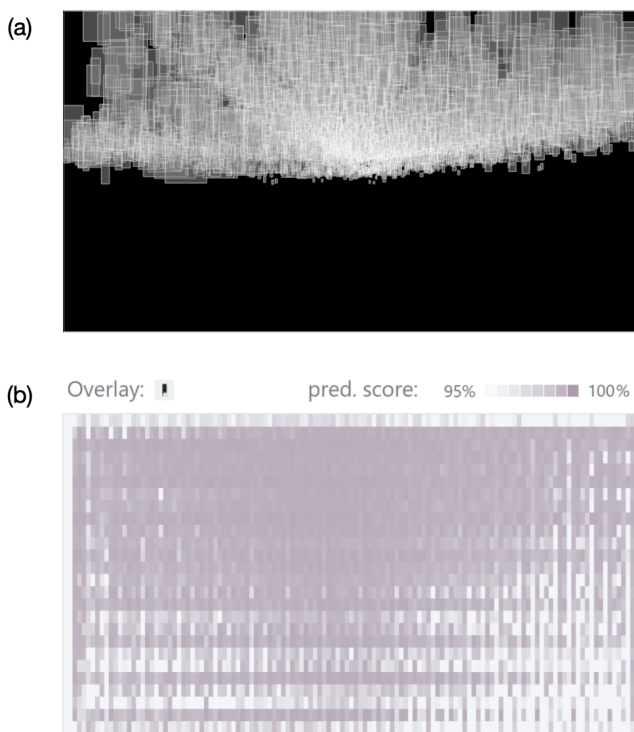


Figure 6: The impact of spatial distribution on RCNN-based small object detection. (a) The distribution of traffic light bounding boxes in the BSTLD dataset [4]. (b) The prediction scores for a traffic light stimulus when placed at various locations in a blank image, computed via Faster RCNN. (Figure adapted from [2]).

## 2.3 Average Image Analysis

Averaging a collection of images offers a useful visual summary that helps detect various issues in CV datasets [43]. Average images can also be used to compare subsets of images of the same nature and semantics, e.g., to trace how portrait photos change over time [19] or to compare hand-drawings of the same object across

different cultures [34]. Interactive clustering exploration is often helpful to select subsets that represent representative manifestations or interesting outliers [34, 59]. Likewise, when computing average images of selected objects, it is helpful to group the individual images based on the object pose, as illustrated in Figure 7.



Figure 7: Average images of the car category in PASCAL VOC [14], computed for different poses (adapted from <https://cs.cmu.edu/~tmalisie/pascal>).

### How useful is this analysis?

Computing average images per class [51] can reveal visual cues in classification datasets that classifiers can use as “shortcuts” instead of learning robust semantic features. Analyzing these cues helps in designing dataset augmentations that prevent models from using them. In addition, the mean image of ImageNet was used to investigate the source of weight banding effects [42] in models that use global average pooling [1]. The authors demonstrate further applications of image averaging to analyze and debug the internals of convolutional networks.

## 2.4 Metadata and Content-based Analysis

Google’s Know Your Data<sup>2</sup> offers means to understand datasets with the goal of improving data quality and mitigating bias. Users can analyze metadata about the images such as their aspect ratios and resolution, extracted signals such as image quality and sharpness, or various high-level features such as the presence of faces. Likewise, Google’s Facets<sup>3</sup> offer a zoomable interface to explore large datasets using thumbnails, with several possibilities to split and arrange the data points into scatter plots, based on their metadata.

It is often possible to analyze the geographic distribution of vision datasets based on available location information in their metadata. Likewise, exploring datasets based on their temporal information is useful to assess their appropriateness for the target task. Such analysis can account for several aspects of the time dimension such as the time of day, time of year, and overall

<sup>2</sup><http://knowyourdata.withgoogle.com>

<sup>3</sup><http://pair-code.github.io/facets/>

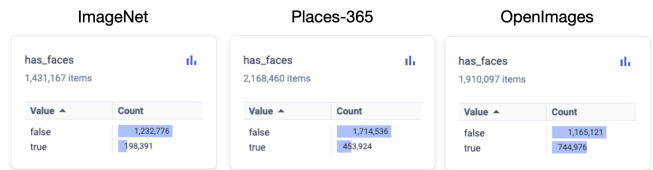


Figure 8: Examples of metadata summaries in Google’s Know Your Data, extracted from three popular datasets. Notice how OpenImages V4 [31] has the highest proportion of images that contain faces.

recency of the samples. Metadata can sometimes be recovered using specialized techniques, e.g. for dating images [29, 40] or approximating their geolocation [12, 55].

