Consistency of Scale Equivariance in Internal Representations of CNNs

Vincent Andrearczyk[1], Mara Graziani[1,2], Henning Müller[1,2], and Adrien Depeursinge[1,3]

[1]*Institute of Information Systems, University of Applied Sciences Western Switzerland (HES-SO), Sierre, Switzerland*
[2]*Hopitaux Universitaires de Genève (HUG), Geneva, Switzerland*
[3]*Centre Hospitalier Universitaire Vaudois (CHUV), Lausanne, Switzerland*

August 27, 2020

Abstract

Despite the approximate invariance to scale learned in deep Convolutional Neural Networks (CNNs) trained on natural images, intermediate layers have been shown to contain information of scale while the invariance is only obtained in the final layers. In this paper, we experimentally analyze how this scale information is encoded in the hidden layers. Linear regression of scale is used to (i) evaluate whether scale information can be encoded, at a given layer, by individual response maps or a combination of many of them is necessary; (ii) evaluate whether the encoding of scale is shared among classes. If we can find a direction representative of scale variations in the hidden space, is this consistent across the data manifold? Or is it rather encoded locally within class-specific neighborhoods? We observe that scale information is encoded as a combination of a few response maps (around 3%) and that the encoding is relatively consistent across classes, with some amount of class-specific encoding.

1 Introduction

Convolutional Neural Networks (CNNs) can implicitly learn an approximate invariance to transformations and variations in the training data [1]. Excellent generalization to unseen images has been obtained in natural images (ImageNet [2]), despite their large intra-class variability. In particular, CNNs must learn invariance to scale to be able to recognize textures, objects and scenes regardless of the point of view. Scale plays a crucial role in computer vision and medical imaging [3]. Besides, transfer learning has been extensively used in many domains. Various empirical studies, e.g in medical imaging, compared vanilla CNNs with pre-trained models with little attempts in understanding why one may work better than another in different scenarios besides the speed of convergence. Studying scale equivariance in pre-trained CNNs informs us on the adaptability to domains distant from natural images, in which scale often carries crucial information. Motivated by the transfer learning of CNNs trained on natural images to other domains in which the scale and object size may be informative (e.g. fixed viewpoint in medical imaging), we studied the presence of scale information in intermediate activations in [4, 5]. It was shown that, for a given class, the scale can be linearly regressed within intermediate activations and that the invariance is learned in the final layers. In [5], this finding was used to improve the regression of magnification in histopathology images using pretrained CNNs.

In this paper, we build on top of the analyses in [4, 5] to better understand how the scale information is encoded in the internal activations of commonly used state of the art CNNs [1]. We evaluate empirically two key questions related to the learning process and data processing of CNNs. Is scale encoded by a single or by multiple response maps? Is scale encoded differently for different classes? To address the first question, we evaluate if a minimal combination of response maps suffices to encode the scale information, starting from a single response only. While individualized neuron responses to a specific representation of a concept (e.g. scale, lighting or texture) were analyzed in [6, 7] by visualizing activations for exemplar input images, this paper provides a quantitative analysis. Second, the initial evaluation of scale regression in [4] considered only individual classes, leaving opened the question of whether scale is learned individually for each class or as a general feature that comprises multiple classes. We evaluate the consistency of the direction of increasing values of scale in the deep hidden space. This helps to understand whether scale encoded in the same way across the data manifold or if local regression and manifold learning should be used to capture the scale encoding.

As described in Section 3, the analysis is based on linear regression of scale at intermediate layers of CNNs trained on ImageNet. The scale is measured as a ratio based on manually annotated bounding boxes from the Pascal-VOC annotations. Section 4 is dedicated to the alignment of the scale regression along individual response maps. The consistency of the scale encoding across the data manifold is studied in Section 5.

2 Related Work

Several researchers have studied the invariance and equivariance to scale learned in CNNs by evaluating internal activations for images at different scales [6, 8]. More generally, the equivariance to geometric image transformations such as flips and rescaling in the intermediate feature space was studied in [9], concluding that scale invariance is implicitly learned in CNNs as prediction results are not improved by reversing the scaling transformations in the feature space. Contrasting these results, the vulnerability of standard CNNs to adversarial attacks with transformations including scaling was studied in [10], supporting the need for built-in invariance. A supervised training method was proposed in [11] to disentangle the transformations including rotations and scales, providing built-in equivariance properties.

