Visualizations of Deep Neural Networks in Computer Vision: A Survey

Christin Seifert, Aisha Aamir, Aparna Balagopalan, Dhruv Jain, Abhinav Sharma, Sebastian Grottel, and Stefan Gumhold

Abstract In recent years, Deep Neural Networks (DNNs) have been shown to outperform the state-of-the-art in multiple areas, such as visual object recognition, genomics and speech recognition. Due to the distributed encodings of information, DNNs are hard to understand and interpret. To this end, visualizations have been used to understand how deep architecture work in general, what different layers of the network encode, what the limitations of the trained model was and how to interactively collect user feedback. In this chapter, we provide a survey of visualizations of DNNs in the field of computer vision. We define a classification scheme describing visualization goals and methods as well as the application areas. This survey gives an overview of what can be learned from visualizing DNNs and which visualization methods were used to gain which insights. We found that most papers use Pixel Displays to show neuron activations. However, recently more sophisticated visualizations like interactive node-link diagrams were proposed. The presented overview can serve as a guideline when applying visualizations while designing DNNs.

Network Architecture Type Abbreviations

CDBN	Convolutional Deep Belief Networks
CNN	Convolutional Neural Networks
DBN	Deep Belief Networks
DCNN	Deep Convolution Neural Network
DNNs	Deep Neural Networks
MCDNN	Multicolumn Deep Neural Networks

C. Seifert (✉) • A. Aamir • A. Balagopalan • D. Jain • A. Sharma • S. Grottel • S. Gumhold
Technische Universität Dresden, Dresden, Germany
e-mail: Christin.42.Seifert@gmail.com; aishaaamir7@gmail.com; aparna.balagopalan@gmail.com; dhruvjain.1027@gmail.com; abhinav0301@gmail.com; sebastian.grottel@tu-dresden.de; stefan.gumhold@tu-dresden.de

T. Cerquitelli et al. (eds.), *Transparent Data Mining for Big and Small Data*,
Studies in Big Data 32, DOI 10.1007/978-3-319-54024-5_6

Dataset Name Abbreviations

CASIA-HWDB	Institute of Automation of Chinese Academy of Sciences-Hand Writing Databases
DTD	Describable Textures Dataset
FLIC	Frames Labeled In Cinema
FMD	Flickr Material Database
GTSRB	German Traffic Sign Recognition Benchmark
ISLVRC	ImageNet Large Scale Visual Recognition Challenge
LFW	Labeled Faces in the Wild
LSP	Leeds Sports Pose
MNIST	Mixed National Institute of Standards and Technology
VOC	Visual Object Classes
WAF	We Are Family
YTF	YouTube Faces

Other Abbreviations

CVPR	Computer Vision and Pattern Recognition
ICCV	International Conference on Computer Vision
IEEE	Institute of Electrical and Electronics Engineers
NIPS	Neural Information Processing Systems
t-SNE	Stochastic Neighbor Embedding

1 Introduction

Artificial Neural Networks for learning mathematical functions have been introduced in 1943 [48]. Despite being theoretically able to approximate any function [7], their popularity decreased in the 1970s because their computationally expensive training was not feasible with available computing resources [49]. With the increase in computing power in recent years, neural networks again became subject of research as Deep Neural Networks (DNNs). DNNs, artificial neural networks with multiple layers combining supervised and unsupervised training, have since been shown to outperform the state-of-the-art in multiple areas, such as visual object recognition, genomics and speech recognition [36]. Despite their empirically superior performance, DNN models have one disadvantage: their trained models are not easily understandable, because information is encoded in a distributed manner.

However, understanding and trust have been identified as desirable property of data mining models [65]. In most scenarios, experts can assess model performance on data sets, including gold standard data sets, but have little insights on how and

why a specific model works [81]. The missing understandability is one of the reasons why less powerful, but easy to communicate classification models such as decision trees are in some applications preferred to very powerful classification models, like Support Vector Machines and Artificial Neural Networks [33]. Visualization has been shown to support understandability for various data mining models, e.g. for Naive Bayes [1] and Decision Forests [66].

In this chapter, we review literature on visualization of DNNs in the computer vision domain. Although DNNs have many application areas, including automatic translation and text generation, computer vision tasks are the earliest applications [35]. Computer vision applications also provide the most visualization possibilities due to their easy-to-visualize input data, i.e., images. In the review, we identify questions authors ask about neural networks that should be answered by a visualization (*visualization goal*) and which *visualization methods* they apply therefore. We also characterize the application domain by the computer vision *task* the network is trained for, the *type* of network architecture and the *data sets* used for training and visualization. Note that we only consider visualizations which are automatically generated. We do not cover manually generated illustrations (like the network architecture illustration in [35]). Concretely, our research questions are:

RQ-1 Which insights can be gained about DNN models by means of visualization?
RQ-2 Which visualization methods are appropriate for which kind of insights?

To collect the literature we pursued the following steps: since deep architectures became prominent only a few years ago, we restricted our search starting from the year 2010. We searched the main conferences, journals and workshops in the area of computer vision, machine learning and visualization, such as: IEEE International Conference on Computer Vision (ICCV), IEEE Conferences on Computer Vision and Pattern Recognition (CVPR), IEEE Visualization Conference (VIS), Advances in Neural Information Processing Systems (NIPS). Additionally, we used keyword-based search in academic search engines, using the following phrases (and combinations): “deep neural networks”, “dnn”, “visualization”, “visual analysis”, “visual representation”, “feature visualization”.

This chapter is organized as follows: the next section introduces the classification scheme and describes the categories we applied to the collected papers. Section 3 reviews the literature according to the introduced categories. We discuss the findings with respect to the introduced research questions in Sect. 4, and conclude the work in Sect. 5.

2 Classification Scheme

In this chapter we present the classification scheme used to structure the literature: we first introduce a general view, and then provide detailed descriptions of the categories and their values. An overview of the classification scheme is shown in Fig. 1.

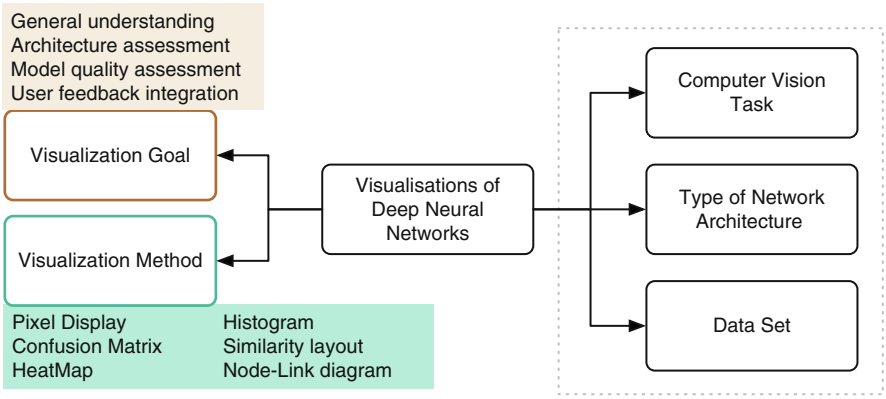

Fig. 1 Classification Scheme for Visualizations of Deep Neural Networks. The *dotted border* subsumes the categories characterizing the application area

First, we need to identify the purpose the visualization was developed for. We call this category **visualization goal**. Possible values are for instance *general understanding* and *model quality assessment*. Then, we identified the **visualization methods** used to achieve the above mentioned goals. Such methods can potentially cover the whole visualization space [51], but literature review shows that only a very small subset has been used so far in the context of DNNs, including *heat maps* and visualizations of *confusion matrices*. Additionally, we introduced three categories to describe the application domain. These categories are the *computer vision task*, the *architecture type* of the network and the *data set*s the neural network was trained on, which is also used for the visualization.