### How useful is this analysis?

Geolocation analysis helps assess the diversity of a dataset and explore different manifestations of targeted classes and features, in order to guide the curation and labeling of representative datasets [10, 48, 54]. Likewise, exploring datasets based on their temporal information is useful to assess their appropriateness for the target task. The REVISE tool [54] demonstrates a variety of examples of how the above ideas can be applied to analyze various types of biases in vision datasets along three dimensions: objects, attributes, and geography. Figure 9 shows an example of such analysis.

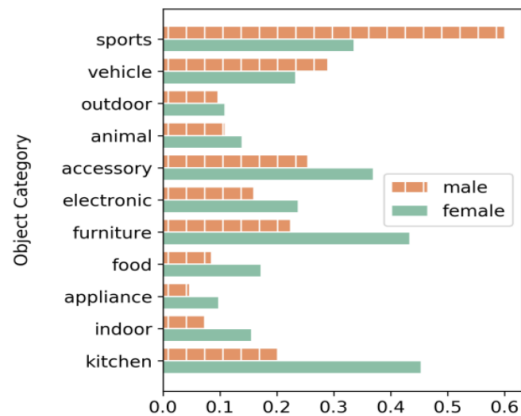


Figure 9: Content-based analysis of MS COCO using the REVISE tool [54] focusing on intersectional gender bias in scenes containing persons alongside other objects.

## 2.5 Analysis using Trained Models

The techniques presented so far can be applied directly to datasets, without requiring a model trained on them. When available, such models offer a lens into the dataset that can reveal potential biases or deficiencies.

A variety of analysis methods have been developed to understand the features learned by CV models, such as feature saliency in a given input [3, 46, 49], input

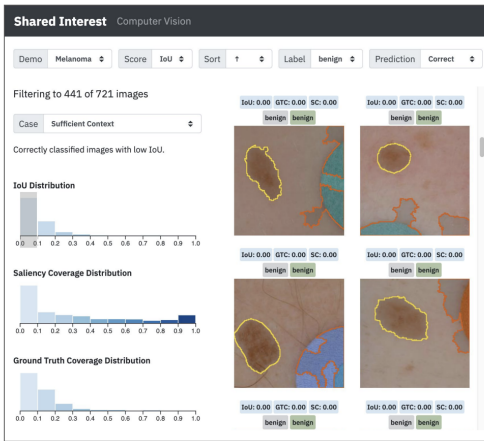


Figure 10: The Shared Interest UI [6] showing how well human annotations in the ISIC dataset [8] match salient detection features.

optimization [37, 39, 57], and concept-based interpretation [18, 28]. While these techniques primarily focus on analyzing model behavior, they were shown to be useful for identifying dataset issues. For example dumbbell images in ImageNet often contain arms [36], leading to pictures of arms being classified as `dumbbell`. Likewise, specialized saliency techniques developed for video recognition models can show biases in video datasets [16]. Shared Interest [6] and Activation Atlas [7] enable systematic analysis of these issues on the dataset level.

Furthermore, various techniques use prediction results as a proxy to analyze dataset properties. For example, analyzing these results during training can reveal inherent sample difficulty [20, 35]. Likewise, analyzing error patterns in classification datasets using confusion matrices helps reveal latent hierarchical structures that govern their classes [5].

Finally, trained models can be used to compute embeddings that are suited to project the datasets into a 2D plot based on semantic features. Such a plot helps identify potential patterns and outliers in the dataset, especially when aided with interactive exploration.

### How useful is this analysis?

Difficulty analysis reveals the dichotomous nature of various CV datasets [35]. For example, the majority of samples in ImageNet can either be correctly classified after a few training epochs, or continue to be misclassified throughout the training process. Such insights are very helpful to understand the behavior of different learning paradigms and architectures, and how they are impacted by inherent issues in the training data. Likewise, comparing human annotations with feature attribution results (Figure 10) was shown useful in revealing ambiguities and shortcoming of these annotations [6]. Finally, visualizing latent hierarchical structures in classification datasets helps reveal the properties of different groups of classes, such as their reliance on color or orientation.