Particularly related to our work, post-hoc interpretability methods, as defined in [12], interpret trained models without modifying their optimization. Among these methods, linear models have been used as probes to explain intermediate network layers, in line with our strategy to analyze scale equivariance. Linear classifier probes [13] were proposed to analyze class-separability at intermediate network layers in terms of the classification of the class labels by a linear model. Testing with Concept Activation Vectors (TCAV) [14] and Regression Concept Vectors (RCVs) [15, 16] were developed to analyze the presence of concepts (binary and continuous measures) in intermediate layers of a deep network by linear classification and regression, respectively. Linear probes, and particularly concept-based ones, have shown relevant results in different analyses and applications [17, 18]. This approach is well suited for our task of investigating scale equivariance in deep representation by modeling a linear probe for scale variations.

3 Methods

This section describes the notations and methods used in the experiments. We use an InceptionV3 [1] CNN trained on ImageNet [2][1]. We analyze this model at an intermediate layer by looking at the activations for different images of varying scale. In this paper, we evaluate the *mixed8* layer in InceptionV3 as it is a rather deep layer with large receptive fields and was particularly shown to encode scale in [4, 5].

We consider a scaling transformation of an input image I, $g_\sigma(I)$, parameterized by a scaling factor σ. We search a predictable linear transformation $g'_\sigma(\phi(I))$ in the d-dimensional feature space $\phi(\cdot)$ of the scaling $g_\sigma(\cdot)$ in the input space. If such a property is found, the representation $\phi(\cdot)$ is linearly equivariant to scale. To find a linear equivariance, one can search a regression vector $\mathbf{v}$ in the feature space to predict the scaling factor σ as a linear combination of the features $\phi_i(g_\sigma(I))$:[2]

$$\sigma = \sum_{i=1}^{d} v_i \phi_i(g_\sigma(I)) = \mathbf{v} \cdot \phi(g_\sigma(I)). \tag{1}$$

[1] Similar results were observed with ResNet50.

[2] For simplicity, we omit the intercept. In Eq. (1), the intercept would be v_0 with $\phi_0(g_\sigma(I)) = 1$

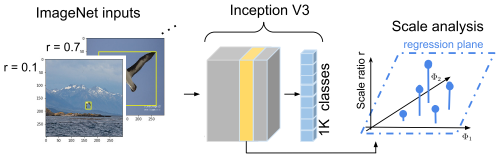


Figure 1: Overview of the scale analysis. The scale equivariance is analyzed at a given CNN layer (in yellow). Each data point I projected into this hidden space $\phi(I)$ is associated with its scale ratio r. We represent a regression plane corresponding to the vector $\mathbf{v}$ in two dimensions.

In this scenario, we can represent $g'_\sigma(\cdot)$ as a translation matrix in $\mathbb{R}^d$ by σ along $\mathbf{v}$, so that $g'_\sigma(\phi(I)) = \phi(I) + \mathbf{v} \cdot \sigma$.

To approximate $g'_\sigma(\cdot)$ and generalize to multiple input images, we consider images of objects that appear at various scales as shown in Fig. 2, and learn the regression of their corresponding scale ratios as in (1). The correlation of hidden features with the scale can be evaluated for rescaled versions of an image as in [8]. The rescaling, however, induces artifacts that can be highly correlated with response maps, as shown in [4, 19].

The approach used to analyze the scale information at a given layer is summarized in Fig. 1 and detailed in the following. We extract the activations for a set of inputs and spatially average the d response maps to obtain a d-dimensional feature vector $\phi(I)$ for each input image I. The averaging step is needed to remove locality and to reduce dimensionality as discussed in [16]. When dealing with flattened final layers, the spatial averaging does not apply. We also normalize the $\phi(I)$'s to have zero mean and unit variance on each dimension using the set of regression training data (not to be mistaken with the CNN training data). We then learn the regression vector $\mathbf{v}$ in (1). To this end, we regress a scale ratio which we define as $r = \sqrt{\frac{h_b \times w_b}{h_i \times w_i}}$, where h_b, w_b, h_i and w_i are the height and width of the object bounding box and of the image. Additionally to the standard residual sum of squares regression, we use a Lasso (L_1 norm minimization) regularization. It is used as a feature selection to evaluate the regression with a few features. The Lasso optimization function of the linear regression can be written as the residual sum of squares with L_1 penalization as follows.