Note, that the categorization is not distinct. This means that one paper can be assigned multiple values in one category. For instance, a paper can use multiple visualization methods (CNNVis uses a combination of node-link diagrams, matrix displays and heatmaps [44]) on multiple data sets.

Related to the proposed classification scheme is the taxonomy of Grün et al. for visualizing learned features in convolutional neural networks [25]. The authors categorize the visualization methods into *input modification*, *de-convolutional*, and *input reconstruction* methods. In input modification methods, the output of the network and intermediate layers is measured while the input is modified. De-Convolutional methods adapt a reverse strategy to calculate the influence of a neuron's activation from lower layers. This strategy demonstrates which pixels are responsible for the activation of neurons in each layer of the network. Input reconstruction methods try to assess the importance of features by reconstructing input images. These input images can either be real or artificial images, that either maximize or lead to an output invariance of a unit of interest. This categorization is restricted to feature visualizations and therefore is narrower as the proposed scheme. For instance, it does not cover the general application domain, and is restricted to specific types of visualizations, because it categorizes the calculation methods used for Pixel Displays and heatmaps.

2.1 Visualization Goals

This category describes the various goals of the authors visualizing DNNs. We identified the following four main goals:

- *General Understanding:* This category encompasses questions about general behavior of the neural network, either during training, on the evaluation data set or on unseen images. Authors want to find out what different network layers are learning or have learned, on a rather general level.
- *Architecture Assessment:* Work in this category tries to identify how the network architecture influences performance in detail. Compared to the first category the analyses are on a more fine-grained level, e.g. assessing which layers of the architecture represent which features (e.g., color, texture), and which feature combinations are the basis for the final decision.
- *Model Quality Assessment:* In this category authors have focused their research goal in determining how the number of layers and role played by each layer can affect the visualization process.
- *User Feedback Integration:* This category comprises work in which visualization is the means to integrate user feedback into the machine learning model. Examples for such feedback integration are user-based selection of training data [58] or the interactive refinement of hypotheses [21].

2.2 Visualization Methods

Only a few visualization methods [51] have been applied to DNNs. We briefly describe them in the following:

- *Histogram:* A histogram is a very basic visualization showing the distribution of univariate data as a bar chart.
- *Pixel Displays:* The basic idea is that each pixel represents a data point. In the context of DNN, the (color) value for each pixel is based on network activation, reconstructions, or similar and yield 2-dimensional rectangular images. In most cases the pixels next to each other in the display space are also next to each other in the semantic space (e.g., nearby pixels of the original image). This nearness criterion is defined on the difference from Dense Pixel Displays [32]. We further distinguish whether the displayed values originate from a single image, from a set of images (i.e., a batch), or only from a part of the image.
- *Heat Maps:* Heat maps are a special case of Pixel Displays, where the value for each pixel represents an accumulated quantity of some kind and is encoded using a specific coloring scheme [72]. Heat maps are often transparently overlaid over the original data.
- *Similarity Layout:* In similarity-based layouts the relative positions of data objects in the low-dimensional display space is based on their pair-wise similar-

ity. Similar objects should be placed nearby in the visualization space, dissimilar objects farther apart. In the context of images as objects, suitable similarity measures between images have to be defined [52].
- *Confusion Matrix Visualization:* This technique combines the idea of heatmaps and matrix displays. The classifier confusion matrix (showing the relation between true and predicted classes) is colored according to the value in each cell. The diagonal of the matrix indicates correct classification and all the values other than the diagonal are errors that need to be inspected. Confusion matrix visualizations have been applied to clustering and classification problems in other domains [69].
- *Node-Link Diagrams* are visualizations of (un-)directed graphs [10], in which nodes represents objects and links represent relations between objects.

2.3 Computer Vision Tasks

In the surveyed papers different computer vision tasks were solved by DNNs. These are the following:

- *Classification:* The task is to categorize image pixels into one or more classes.
- *Tracking:* Object tracking is the task of locating moving objects over time.
- *Recognition:* Object recognition is the task of identifying objects in an input image by determining their position and label.
- *Detection:* Given an object and an input image the task in object detection is to localize this object in the image, if it exists.
- *Representation Learning:* This task refers to learning features suitable for object recognition, tracking etc. Examples of such features are points, lines, edges, textures, or geometric shapes.

2.4 Network Architectures

We identified six different types of network architectures in the context of visualization. These types are not mutually exclusive, since all types belong to DNNs, but some architectures are more specific, either w.r.t. the types of layers, the type of connections between the layers or the learning algorithm used.

- *DNN:* Deep Neural Networks are the general type of feed-forward networks with multiple hidden layers.
- *CNN:* Convolutional Neural Networks are a type of feed-forward network specifically designed to mimic the human visual cortex [22]. The architecture consists of multiple layers of smaller neuron collections processing portions of the input image (convolutional layers) generating low-level feature maps. Due to their specific architecture CNNs have much fewer connections and parameters compared to standard DNNs, and thus are easier to train.

- *DCNN:* The Deep Convolution Neural Network is a CNN with a special eight-layer architecture [35]. The first five layers are convolutional layers and the last three layers are fully connected.
- *DBN:* Deep Belief Networks can be seen as a composition of Restricted Boltzmann Machines (RBMs) and are characterized by a specific training algorithm [27]. The top two layers of the network have undirected connections, whereas the lower layers have directed connection with the upper layers.
- *CDBN:* Convolutional Deep Belief Networks are similar to DBNs, containing Convolutional RBMs stacked on one another [38]. Training is performed similarly to DBNs using a greedy layer-wise learning procedure i.e. the weights of trained layers are fixed and considered as input for the next layer.
- *MCDNN:* The Multicolumn Deep Neural Networks is basically a combination of several DNN stacked in column form [6]. The input is processed by all DNNs and their output aggregated to the final output of the DNN.

In the next section we will apply the presented classification scheme (cf. Fig. 1) to the selected papers and provide some statistics on the goals, methods, and application domains. Additionally, we categorize the papers according to the taxonomy of Grün [25] (input modification methods, de-convolutional methods and input reconstruction) if this taxonomy is applicable.

3 Visualizations of Deep Neural Networks

Table 1 provides an overview of all papers included in this survey and their categorization. The table is sorted first by publication year and then by author name. In the following, the collected papers are investigated in detail, whereas the subsections correspond to the categories derived in the previous section.

3.1 Visualization Goals

Table 2 provides an overview of the papers in this category. The most prominent goal is architecture assessment (16 papers). Model quality assessment was covered in 8 and general understanding in 7 papers respectively, while only 3 authors approach interactive integration of user feedback.