## 3 Discussion

Visual analysis offers distinctive opportunities to improve our understanding of CV datasets. It can provide insights into their properties that are not always obvious or well understood. The techniques presented focus almost exclusively on image datasets. Here we discuss opportunities for further research in the area.

### Video, Point Cloud, 3D, and Keypoint Datasets

Generally, there is a lack of dataset-level visualizations that are applicable to modalities in CV other than images. Nevertheless, many of the techniques we presented can be extended to support these modalities. Hyvriin demonstrated how ICA helps understand spatiotemporal features in image sequences [23]. Likewise, t-SNE plots [53] are often used to visualize video embeddings in order to analyze the separability and overlaps between different classes [44, 52]. Dedicated visualization techniques can be further designed to reveal structures and patterns in non-image CV datasets based on the rich nature of their data modalities.

**Task-driven Visual Analysis** Besides supporting specific data modalities, the design of new visualization techniques should also take into account the target CV task. For example, image classification datasets are different in nature than image segmentation datasets. The corresponding visualizations can hence assign visual primacy in their designs to different facets, such as the class hierarchy or the spatial distribution.

### Visually Linking Model Results with Dataset Properties

Such an analysis can be useful to explain model behavior and find a suitable mitigation for observed shortcomings. Figure 6 demonstrates how the spatial distribution of the training set dictates object predictability at different locations.

### Comparing Datasets and Subsets Thereof

Such comparisons have various applications, e.g., to assessing domain suitability in transfer learning and analyzing distribution shift [56]. We presented two simple examples of comparing specific features between two image datasets (Figure 1) or between two splits of CityScapes (Figure 5). Further techniques can be designed to visualize similarities and differences between datasets e.g. by exposing the internals of similarity metrics.

## 4 Conclusion

Visualization is a powerful means for understanding CV datasets. We presented a number of visualization techniques that enable analyzing various pieces of information in these datasets. These techniques support dataset-level analysis to reveal latent patterns in the data and help us understand its properties. For each technique, we demonstrated example insights it can enable in popular datasets. We also provided application scenarios of how these insights help us understand the impact of dataset properties on CV models. We finally explored avenues for further research on visual dataset understanding, towards robust and transparent CV.