$$\sum_{j=1}^{n} (y_j - \hat{y}_j)^2 + \alpha \sum_{i=0}^{d} |v_i|, \tag{2}$$

where n is the number of training images, y_j and $\hat{y}_j$ are the ground truth and predicted scale ratio r, α is the weight given to the Lasso regularization. Once optimized for a set of training images, the regression is evaluated on held-out images either from the same class or from another class using the R^2 coefficient of determination. $R^2 > 0$ means that the model is doing better than predicting the mean of the test set, while $R^2 = 1$ represents a perfect prediction. In the experiments, 220 training images from the *albatross* ImageNet class (ID: n02058221) are randomly drawn to regress the scale. Similarly, 220 test images are used as test set either from the same or from a different class as specified in the experiments.

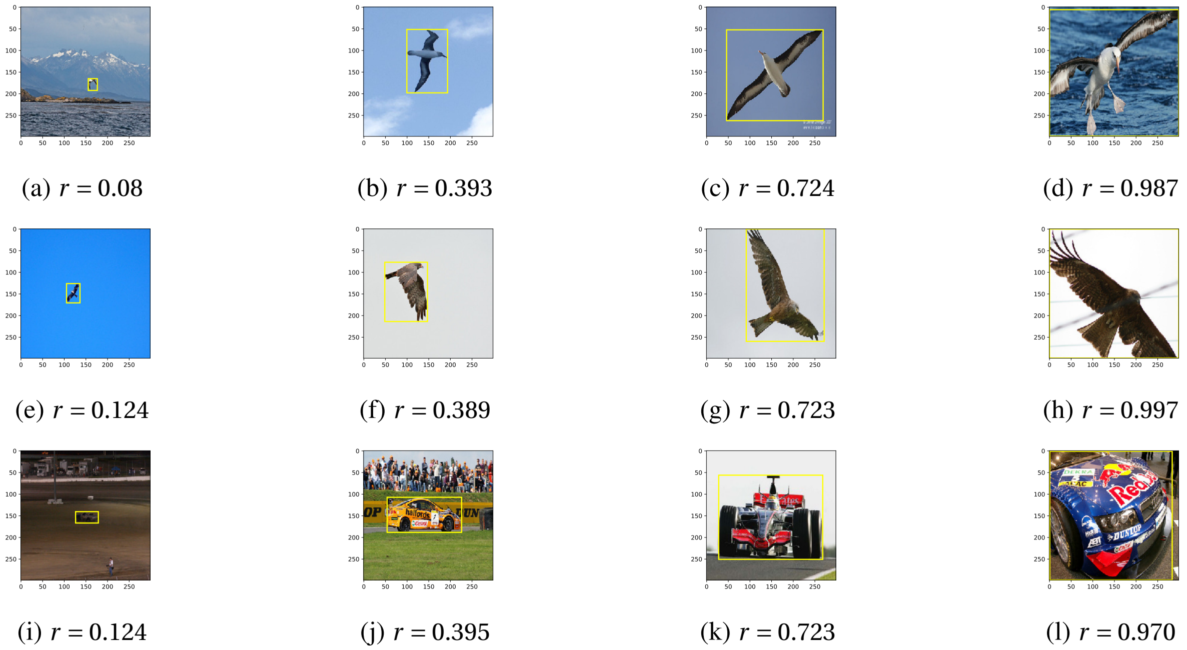
Examples of images used in the analysis are illustrated in Fig.2. These three classes were selected from the ImageNet so that images contain a single object and these objects either share similarities across classes (two types of birds), or are fundamentally different (birds and racing cars).

4 Alignment of Scale Information Along Individual Dimensions

In this section, we consider training and test images of the scale regression of a single class (*albatross*) as in [4].