Authors who have contributed work on visualizing DNNs with the goal **general understanding** have focused on gaining basic knowledge of how the network performs its task. They aimed to understand what each network layer is doing in general. Most of the work in this category conclude that lower layers of the networks contains representations of simple features like edges and lines, whereas deeper layers tend to be more class-specific and learn complex image features [41, 47, 61]. Some authors developed tools to get a better understanding of learning capabilities

Table 1 Overview of all reviewed papers

Author(s)	Year	Vis. Goal	Vis. Method	CV task	Arch.	Data Sets
Simonyan et al. [61]	2014	General understanding	Pixel Displays	Classification	CNN	ImageNet
Yu et al. [80]	2014	General understanding	Pixel Displays	Classification	CNNs	ImageNet
Li et al. [41]	2015	General understanding	Pixel Displays	Representation learning	DCNN	Buffy Stickmen, ETHZ Stickmen, LSP, Synchronic Activities Stickmen, FLIC, WAF
Montavon et al. [50]	2015	General understanding	Heat maps	Classification	DNNs	ImageNet, MNIST
Yosinski et al. [79]	2015	General understanding	Pixel Displays	Classification	DNN	ImageNet
Mahendran and Vedaldi [47]	2016	General understanding	Pixel Displays	Representation learning	CNN	ILSVRC-2012, VOC2010
Wu et al. [74]	2016	General understanding	Pixel Displays	Recognition	DBN	ChaLearn LAP
Ciresan et al. [6]	2012	Architecture assessment	Pixel Displays, Confusion Matrix	Recognition	MCDNN	MNIST, NIST SD, CASIA-HWDB1.1, GTSRB traffic sign dataset, CIFAR10
Huang [28]	2012	Architecture assessment	Pixel Displays	Representation learning	CDBN	LFW
Szegedy et al. [63]	2013	Architecture assessment	Heat maps	Detection	DNN	VOC2007
Long et al.[45]	2014	Architecture assessment	Pixel Displays	Classification	CNNs	ImageNet, VOC
Taigman et al. [64]	2014	Architecture assessment	Pixel Displays	Representation learning	DNN	SFC, YTF, LFW
Yosinski et al. [78]	2014	Architecture assessment	Pixel Displays	Representation learning	CNN	ImageNet
Zhou et al. [84]	2014	Architecture assessment	Pixel Displays	Recognition	CNN	ImageNet, SUN397, MIT Indoor67, Scene15, SUNAttribute, Caltech-101, Caltech256, Stanford Action40, UIUC Event8
Zhou et al. [82]	2014	Architecture assessment	Pixel Displays	Classification	CNNs	SUN397, Scene15
Mahendran and Vedaldi [46]	2015	Architecture assessment	Pixel Displays	Representation learning	CNN	ILSVRC-2012
Samek et al. [56]	2015	Architecture assessment	Pixel Displays, Heat maps	Classification	DNN	SUN397, ILSVRC-2012, MIT
Wang et al. [70]	2015	Architecture assessment	Pixel Displays	Detection	CNNs	PASCAL3D+
Zhou et al. [83]	2015	Architecture assessment	Heat maps	Recognition	CNNs	ImageNet
Gruen et al. [25]	2016	Architecture assessment	Pixel Displays	Representation learning	DNN	ImageNet
Lin and Maji [42]	2016	Architecture assessment	Pixel Displays	Recognition	CNN	FMD, DTD, KTH-T2b, ImageNet
Nguyen et al. [53]	2016	Architecture assessment	Pixel Displays	Tracking	DNN	ImageNet, ILSVRC-2012
Zintgraf [85]	2016	Architecture assessment	Pixel Displays, HeatMaps	Classification	DCNN	ILSVRC
Erhan et al. [16]	2010	Model quality assessment	Pixel Displays	Representation learning	DBN	MNIST, Caltech-101
Krizhevsky et al. [35]	2012	Model quality assessment	Histogram	Classification	DCNN	ILSVRC-2010, ILSVRC-2012
Dai and Wu [8]	2014	Model quality assessment	Pixel Displays	Classification	CNNs	ImageNet, MNIST
Donahue et al. [11]	2014	Model quality assessment	Similarity layout	Classification	DNN	ILSVRC-2012, SUN397, Caltech-101, Caltech-UCSD Birds
Zeiler and Fergus [81]	2014	Model quality assessment	Pixel Displays, HeatMap	Classification	CNN	ImageNet, Caltech-101, Caltech256
Cao et al. [3]	2015	Model quality assessment	Pixel Displays	Tracking	CNN	ImageNet 2014
Wang et al. [71]	2015	Model quality assessment	Heat maps	Tracking	CNN	ImageNet
Dosovitskiy and Brox [12]	2016	Model quality assessment	Pixel Displays	Representation learning	CNN	ImageNet
Bruckner [2]	2014	User feedback integration	Pixel Displays, Confusion Matrix	Classification	DCNN	CIFAR-10, ILSVRC-2012
Harley [26]	2015	User feedback integration	Pixel Displays, Node-Link-Diagram	Recognition	CNNs	MNIST
Liu et al. [44]	2016	User feedback integration	Pixel Displays, Node-Link-Diagrams	Classification	CNNs	CIFAR10

Table 2 Overview of visualization goals

Category	# Papers	References
Architecture assessment	16	[6, 25, 28, 42, 45, 46, 53, 56, 63, 64, 70, 78, 82–85]
Model quality assessment	8	[3, 8, 11, 12, 16, 35, 71, 81]
General understanding	7	[41, 47, 50, 61, 74, 79, 80]
User feedback integration	3	[2, 26, 44]

of convolutional networks[1] [2, 79]. They demonstrated that such tools can provide a means to visualize the activations produced in response to user inputs and showed how the network behaves on unseen data.

Approaches providing deeper insights into the architecture were placed into the category **architecture assessment**. Authors focused their research on determining how these networks capture representations of texture, color and other features that discriminate an image from another, quite similar image [56]. Other authors tried to assess how these deep architectures arrive at certain decisions [42] and how the input image data affects the decision making capability of these networks under different conditions. These conditions include image scale, object translation, and cluttered background scenes. Further, authors investigated which features are learned, and whether the neurons are able to learn more than one feature in order to arrive at a decision [53]. Also, the contribution of image parts for activation of specific neurons was investigated [85] in order to understand for instance, what part of a dog's face needs to be visible for the network to detect it as a dog. Authors also investigated what types of features are transferred from lower to higher layers [78, 79], and have shown for instance, that scene centric and object centric features are represented differently in the network [84].

Eight papers contributed work on **model quality assessment**. Authors have focused their research on how the individual layers can be effectively visualized, as well as the effect on the network's performance. The contribution of each layer at different levels greatly influence their role played in computer vision tasks. One such work determined how the convolutional layers at various levels of the network show varied properties in tracking purposes [71]. Dosovitskiy and Bronx have shown that higher convolutional layers retain details of object location, color, and contour information of the image [12]. Visualization is used as a means to improve tools for finding good interpretations of features learned in higher levels [16]. Kriszhesvsky et al. focused on performance of individual layers and how performance degrades when certain layers in the network are removed [35].

Some authors researched **user feedback integration**. In the interactive node-link visualization in [26] the user can provide his/her own training data using a drawing area. This method is strongly tied to the used network and training data (MNIST hand written digit). In the *Ml-O-Scope* system users can interactively analyze convolutional neural networks [2]. Users are presented with a visualization of the current model performance, i.e. the a-posteriori probability distribution for input images and Pixel Displays of activations within selected network layers. They are also provided with a user interface for interactive adaption of model hyper-parameters. A visual analytics approach to DNN training has been proposed recently [44]. The authors present 3 case studies in which DNN experts evaluated a network, assessed errors, and found directions for improvement (e.g. adding new layers).