## References

- [1] B. Alsallakh, N. Kohlikyan, V. Miglani, S. Mutteparwar, E. Wang, S. Zhang, D. Adkins, and O. Reblitz-Richardson. Debugging the internals of convolutional networks. In *eXplainable AI approaches for debugging and diagnosis*, 2021. 5
- [2] Bilal Alsallakh, LI Nanxiang, Lincan Zou, Axel Wendt, and Liu Ren. Visual analytics exposure of image object detector weakness, 2021. US Patent 10,997,467. 4
- [3] 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), 2015. 5
- [4] Karsten Behrendt, Libor Novak, and Rami Botros. A deep learning approach to traffic lights: Detection, tracking, and classification. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1370–1377. IEEE, 2017. 4
- [5] Alsallakh Bilal, Amin Jourabloo, Mao Ye, Xiaoming Liu, and Liu Ren. Do convolutional neural networks learn class hierarchy? *IEEE transactions on visualization and computer graphics*, 24(1):152–162, 2017. 6
- [6] Angie Boggust, Benjamin Hoover, Arvind Satyanarayan, and Hendrik Strobelt. Shared interest: Measuring human-ai alignment to identify recurring patterns in model behavior. In *CHI Conference on Human Factors in Computing Systems*, pages 1–17, 2022. 6
- [7] Shan Carter, Zan Armstrong, Ludwig Schubert, Ian Johnson, and Chris Olah. Activation atlas. *Distill*, 4(3):e15, 2019. 6
- [8] Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019. 6
- [9] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, et al. The CityScapes dataset for semantic urban scene understanding. In *IEEE conference on computer vision and pattern recognition (CVPR)*, pages 3213–3223, 2016. 4
- [10] Terrance De Vries, Ishan Misra, Changan Wang, and Laurens Van der Maaten. Does object recognition work for everyone? In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 52–59, 2019. 5
- [11] Charles-Alban Deledalle, Joseph Salmon, Arnak S Dalalyan, et al. Image denoising with patch based pca: local versus global. In *BMVC*, volume 81, pages 425–455, 2011. 3
- [12] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. What makes paris look like paris? *ACM Transactions on Graphics*, 31(4), 2012. 5
- [13] Jan Eichhorn, Fabian Sinz, and Matthias Bethge. Natural image coding in v1: how much use is orientation selectivity? *PLoS computational biology*, 5(4):e1000336, 2009. 3
- [14] M. Everingham, A. Zisserman, C. K. I. Williams, and L. Van Gool. The PASCAL Visual Object Classes Challenge 2006 (VOC2006) Results. <http://www.pascal-network.org/challenges/VOC/voc2006/results.pdf>. 5
- [15] Simone Fabbrizzi, Symeon Papadopoulos, Eirini Ntoutsis, and Ioannis Kompatsiaris. A survey on bias in visual datasets. *arXiv preprint arXiv:2107.07919*, 2021. 1
- [16] Christoph Feichtenhofer, Axel Pinz, Richard P Wildes, and Andrew Zisserman. Deep insights into convolutional networks for video recognition. *International Journal of Computer Vision*, 128(2):420–437, 2020. 6
- [17] Kent Gauen, Ryan Dailey, John Laiman, Yuxiang Zi, Nirmal Asokan, et al. Comparison of visual datasets for machine learning. In *IEEE International Conference on Information Reuse and Integration*, pages 346–355, 2017. 4
- [18] Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. Towards automatic concept-based explanations. *Advances in Neural Information Processing Systems*, 32:9277–9286, 2019. 6
- [19] Shiry Ginosar, Kate Rakelly, Sarah Sachs, Brian Yin, and Alexei A Efros. A century of portraits: A visual historical record of american high school yearbooks. In *IEEE International Conference on Computer Vision Workshops*, pages 1–7, 2015. 4
- [20] Guy Hacoen, Leshem Choshen, and Daphna Weinshall. Let’s agree to agree: Neural networks share classification order on real datasets. In *International Conference on Machine Learning*, pages 3950–3960. PMLR, 2020. 6
- [21] Peter JB Hancock, Roland J Baddeley, and Leslie S Smith. The principal components of natural images. *Network: computation in neural systems*, 3(1):61, 1992. 3
- [22] Aapo Hyvarinen. Fast ica for noisy data using gaussian moments. In *1999 IEEE international symposium on circuits and systems (ISCAS)*, volume 5, pages 57–61. IEEE, 1999. 3
- [23] Aapo Hyvriinen, Jarmo Hurri, and Patrick O. Hoyer. *Natural Image Statistics: A Probabilistic Approach to Early Computational Vision*. Springer, 1st edition, 2009. 3, 6
- [24] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4):411–430, 2000. 3
- [25] Md Amirul Islam, Sen Jia, and Neil DB Bruce. How much position information do convolutional neural networks encode? In *International Conference on Learning Representations (ICLR)*, 2020. 4
- [26] Tai-Xiang Jiang, Ting-Zhu Huang, Xi-Le Zhao, and Tian-Hui Ma. Patch-based principal component analysis for face recognition. *Computational intelligence and neuroscience*, 2017, 2017. 3
- [27] Osman Semih Kayhan and Jan C van Gemert. On translation invariance in CNNs: Convolutional layers can exploit absolute spatial location. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 4
- [28] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *International conference on machine learning*, pages 2668–2677. PMLR, 2018. 6
- [29] Gunhee Kim and Eric P Xing. Time-sensitive web image ranking and retrieval via dynamic multi-task regression. In *3ACM international conference on Web search and data mining*, pages 163–172, 2013. 5