4.1 Measuring Alignment

In this section, we evaluate whether the scale information is encoded by individual feature maps or as complex combinations of several feature maps. The alignment thus refers to scale values varying along a single feature map. After training the regression model, the regression vector is normalized: $\hat{\mathbf{v}} = \frac{\mathbf{v}}{|\mathbf{v}|}$. We then compute the



(a) $r = 0.08$ (b) $r = 0.393$ (c) $r = 0.724$ (d) $r = 0.987$

(e) $r = 0.124$ (f) $r = 0.389$ (g) $r = 0.723$ (h) $r = 0.997$

(i) $r = 0.124$ (j) $r = 0.395$ (k) $r = 0.723$ (l) $r = 0.970$

Figure 2: Examples of images and their respective scale ratio $r = \sqrt{\frac{h_b \times w_b}{h_i \times w_i}}$. Top row: *albatross* class; middle row *kite bird* class and bottom row: *racing car* class.

norm of the vector when retaining only a portion d' of its d dimensions sorted by their value $\hat{v}_i$. In this paper, we evaluate the layer *mixed8* of InceptionV3 which has $d = 1280$ feature maps. $|\hat{\mathbf{v}}_{1 \dots d'}| = 1$ means that all the scale information captured by the regression model is contained in the corresponding d' feature maps. This measure reflects how many feature maps (dimensions) are necessary to regress the scale. The first value, for a single coefficient, is $max(\hat{\mathbf{v}})$, which represents the largest alignment with an individual dimension. In Fig. 3 ('Non-reg' orange line), we report this contribution of dimensions in the regression of scale. We remind here that a dimension represents the spatial aggregation (average) of a deep response map.

With the Lasso regression (Fig. 3 'Lasso reg.' blue line) the regression vector aligns with few dimensions (40 out of 1280) while obtaining good generalization to new data. This means that the scale information can be represented by only a small portion of the response maps.

To complete this analysis of alignment along individual dimensions, we report the prediction performance of the Lasso scale regression for different values of α (see Eq. (2)), together with the number of non-zero coefficients in Fig. 4. The coefficient of determination R^2 is averaged across 10 runs (for each run, different splits are randomly drawn for the training and test sets). The results show that a good prediction of scale is obtained with a few dimensions (feature maps) retained by the Lasso regression. Even when drastically reducing the number of dimensions up to 1% of the original 1280, a good scale prediction is maintained with $R^2 > 0.6$. This compares well to the maximum value of 0.874 when using all dimensions. With 70 dimensions, no significant performance drop is observed as compared to this value. These results suggest that a combination of 1% of the features linearly encodes the scale information.

Yosinski et al. [6] showed by examples of activations that important features are learned in hidden layers and are encoded by individual neurons. It is generally referred to as local, as opposed to a representation distributed across multiple neurons. Our results are in line with previous work that showed the encoding of complex concepts distributed across response maps [7, 20].

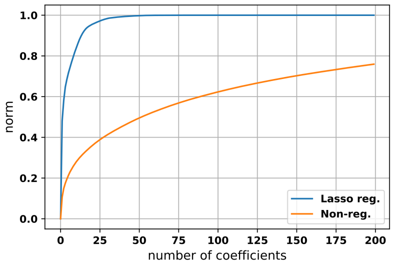


Figure 3: Analysis of the contribution of response maps for linear regression of scale. The norm $|\hat{\mathbf{v}}_{1 \cdots d'}|$ of the normalized regression vector is reported for an increasing number d' of dimensions (sorted by decreasing contribution $\hat{v}_i$). We limit this analysis to the 200 largest values of $\hat{v}_i$. The non-regularized regression predicts the scale of the test data with a coefficient of determination $R^2 = 0.874$. The regularized Lasso regression is set to $\alpha = 0.001$. Only 40 coefficients out of 1280 are non-zero (~3%) and the scale is predicted on the test data with $R^2 = 0.839$.

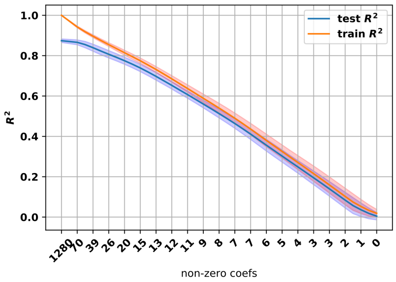


Figure 4: Evaluation of the scale prediction (R^2) of Lasso regression with different values of α. The model is trained and evaluated on images of the *albatross* class. The numbers of non-zero coefficients are reported on the x-axis instead of the corresponding α values (ranging from 0 to 0.01). The results are averaged across 10 runs and the 95% confidence intervals are reported. For comparison, we also report the results on the training data.