[1]Tools available http://yosinski.com/deepvis and https://github.com/bruckner/deepViz, last accessed 2016-09-08.

Table 3 Overview of visualization methods

Category	Sub-category	# Papers	References
Pixel Displays	Single image	24	[3, 6, 8, 12, 16, 25, 26, 41, 42, 44–47, 53, 56, 61, 70, 71, 74, 78–81, 85]
	Image batch	4	[2, 28, 35, 84]
	Part of image	2	[64, 82]
Heat maps		6	[50, 56, 63, 71, 81, 83, 85]
Confusion matrix		2	[2, 6]
Node-Link-Diagrams		2	[26, 44]
Similarity layout		1	[11]
Histogram		1	[35]

Table 4 Overview of categorization by Grün [25]

Category	# Papers	References
Deconvolution	24	[2, 3, 6, 8, 12, 16, 26, 28, 35, 41, 45, 50, 53, 56, 61, 63, 64, 70, 71, 80, 82–84]
Input modification	6	[44, 74, 78, 79, 81, 85]
Input reconstruction	4	[42, 46, 47, 61]

3.2 Visualization Methods

In this section we describe the different visualization methods applied to DNNs. An overview of the methods is provided in Table 3. We also categorize the papers according to Grün's taxonomy [25] in Table 4. In the following we describe the papers for each visualization method separately.

3.2.1 Pixel Displays

Most of the reviewed work has utilized pixel based activations as a means to visualize different features and layers of deep neural networks. The basic idea behind such visualization is that each pixel represents a data point. The color of the pixel corresponds to an activation value, the maximum gradient w.r.t. to a given class, or a reconstructed image. The different computational approaches for calculating maximum activations, sensitivity values, or reconstructed images are not within the scope of this chapter. We refer to the survey paper for feature visualizations in DNNs [25] and provide a categorization of papers into Grün's taxonomy in Table 4.

Mahendran and Vedaldi [46, 47] have visualized the information contained in the image by using a process of inversion using an optimized gradient descent function. Visualizations are used to show the representations at each layer of the

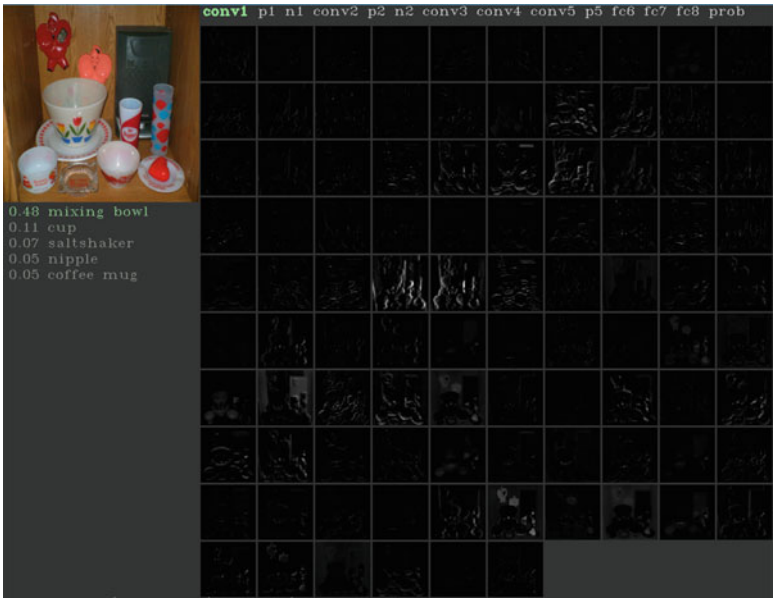

Fig. 2 Pixel based display. Activations of first convolutional layer generated with the DeepVis toolboxfrom [79] https://github.com/yosinski/deep-visualization-toolbox/

network (cf. Fig. 2). All the convolutional layers maintain photographically realistic representations of the image. The first few layers are specific to the input images and form a direct invertible code base. The fully connected layers represent data with less geometry and instance specific information. Activation signals can thus be invert back to images containing parts similar, but not identical to the original images. Cao et al. [3] have used Pixel Displays on complex, cluttered, single images to visualize their results of CNNs with feedback. Nguyen et al. [53] developed an algorithm to demonstrate that single neurons can represent multiple facets. Their visualizations show the type of image features that activate specific neurons. A regularization method is also presented to determine the interpretability of the images to maximize activation. The results suggest that synthesizing visualizations from activated neurons better represent input images in terms of the overall structure and color. Simonyan et al. [61] visualized data for deep convolutional networks. The first visualization is a numerically generated image to maximize a classification score. As second visualization, saliency maps for given pairs of images and classes indicate the influence of pixels from the input image on the respective class score, via back-propagation.

3.2.2 Heat Maps

In most cases, heat maps were used for visualizing the extent of feature activations of specific network layers for various computer vision tasks (e.g. classification [81], tracking [71], detection [83]). Heat maps have also been used to visualize the final network output, e.g. the classifier probability [63, 81]. The heat map visualizations are used to study the contributions of different network layers (e.g. [71]), compare different methods (e.g. [50]), or investigate the DNNs inner features and results on different input images [83]. Zintgraf et al. [85] used heat maps to visualize image regions in favor of, as well as image regions against, a specific class in one image. Authors use different color codings for their heat maps: blue-red-yellow color schemes [71, 81, 83], white-red schemes [50], blue-white-red schemes [85], and also a simple grayscale highlighting interesting regions in white [63].

3.2.3 Confusion Matrix and Histogram

Two authors have shown the confusion matrix to illustrate the performance of the DNN w.r.t., a classification task (see Fig. 3). Bruckner et al. [2] additionally encoded the value in each cell using color (darker color represents higher values). Thus, in this visualization, dark off-diagonal spots correspond to large errors. In [6] the encoding used is different: each cell value is additionally encoded by the size of a square. Cells containing large squares represent large values; a large off-diagonal square corresponds to a large error between two classes. Similarly, in one paper histograms have been used to visualize the decision uncertainty of a classifier, indicating using color whether the highest-probable class is the correct one [35].

3.2.4 Similarity Based Layout

In the context of DNNs, similarity based layouts so far have been applied only by Donahue et al. [11], who specifically used t-distributed stochastic neighbor embedding (t-SNE) [67] of feature representations. The authors projected feature representations of different networks layers into the 2-dimensional space and found

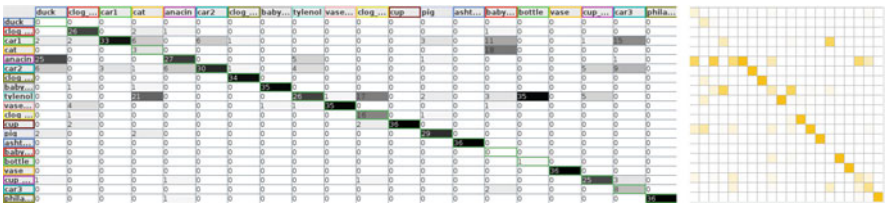

Fig. 3 Confusion Matrix example. Showing classification results for the COIL-20 data set. Screenshots reproduced with software from [59]

Fig. 4 Similarity based layout of the MNIST data set using raw features. Screenshot was taken with a JavaScript implementation of t-SNE [67] https://scienceai.github.io/tsne-js/

a visible clustering for the higher layers in the network, but none for features of the lower network layer. This finding corresponds to the general knowledge of the community that higher levels learn semantic or high-level features. Further, based on the projection the authors could conclude that some feature representation is a good choice for generalization to other (unseen) classes and how traditional features compare to feature representations learned by deep architectures. Figure 4 provided an example of the latter.