- [30] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1097–1105, 2012. 3
- [31] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, et al. The open images dataset v4. *International Journal of Computer Vision*, 128(7):1956–1981, 2020. 5
- [32] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 4
- [33] Cheng-Lin Liu, Fei Yin, Da-Han Wang, and Qiu-Feng Wang. Casia online and offline chinese handwriting databases. In *International Conference on Document Analysis and Recognition*, pages 37–41. IEEE, 2011. 3
- [34] Mauro Martino, Hendrik Strobelt, Owen Corne, and Evan Phibbs. Forma fluens-abstraction, simultaneity and symbolization in drawings. <http://formafluens.io/>, 2017. 5
- [35] Kristof Meding, Luca M. Schulze Buschoff, Robert Geirhos, and Felix A. Wichmann. Trivial or impossible — dichotomous data difficulty masks model differences (on ImageNet and beyond). In *International Conference on Learning Representations*, 2022. 6
- [36] Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks. *Google Research Blog*, 2015. <https://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html>. 6
- [37] Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. 6
- [38] Kemal Oksuz, Baris Can Cam, Sinan Kalkan, and Emre Akbas. Imbalance problems in object detection: A review. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3388–3415, 2020. 1
- [39] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2(11):e7, 2017. 6
- [40] Frank Palermo, James Hays, and Alexei A. Efros. Dating historical color images. In *European Conference on Computer Vision*, pages 499–512. Springer Berlin Heidelberg, 2012. 5
- [41] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *British Machine Vision Conference*, 2015. 3
- [42] Michael Petrov, Chelsea Voss, Ludwig Schubert, Nick Cammarata, Gabriel Goh, and Chris Olah. Weight banding. *Distill*, 6(4):e00024–009, 2021. 5
- [43] J Ponce, T L Berg, M Everingham, D A Forsyth, M Hebert, S Lazebnik, M Marszalek, C Schmid, B C Russell, et al. Dataset issues in object recognition. In *Toward category-level object recognition*, pages 29–48. Springer, 2006. 4
- [44] Soroosh Poorgholi, Osman Semih Kayhan, and Jan C van Gemert. t-eva: Time-efficient t-sne video annotation. In *International Conference on Pattern Recognition*, pages 153–169. Springer, 2021. 6
- [45] Daniel L Ruderman, Thomas W Cronin, and Chuan-Chin Chiao. Statistics of cone responses to natural images: implications for visual coding. *JOSA A*, 15(8):2036–2045, 1998. 3
- [46] 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 *IEEE international conference on computer vision (ICCV)*, pages 618–626, 2017. 5
- [47] Honghao Shan and Garrison W Cottrell. Looking around the backyard helps to recognize faces and digits. In *IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1–8. IEEE, 2008. 3
- [48] S Shankar, Y Halpern, E Breck, J Atwood, J Wilson, and D Sculley. No classification without representation: Assessing geodiversity issues in open data sets for the developing world. *arXiv preprint arXiv:1711.08536*, 2017. 5
- [49] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Gradients of counterfactuals. *arXiv preprint arXiv:1611.02639*, 2016. 5
- [50] Antonio Torralba and Alexei A. Efros. Unbiased look at dataset bias. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1521–1528, 2011. 1, 4
- [51] Antonio Torralba and Aude Oliva. Statistics of natural image categories. *Network: computation in neural systems*, 14(3):391, 2003. 5
- [52] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *IEEE international conference on computer vision (ICCV)*, pages 4489–4497, 2015. 6
- [53] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11), 2008. 6
- [54] Angelina Wang, Arvind Narayanan, and Olga Russakovsky. REVISE: A tool for measuring and mitigating bias in visual datasets. In *European Conference on Computer Vision (ECCV)*, 2020. 1, 4, 5
- [55] Tobias Weyand, Ilya Kostrikov, and James Philbin. Planet-photo geolocation with convolutional neural networks. In *European Conference on Computer Vision*, pages 37–55. Springer, 2016. 5
- [56] Olivia Wiles, Sven Gowal, Florian Stimberg, Sylvestre Alvisé-Rebuffi, Ira Ktena, and Taylan Cemgil. A fine-grained analysis on distribution shift. *arXiv preprint arXiv:2110.11328*, 2021. 6
- [57] Jason Yosinski, Jeff Clune, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. In *ICML Workshop on Deep Learning*, 2015. 6
- [58] Oliver Zendel, Katrin Honauer, Markus Murschitz, Martin Humenberger, and Gustavo Fernandez Dominguez. Analyzing computer vision data-the good, the bad and the ugly. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1980–1990, 2017. 1
- [59] Jun-Yan Zhu, Yong Jae Lee, and Alexei A Efros. AverageExplorer: Interactive exploration and alignment of visual data collections. *ACM Transactions on Graphics (TOG)*, 33(4):1–11, 2014. 5