5 Directional Consistency of Scale Regression Across the Data Manifold

The data (ImageNet images here) lie on a complex manifold both at the input and hidden activations levels. The directions found with the scale regression lie in this space of (pooled) activations. Images from different classes can be far apart in this space and the question arises of whether learning a scale regression in one neighborhood (of a given class for instance) will generalize to the rest of the manifold. In other words, can we find a direction that is constantly equivariant with the scale information across the manifold? To empirically address this question, we adopt two complementary approaches. First, the generalization of a regression model trained on images from a given class will be evaluated on images from a different class. Second, the regression coefficients obtained from different classes will be compared (angle and cosine similarity between vectors).

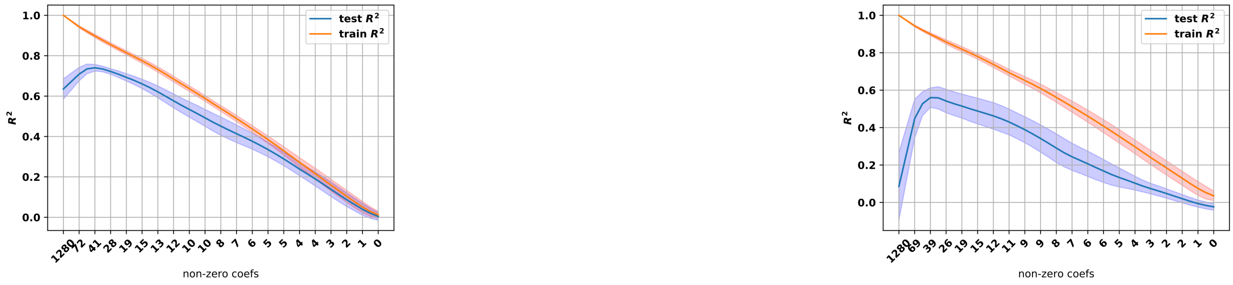
A point that we have not mentioned yet is that the scale information does not come only from the averaging operation on response maps (see Section 3). Spatially large objects (i.e. large ratio r) cover a large region of the response maps thus resulting in larger averaged values that could be regressed. Note that this is one of the reasons why we use deep features (*mixed8* layer) with large effective receptive fields [21]. If the scale information, however, was solely contained in the size of the region covered by the object in the response maps, the regression would not generalize to new classes as different neurons activate for different objects or parts. We will show that the scale regression is, to some extent, similarly encoded for very different classes, suggesting that some neurons specifically encode scale information regardless of the object type. We can also analyze activations at the center of the object instead of a spatial average to get rid of the information contained in the region covered by the object in the response maps. In practice, we found this difficult to implement as the center of the object can vary substantially within the bounding box. Besides this, the effective receptive field, depending on the depth and input image, can be larger or smaller than the object of interest.

5.1 Predicting Scale in Unseen Classes

In this section, we evaluate the generalizability of a scale regression to images from unseen classes (unseen in the regression training phase). The regression model trained on images of a single class is evaluated on unseen images coming from a different class using the R^2. These results are compared with those obtained on images of the same class (results in Fig. 4) as reported in Table 1. We report the results for $\alpha = 0.001$ (for a

Table 1: Comparison of scale prediction within and across classes. The scale regression is trained with Lasso ($\alpha = 0.001$) on the training class and evaluated on held-out data of the same or a different class. Average accuracy across 10 runs and standard deviation are reported.