3.2.5 Node-Link Diagrams

Two authors have approach DNN visualization with node-link diagrams (see examples in Fig. 5). In his interactive visualization approach, Adam Harley represented layers in the neural networks as nodes using Pixel Displays, and activation levels as edges [26]. Due to the denseness of connections in DNNs only active edges are visible. Users can draw input images for the network and interactively explore how the DNN is trained. In CNNVis [44] nodes represent neuron clusters and are visualized in different ways (e.g., activations) showing derived features for the clusters.

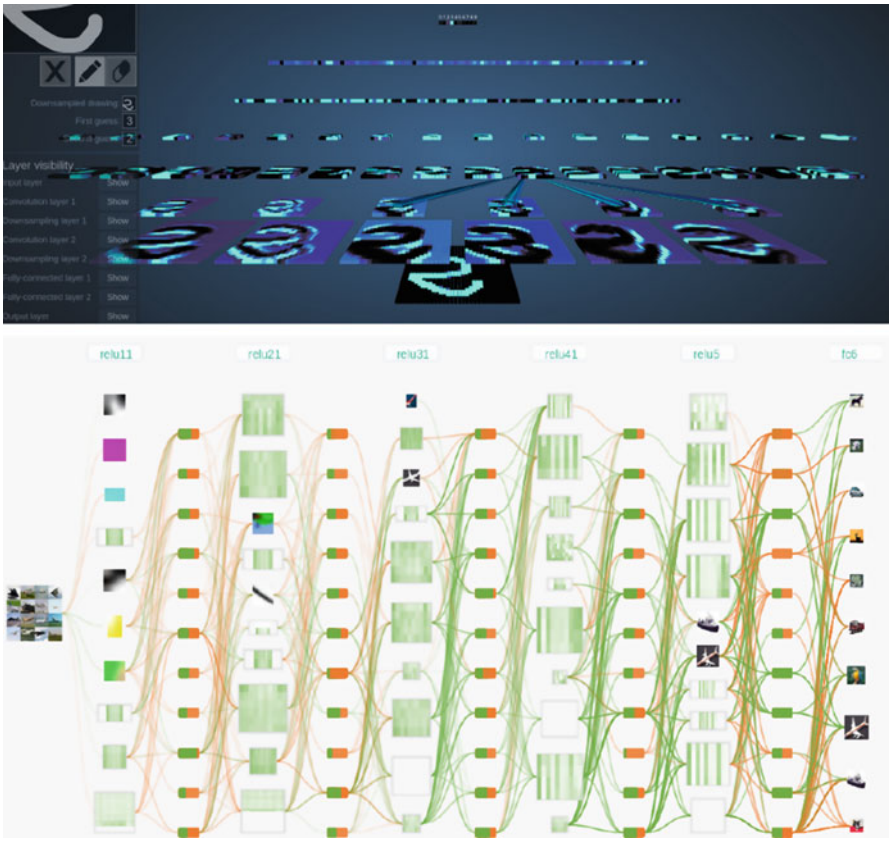

Fig. 5 Node-link diagrams of DNNs. *Top*: Example from [26] taken with the online application at http://scs.ryerson.ca/~aharley/vis/conv/. *Bottom*: screenshot of the CNNVis system [44] taken with the online application at http://shixialiu.com/publications/cnnvis/demo/

3.3 Network Architecture and Computer Vision Task

Table 5 provides a summary of the architecture types. The majority of papers applied visualizations to CNN architectures (18 papers), while 8 papers dealt with the more general case of DNNs. Only 8 papers have investigated more special architectures, like DCNN (4 papers), DBNs (2 papers), CDBN (1 paper) and MCDNNs (1 paper).

Table 6 summarizes the computer vision tasks for which the DNNs have been trained. Most networks were trained for classification (14 papers), some for representation learning and recognition (9 and 6 papers, respectively). Tracking and Detection were pursued the least often.

Table 5 Overview of network architecture types

Category	# Papers	References
CNN	18	[3, 8, 12, 26, 42, 44–47, 61, 70, 71, 78, 80–84]
DNN	8	[11, 25, 50, 53, 56, 63, 64, 79]
DCNN	4	[2, 35, 41, 85]
DBN	2	[16, 74]
CDBN	1	[28]
MCDNN	1	[6]

Table 6 Overview of computer vision tasks

Category	# Papers	References
Classification	14	[2, 8, 11, 35, 44, 45, 50, 56, 61, 79–82, 85]
Representation learning	9	[12, 16, 25, 28, 41, 46, 47, 64, 78]
Recognition	6	[6, 26, 42, 74, 83, 84]
Tracking	3	[3, 53, 71]
Detection	2	[63, 70]

3.4 Data Sets

Table 7 provides an overview of the data sets used in the reviewed papers. In the field of classification and detection, the ImageNet dataset represent the most frequently used dataset, used around 21 times. Other popular datasets used in tasks involving detection and recognition such as Caltech101, Caltech256 etc. have been used 2–3 times (e.g. in [11, 56, 81, 84]).

While ImageNet and its subsets (e.g. ISLVRC) are large datasets with around 10,000,000 images each, there are smaller datasets such as the ETHZ stickmen and VOC2010 which are generally used for fine-grained classification and learning. VOC2010, consisting of about 21,738 images, has been used twice, while more specialized data sets, such as Buffy Stickmen for representation learning, have been used only once in the reviewed papers [41]. There are datasets used in recognition with fewer classes such as CIFAR10, consisting of 60,000 colour images, with about 10 classes; and MNIST used for recognition of handwritten digits.

4 Discussion

In this section we discuss the implications of the findings from the previous section with respect to the research questions. We start the discussion by evaluating the results for the stated research questions.