Training class	testing class	R^2 non-reg.	R^2 Lasso reg.
albatross	albatross	$0.843_{\pm 0.012\%}$	$0.839_{\pm 0.023\%}$
albatross	kite	$0.672_{\pm 0.076\%}$	$0.745_{\pm 0.042\%}$
albatross	car	$0.086_{\pm 0.240\%}$	$0.560_{\pm 0.069\%}$



(a) test class: *kite bird*

(b) test class: *racing car*

Figure 5: Evaluation of the scale prediction (R^2) of Lasso regression with different values of α. The model is trained on the *albatross* class and evaluated on (a) *kite bird* class and (b) *racing car* class. The numbers of non-zero coefficients are reported on the x-axis instead of the corresponding α values (ranging from 0 to 0.01). The results are averaged across 10 runs and the 95% confidence intervals are reported. For comparison, we also report the results on the training data.

good compromise between dimensionality reduction and performance), averaged across 10 runs. As mentioned previously, we train on the *albatross* class with 220 images and test on one class relatively similar (*kite birds*, ID: n01608432, 220 images) and one radically different (*racing cars*, ID: n04037443 220 images).

In Fig. 5, we report the scale prediction performance in different classes for varying values of α in the Lasso regression (see Eq. (2)). Unlike the prediction of images of the same class (Fig. 4), the best results are obtained with Lasso regression when it retains approximately 3% of the regression coefficients (39 out of 1280). The regression without regularization overfits the training class and does not generalize to new types of objects.

Table 1 and Fig. 5 show that the prediction of scale for images of a different type than the regression training data is relatively accurate, yet lower than on the same class ($R^2 = 0.745$ and 0.560 vs $R^2 = 0.839$). The prediction is also better for similar classes (two types of birds) than radically different classes (bird and car). It suggests some encoding of scale information relatively consistent across the data manifold, with some encoding more specific to each specific class or group of classes.

5.2 Comparing Regression Coefficients

In this section, we compare regression coefficients between multiple regression models to understand whether the scale encoding is the same for very different images (i.e. of different classes). For comparison of the regression vectors, we use the angle and cosine similarity between pairs of vectors. Both measures evaluate the alignment of two vectors, although the cosine similarity takes into account their magnitude while the angle does not. The angle, expressed in degrees, is bounded in $[0, 90]$. The cosine similarity is bounded in $[0, 1]$ and is 1 for an angle of 0°. To obtain a baseline, we first train and compare two regression models on images from the same class. To evaluate the generalization, we then train and compare models on images from different classes. In Table 2, we report the angle between the coefficient vectors of different models as well as the cosine similarity, i.e. for two vectors $\mathbf{v_1}$ and $\mathbf{v_2}$, $\frac{\mathbf{v_1} \cdot \mathbf{v_2}}{\|\mathbf{v_1}\| \|\mathbf{v_2}\|}$. The angle and cosine similarity are approximately 90° and 0 respectively when we regress randomly shuffled ratios. In a high-dimensional space, vectors are likely

Table 2: Comparison of regression models trained on several classes using angle and cosine similarity between vectors in the 1280-d space.

		non-reg.		Lasso	
Train	test	angle	cos.	angle	cos.
albatross	albatross	67.7°	0.410	46.2°	0.66
albatross	kite	72.4°	0.315	64.1°	0.414
albatross	car	86.7°	0.058	76.9°	0.227

to be orthogonal (angle 90° and cosine similarity of 0). The number of "almost-orthogonal" vectors grows exponentially with the dimension of the space (here $d = 1280$). Therefore, even an angle slightly below 90° and a cosine similarity slightly larger than zero can still reflect similarities in these high-dimensional vectors. These results further support the hypothesis of scale encoding being common across the data manifold, with some local encoding specific to different types of objects.

6 Conclusion

This paper proposed an experimental evaluation of scale equivariance inside CNNs. We built on top of our previous research using bounding-box-to-image ratios to train and evaluate regression models in deep activations. As an extension, we first showed that the scale information is encoded as a combination of a few response maps. Indeed, a good scale prediction was obtained when retaining less than 3% of the feature maps. A single response map, however, is not sufficient to encode scale information. We showed that the concept of scale is distributed across multiple response maps. As a second main result, we showed that scale information is encoded in a relatively consistent way across the data manifold. By learning a scale regression on a set of images from a given class, we can infer the scale of images from a completely different class. This result explains the generalization to scale regression in histopathology images from ImageNet pre-trained models in [5]. A limitation of our analysis is that we have only considered a linear regression. Besides, we considered only three classes for this exploratory work with a limited number of images.