RQ-1 (Which insights can be gained about DNN models by means of visualization) has been discussed along with the single papers in the previous section in

Table 7 Overview of data sets sorted after their usage. Column "#" refers to the number of papers in this survey using this data set

Data Set	Year	# Images	CV Task	Comment	#	References
ImageNet [9]	2009	14,197,122	Classification, tracking	21,841 synsets	21	[2, 3, 11, 12, 25, 35, 42, 46, 47, 53, 53, 56, 56, 61, 71, 78, 79, 81, 84, 85, 85]
ILSVRC2012 [55]	2015	1,200,000	Classification, detection, representation learning	1000 object categories	7	[2, 11, 35, 46, 47, 53, 56]
VOC2010 [18]	2010	21, 738	Detection, representation learning	50/50 train-test split	3	[45, 47, 63]
Caltech-101 [20]	2006	9146	Recognition, classification	101 categories	3	[11, 81, 84]
Places [84]	2014	2,500,000	Classification, recognition	205 scene categories	2	[56, 84]
Sun397 [76]	2010	130,519	Classification, recognition	397 categories	2	[56, 84]
Caltech256 [23]	2007	30,607	Classification,recognition	256 categories	2	[81, 84]
LFW [29]	2007	13,323	Representation learning	5749 faces, 6000 pairs	2	[28, 64]
MNIST [37]	1998	70,000	Recognition	60,000 train, 10,000 test 10 classes, handwritten digits	2	[6, 16]
DTD [5]	2014	5640	Recognition	47 terms (categories)	1	[42]
ChaLearn LAP [17]	2014	47,933	Recognition	RGB-D gesture videos with 249 gestures labels, each 249 for train/testing/validation	1	[74]
SFC [64]	2014	4,400,000	Representation learning	Photos of 4030 people	1	[64]
PASCAL3D+ [75]	2014	30,899	Detection		1	[70]
FLIC [57]	2013	5003	Representation learning	30 movies with person detector, 20% for testing	1	[40]
Synchronic Activities Stickmen [15]	2012	357	Representation learning	Upper-body annotations	1	[40]
Buffy Stickmen [30]	2012	748	Representation learning	Ground-truth stickmen annotations, annotated video frames	1	[40]
SUNAttribute [54]	2012	14,000	Recognition	700 categories	1	[84]
CASIA-HWDB1.1 [43]	2011	1,121,749	Recognition	897,758 train, 223,991 test ,3755 classes, Chinese handwriting	1	[6]
GTSRB traffic sign dataset [62]	2011	50,000	Recognition	>40 classes, single-image, multi-class classification	1	[6]
Caltech-UCSD Birds [68]	2011	11,788	Classification	200 categories	1	[11]
YTF [73]	2011	3425	Representation learning	Videos, subset of LFW	1	[64]
Stanford Action40 [77]	2011	9532	Recognition	40 actions, 180–300 images per action class	1	[84]
WAF [14]	2010	525	Representation learning	Downloaded via Google Image Search	1	[40]
LSP [31]	2010	2000	Representation learning	Pose annotated images with 14 joint location	1	[40]
ETHZ Stickmen [13]	2009	549	Representation learning	Annotated by a 6-part stickman	1	[40]
CIFAR10 [34]	2009	60,000	Recognition	50,000 training and 10,000 test of 10 classes	1	[6]
FMD [60]	2009	1000	Recognition	10 categories, 100 images per category	1	[42]
UIUC Event8 [39]	2007	1579	Recognition	sports event categories	1	[84]
KTH-T2b [4]	2005	4752	Recognition	11 materials captured under controlled scale, pose, and illumination	1	[42]
Scene15 [19]	2005	4485	Recognition	200–400 images per class of 15 class scenes	1	[84]
NIST SD 19 [24]	1995	800,000	Recognition	Forms and digits	1	[6]

detail. We showed by examples which visualizations have previously been shown to lead to which insights. For instance, visualizations are used to learn which features are represented in which layer of a network or which part of the image a certain node reacts to. Additionally, visualizing synthetic input images which maximize activation allows to better understand how a network as a whole works. To strengthen our point here, we additionally provide some quotes from authors:

Heat maps: *"The visualisation method shows which pixels of a specific input image are evidence for or against a node in the network."* [85]

Similarity layout: *"[. . .] first layers learn 'low-level' features, whereas the latter layers learn semantic or 'high-level' features. [. . .] GIST or LLC fail to capture the semantic difference [. . .]"* [11]

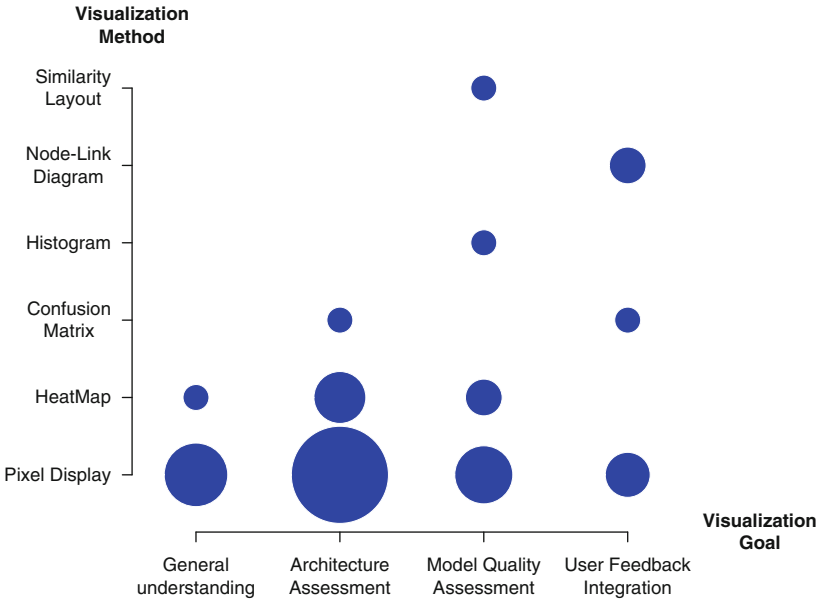

Fig. 6 Relation of visualization goals and applied methods in the surveyed papers following our taxonomy. Size of the circles corresponds to the (square root of the) number of papers in the respective categories. For details on papers see Table 1

Pixel Displays: *"[...] representations on later convolutional layers tend to be somewhat local, where channels correspond to specific, natural parts (e.g. wheels, faces) instead of being dimensions in a completely distributed code. That said, not all features correspond to natural parts [...]"* [79]

The premise to use visualization is thus valid, as the publications agree that visualizations help to understand the functionality and behavior of DNNs in computer vision. This is especially true when investigating specific parts of the DNN.

To answer **RQ-2** (Which visualization methods are appropriate for which kind of insights?) we evaluated which visualizations were applied in the context of which visualization goals. A summary is shown in Fig. 6. It can be seen that not all methods were used in combination with all goals, which is not surprising. For instance, no publication used a similarity layout for assessing the architecture. This provides hints on possibilities for further visualization experiments.

Pixel Displays were prevalent for architecture assessment and general understanding. This is plausible since DNNs for computer vision work on the images themselves. Thus, Pixel Displays preserve the spatial-context of the input data, making the interpretation of the visualization straight-forward. This visualization, however, method has its own disadvantages and might not be the ideal choice in all cases. The visualization design space is extremely limited, i.e. constrained

to a simple color mapping. Especially for more complex research questions, extending this space might be worthwhile, as the other visualization examples in this review show.

The fact that a method has not been used w.r.t. a certain goal does not necessarily mean that it would not be appropriate. It merely means that authors so far achieved their goal with a different kind of visualization. The results based on our taxonomy, cf. Fig. 6 and Table 1, hint at corresponding *white spots*. For example, node-link diagrams are well suited to visualize dependencies and relations. Such information could be extracted for architecture assessment as well, depicting which input images and activation levels correlate highly to activations within individual layers of the network. Such a visualization will neither be trivial to create nor to use, since this first three part correlation requires suitable hyper-graph visualization metaphor, but the information basis is promising. Similar example ideas can be constructed for the other white spots in Fig. 6 and beyond.

5 Summary and Conclusion

In this chapter we surveyed visualizations of DNNs in the computer vision domain. Our leading questions were: "Which insights can be gained about DNN models by means of visualization?" and "Which visualization methods are appropriate for which kind of insights?" A taxonomy containing the categories *visualization method*, *visualization goal*, *network architecture type*, *computer vision task*, and *data set* was developed to structure the domain. We found that Pixel Displays were most prominent among the methods, closely followed by heat maps. Both is not surprising, given that images (or image sequences) are the prevalent input data in computer vision. Most of the developed visualizations and/or tools are expert tools, designed for the usage of DNN/computer vision experts. We found no interactive visualization allowing to integrate user feedback directly into the model. The closest approach is the semi-automatic CNNVis tool [44]. An interesting next step would be to investigate which of the methods have been used in other application areas of DNNs, such as speech recognition, where Pixel Displays are not the most straight-forward visualization. It would be also interesting to see which visualization knowledge and techniques could be successfully transferred between these application areas.

References

1. Becker, B., Kohavi, R., Sommerfield, D.: Visualizing the simple Bayesian classifier. In: KDD Workshop Issues in the Integration of Data Mining and Data Visualization (1997)
2. Bruckner, D.M.: Ml-o-scope: a diagnostic visualization system for deep machine learning pipelines. Tech. Rep. UCB/EECS-2014-99, University of California at Berkeley (2014)

3. Cao, C., Liu, X., Yang, Y., Yu, Y., Wang, J., Wang, Z., Huang, Y., Wang, L., Huang, C., Xu, W., Ramanan, D., Huang, T.S.: Look and think twice: capturing top-down visual attention with feedback convolutional neural networks. In: 2015 IEEE International Conference on Computer Vision (ICCV) (2015)
4. Caputo, B., Hayman, E., Mallikarjuna, P.: Class-specific material categorisation. In: Tenth IEEE International Conference on Computer Vision, vol. 1 (2005)
5. Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A.: Describing textures in the wild. CoRR, abs/1311.3618 (2014)
6. Ciresan, D., Meier, U., Schmidhuber, J.: Multi-column deep neural networks for image classification. In: Computer Vision and Pattern Recognition, pp. 3642–3649 (2012)
7. Cybenko, G.: Approximation by superpositions of a sigmoidal function. Math. Control Signals Syst. **2**(4), 303–314 (1989)
8. Dai, J., Wu, Y.N.: Generative modeling of convolutional neural networks. CoRR, abs/1412.6296 (2014)
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255, June 2009
10. Di Battista, G., Eades, P., Tamassia, R., Tollis, I.G.: Algorithms for drawing graphs: an annotated bibliography. Comput. Geom. **4**(5), 235–282 (1994)
11. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: a deep convolutional activation feature for generic visual recognition. In: International Conference on Machine Learning (2014)
12. Dosovitskiy, A., Brox, T.: Inverting visual representations with convolutional networks. In: IEEE Conference on Computer Vision and Pattern Recognition (2016).
13. Eichner, M., Ferrari, V.: Better appearance models for pictorial structures. In: Proceedings of the British Machine Vision Conference, pp. 3.1–3.11. BMVA Press, Guildford (2009). doi:10.5244/C.23.3
14. Eichner, M., Ferrari, V.: We are family: joint pose estimation of multiple persons. In: Proceedings of the 11th European Conference on Computer Vision: Part I, pp. 228–242. Springer, Berlin/Heidelberg (2010)
15. Eichner, M., Ferrari, V.: Human pose co-estimation and applications. IEEE Trans. Pattern Anal. Mach. Intell. **34**(11), 2282–2288 (2012)
16. Erhan, D., Courville, A., Bengio, Y.: Understanding representations learned in deep architectures. Tech. Rep. 1355, Université de Montréal/DIRO, October 2010
17. Escalera, S., Baró, X., Gonzalez, J., Bautista, M.A., Madadi, M., Reyes, M., Ponce-López, V., Escalante, H.J., Shotton, J., Guyon, I.: Chalearn looking at people challenge 2014: Dataset and results. In: Workshop at the European Conference on Computer Vision (2014)
18. Everingham, M., van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The Pascal visual object classes (voc) challenge. Int. J. Comput. Vis. **88**(2), 303–338 (2010)
19. Fei-Fei, L., Perona, P.: A Bayesian hierarchical model for learning natural scene categories. In: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 - Volume 02, pp. 524–531. IEEE Computer Society, Washington, DC (2005)
20. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE Trans. Pattern Anal. Mach. Intell. **28**(4), 594–611 (2006)
21. Fuchs, R., Waser, J., Gröller, E.: Visual human+machine learning. Proc. Vis. 09 **15**(6), 1327–1334 (2009)
22. Fukushima, K., Miyake, S.: Neocognitron: a new algorithm for pattern recognition tolerant of deformations and shifts in position. Pattern Recogn. **15**(6), 455–469 (1982)
23. Griffin, G., Houlub, A., Perona, P.: Caltech-256 object category dataset. Tech. Rep., California Institute of Technology (2007)
24. Grother, P.J.: NIST special database 19 – Handprinted forms and characters database. Technical report, Natl. Inst. Stand. Technol. (NIST) (1995). https://www.nist.gov/sites/default/files/documents/srd/nistsd19.pdf

25. Grün, F., Rupprecht, C., Navab, N., Tombari, F.: A taxonomy and library for visualizing learned features in convolutional neural networks. In: Proceedings of the International Conference on Machine Learning (2016)
26. Harley, A.W.: An Interactive Node-Link Visualization of Convolutional Neural Networks, pp. 867–877. Springer International Publishing, Cham (2015)
27. Hinton, G.E., Osindero, S., Teh, Y.-W.: A fast learning algorithm for deep belief nets. Neural Comput. **18**(7), 1527–1554 (2006)
28. Huang, G.B.: Learning hierarchical representations for face verification with convolutional deep belief networks. In: *Proceedings Conference on Computer Vision and Pattern Recognition*, pp. 2518–2525. IEEE Computer Society, Washington, DC (2012)
29. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: a database for studying face recognition in unconstrained environments. Tech. Rep. 07–49, University of Massachusetts, Amherst, October 2007
30. Jammalamadaka, N., Zisserman, A., Eichner, M., Ferrari, V., Jawahar, C.: Has my algorithm succeeded? an evaluator for human pose estimators. In: European Conference on Computer Vision (2012)
31. Johnson, S., Everingham, M.: Clustered pose and nonlinear appearance models for human pose estimation. In: Proceedings of the British Machine Vision Conference (2010). doi:10.5244/C.24.12
32. Keim, D., Bak, P., Schäfer, M.: Dense Pixel Displays. In: Liu, L., Öszu, M.T. (eds.) Encyclopedia of Database Systems, pp. 789–795. Springer, New York (2009)
33. Kohavi, R.: Data mining and visualization. Invited talk at the National Academy of Engineering US Frontiers of Engineers (NAE) (2000)
34. Krizhevsky, A.: Learning multiple layers of features from tiny images. Tech. Rep., University of Toronto (2009)
35. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 25, pp. 1097–1105. Curran Associates, Inc., Red Hook (2012)
36. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521** 436–444 (2015)
37. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
38. Lee, H., Grosse, R., Ranganath, R., Ng, A.Y.: Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: Proceedings of the 26th Annual International Conference on Machine Learning, pp. 609–616. ACM, New York (2009)
39. Li, L., Fei-Fei, L.: What, where and who? classifying events by scene and object recognition. In: IEEE International Conference on Computer Vision (2007)
40. Li, S., Liu, Z.-Q., Chan, A.B.: Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (2014)
41. Li, S., Liu, Z.-Q., Chan, A.B.: Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network. Int. J. Comput. Vis. **113**(1), 19–36 (2015)
42. Lin, T.-Y., Maji, S.: Visualizing and understanding deep texture representations. In: *Conference on Computer Vision and Pattern Recognition* (2016)
43. Liu, C.-L., Yin, F., Wang, D.-H., Wang, Q.-F.: Casia online and offline Chinese handwriting databases. In: 2011 International Conference on Document Analysis and Recognition (2011)
44. Liu, M., Shi, J., Li, Z., Li, C., Zhu, J., Liu, S.: Towards better analysis of deep convolutional neural networks. CoRR, abs/1604.07043 (2016)
45. Long, J., Zhang, N., Darrell, T.: Do convnets learn correspondence? CoRR, abs/1411.1091 (2014)
46. Mahendran, A., Vedaldi, A.: Understanding deep image representations by inverting them. In: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (2015)
47. Mahendran, A., Vedaldi, A.: Visualizing deep convolutional neural networks using natural pre-images. In: International Journal of Computer Vision (2016)