With a similar approach, this analysis can be performed for other transformations (e.g. rotation) as well as binary or continuous measures (e.g. presence of a specific object part, first or second-order statistics) of the input images to understand how CNNs learn to detect and encode various types of information. Understanding the internal behavior of CNNs, "opening the black-box", is important to build intuition both for researchers designing or applying new models and end-users who lack understanding of the networks' behaviors. The benefits of interpreting deep representations are multiple. Understanding the encoding of scale can help, for instance, to debug deep models, to modify how scale is compressed and to compare its representation across different architectures. The analysis equivariance to transformations is important to ensure that the network preserves important information (not discarding scale if relevant), while evaluating the redundancy of the representation can be used, for instance, for pruning or compression purposes and ensuring generalization to new data.

Acknowledgment

Work supported by the SNSF grant 205320_179069 and the Horizon 2020 project PROCESS grant 777533.

References

[1] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.

[2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[3] A. Depeursinge, V. Andrearczyk, P. Whybra, J. van Griethuysen, H. Müller, R. Schaer, M. Vallières, and A. Zwanenburg, "Standardised convolutional filtering for radiomics," *preprint arXiv:2006.05470*, 2020.

[4] T. Lompech, M. Graziani, A. Depeursinge, and V. Andrearczyk, "On the scale invariance in state of the art CNNs trained on ImageNet," in *(submitted)*, 2020.

[5] M. Graziani, T. Lompech, A. Depeursinge, and V. Andrearczyk, "Interpretable CNN pruning for preserving scale-covariant features in medical imaging," in *iMIMIC at MICCAI*, 2020.

[6] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson, "Understanding neural networks through deep visualization," in *Deep Learning workshop at ICML*, 2015.

[7] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba, "Network dissection: Quantifying interpretability of deep visual representations," in *Computer Vision and Pattern Recognition*, 2017, pp. 6541–6549.

[8] M. Aubry and B. C. Russell, "Understanding deep features with computer-generated imagery," in *International Conference on Computer Vision*, 2015, pp. 2875–2883.

[9] K. Lenc and A. Vedaldi, "Understanding image representations by measuring their equivariance and equivalence," in *Conference on Computer Vision and Pattern Recognition*, 2015, pp. 991–999.

[10] A. Azulay and Y. Weiss, "Why do deep convolutional networks generalize so poorly to small image transformations?" *Journal of Machine Learning Research*, vol. 20, no. 184, pp. 1–25, 2019.

[11] D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow, "Interpretable transformations with encoder-decoder networks," in *IEEE International Conference on Computer Vision*, 2017.

[12] Z. C. Lipton, "The mythos of model interpretability," *Queue*, vol. 16, no. 3, pp. 30:31–30:57, 2018.

[13] G. Alain and Y. Bengio, "Understanding intermediate layers using linear classifier probes," in *International Conference on Learning Representations 2017 Workshop*, 2016.

[14] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, and R. Sayres, "Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV)," in *ICLR*, 2018.

[15] M. Graziani, V. Andrearczyk, and H. Müller, "Regression concept vectors for bidirectional explanations in histopathology," in *iMIMIC at MICCAI*, 2018, pp. 124–132.

[16] M. Graziani, V. Andrearczyk, S. Marchand-Maillet, and H. Müller, "Concept attribution: Explaining cnn decisions to physicians," *Computers in Biology and Medicine*, p. 103865, 2020.

[17] C. J. Cai, E. Reif, N. Hegde, J. Hipp, B. Kim, D. Smilkov, M. Wattenberg, F. Viegas, G. S. Corrado, M. C. Stumpe *et al.*, "Human-centered tools for coping with imperfect algorithms during medical decision-making," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019.

[18] M. Graziani, H. Müller, and V. Andrearczyk, "Interpreting intentionally flawed models with linear probes," in *SDL-CV workshop at the IEEE International Conference on Computer Vision*, 2019.

[19] H. Touvron, A. Vedaldi, M. Douze, and H. Jégou, "Fixing the train-test resolution discrepancy," in *Advances in Neural Information Processing Systems*, 2019.

[20] B. Zhou, Y. Sun, D. Bau, and A. Torralba, "Interpretable basis decomposition for visual explanation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 119–134.

[21] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," in *Advances in neural information processing systems*, 2016, pp. 4898–4906.