48. McCulloch, W.S., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. Bull. Math. Biophys. **5**(4), 115–133 (1943)
49. Minsky, M., Papert, S.: Perceptrons: An Introduction to Computational Geometry. MIT Press, Cambridge, MA (1969)
50. Montavon, G., Bach, S., Binder, A., Samek, W., Müller, K.-R.: Explaining nonlinear classification decisions with deep Taylor decomposition. CoRR, abs/1512.02479 (2015)
51. Munzner, T.: Visualization Analysis and Design. A K Peters Visualization Series. CRC Press, Boca Raton, FL (2014)
52. Nguyen, G.P., Worring, M.: Interactive access to large image collections using similarity-based visualization. J. Vis. Lang. Comput. **19**(2), 203–224 (2008)
53. Nguyen, A.M., Yosinski, J., Clune, J.: Multifaceted feature visualization: uncovering the different types of features learned by each neuron in deep neural networks. CoRR, abs/1602.03616 (2016)
54. Patterson, G.: Sun attribute database: discovering, annotating, and recognizing scene attributes. In: Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2751–2758. IEEE Computer Society, Washington, DC (2012)
55. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. **115**(3), 211–252 (2015)
56. Samek, W., Binder, A., Montavon, G., Bach, S., Müller, K.-R.: Evaluating the visualization of what a deep neural network has learned. CoRR, abs/1509.06321 (2015)
57. Sapp, B., Taskar, B.: Modec: multimodal decomposable models for human pose estimation. In: Proceedings of the Computer Vision and Pattern Recognition (2013)
58. Seifert, C., Granitzer, M.: User-based active learning. In: Proceedings of 10th International Conference on Data Mining Workshops, pp. 418–425 (2010)
59. Seifert, C., Lex, E.: A novel visualization approach for data-mining-related classification. In: Proceedings of the International Conference on Information Visualisation (IV), pp. 490–495. Wiley, New York (2009)
60. Sharan, L., Rosenholtz, R., Adelson, E.: Material perception: what can you see in a brief glance? J. Vis. **9**, 784 (2009). doi:10.1167/9.8.784
61. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: visualising image classification models and saliency maps. CoRR, abs/1312.6034 (2014)
62. Stallkamp, J., Schlipsing, M., Salmen, J., Igel, C.: The German traffic sign recognition benchmark: a multi-class classification competition. In: Neural Networks (IJCNN), The 2011 International Joint Conference on (2011)
63. Szegedy, C., Toshev, A., Erhan, D.: Deep neural networks for object detection. In: Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 26, pp. 2553–2561. Curran Associates, Inc., Red Hook (2013)
64. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: Deepface: closing the gap to human-level performance in face verification. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701–1708 (2014)
65. Thearling, K., Becker, B., DeCoste, D., Mawby, W., Pilote, M., Sommerfield, D.: Chapter Visualizing data mining models. In: Information Visualization in Data Mining and Knowledge Discovery, pp. 205–222. Morgan Kaufmann Publishers Inc., San Francisco, CA (2001)
66. Urbanek, S.: Exploring statistical forests. In: Proceedings of the 2002 Joint Statistical Meeting (2002)
67. van der Maaten, L., Hinton, G.E.: Visualizing high-dimensional data using t-sne. J. Mach. Learn. Res. **9**, 2579–2605 (2008)
68. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset. Tech. Rep. CNS-TR-2011-001, California Institute of Technology (2011)
69. Wang, J., Yu, B., Gasser, L.: Classification visualization with shaded similarity matrices. Tech. Rep., GSLIS University of Illinois at Urbana-Champaign (2002)
70. Wang, J., Zhang, Z., Premachandran, V., Yuille, A.L.: Discovering internal representations from object-cnns using population encoding. CoRR, abs/1511.06855 (2015)

71. Wang, L., Ouyang, W., Wang, X., Lu, H.: Visual tracking with fully convolutional networks. In: IEEE International Conference on Computer Vision (2015)
72. Wilkinson, L., Friendly, M.: The history of the cluster heat map. Am. Stat. **63**(2), 179–184 (2009)
73. Wolf, L., Hassner, T., Maoz, I.: Face recognition in unconstrained videos with matched background similarity. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2011)
74. Wu, D., Pigou, L., Kindermans, P.J., Le, N.D.H., Shao, L., Dambre, J., Odobez, J.M.: Deep dynamic neural networks for multimodal gesture segmentation and recognition. IEEE Trans. Pattern Anal. Mach. Intell. **38**(8), 1583–1597 (2016)
75. Xiang, Y., Mottaghi, R., Savarese, S.: Beyond Pascal: a benchmark for 3d object detection in the wild. In: IEEE Winter Conference on Applications of Computer Vision (2014)
76. Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: Sun database: large-scale scene recognition from abbey to zoo. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (2010)
77. Yao, B., Jiang, X., Khosla, A., Lin, A.L., Guibas, L., Fei-Fei, L.: Human action recognition by learning bases of action attributes and parts. In: International Conference on Computer Vision (2011)
78. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 27, pp. 3320–3328. Curran Associates, Inc., Red Hook (2014)
79. Yosinski, J., Clune, J., Nguyen, A., Fuchs, T., Lipson, H.: Understanding neural networks through deep visualization. In: Proceedings of the International Conference on Machine Learning (2015)
80. Yu, W., Yang, K., Bai, Y., Yao, H., Rui, Y.: Visualizing and comparing convolutional neural networks. CoRR, abs/1412.6631 (2014)
81. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Computer Vision 13th European Conference (2014)
82. Zhou, B., Khosla, A., Lapedriza, À., Oliva, A., Torralba, A.: Object detectors emerge in deep scene cnns. CoRR, abs/1412.6856 (2014)
83. Zhou, B., Khosla, A., Lapedriza, À., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. CoRR, abs/1512.04150 (2015)
84. Zhou, B., Lapedriza, À., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 27, pp. 487–495. Curran Associates, Inc., Red Hook (2014)
85. Zintgraf, L.M., Cohen, T., Welling, M.: A new method to visualize deep neural networks. CoRR, abs/1603.02518 (2016)