

Learning Deep Features for Discriminative Localization

关于全连接层不能保持spatial information的理解

相比全连接层，卷积层是一个spatial-operation，能够保持物体的空间信息(translation-variant)。比如一个物体原来在左上角，卷积之后的结果feature-map在左上角的激活值大。如果这个物体移动到右下角，那么卷积之后的feature-map同样会在右下角的激活值比较大。但是对于全连接层来说，它是将feature-map所有位置的信息综合之后输出，和物体的具体位置在哪里无关，比如一张图，人在左上角和右下角得到的fc层的输出应该是一致的（因为后面就接softmax分类了）global average pooling(gap)不仅仅是一个regularizer，还能够将卷积层的定位能力一直保持到最后一层。即使这个网络是训练来进行分类的，我们也可以在feature map上获取那些对于分类具有区分性(discriminative)的区域。比如对于一张分类成自行车的图片来说，feature map上面在车轮子，车把这样地方的激活值就会比较大。而且这种网络的训练是end-to-end的，只需要训练classification的网络，我们就可以在forward的时候获取localization的信息

Inspiration

- Convolutional neural networks (CNNs) actually behave as object detectors despite no supervision on the location of the object was provided
- Global average pooling which acts as a structural regularizer, preventing overfitting during training

Class Activation Mapping

- Identify the importance of the image regions by projecting back the weights of the output layer on to the convolutional feature maps
- Before the final output layer (softmax in the case of categorization), we perform **global average pooling** on the convolutional feature maps and use those as features for a fully-connected layer that produces the desired output (categorical or otherwise).

Procedure



$f_k(x, y)$ represent the activation of unit k in the last convolutional layer at spatial location

(x, y) . Then, for unit k , the result of performing global average pooling,

$$F_k = \sum_{x,y} f_k(x, y)$$

w_k^c is the weight corresponding to class c for unit k .

P_c is given by $\frac{\exp(S_c)}{\sum_c \exp(S_c)}$ Here we ignore the bias term:

$$\begin{aligned} S_c &= \sum_k w_k^c \sum_{x,y} f_k(x, y) \\ &= \sum_{x,y} \sum_k w_k^c f_k(x, y). \end{aligned} \quad (1)$$

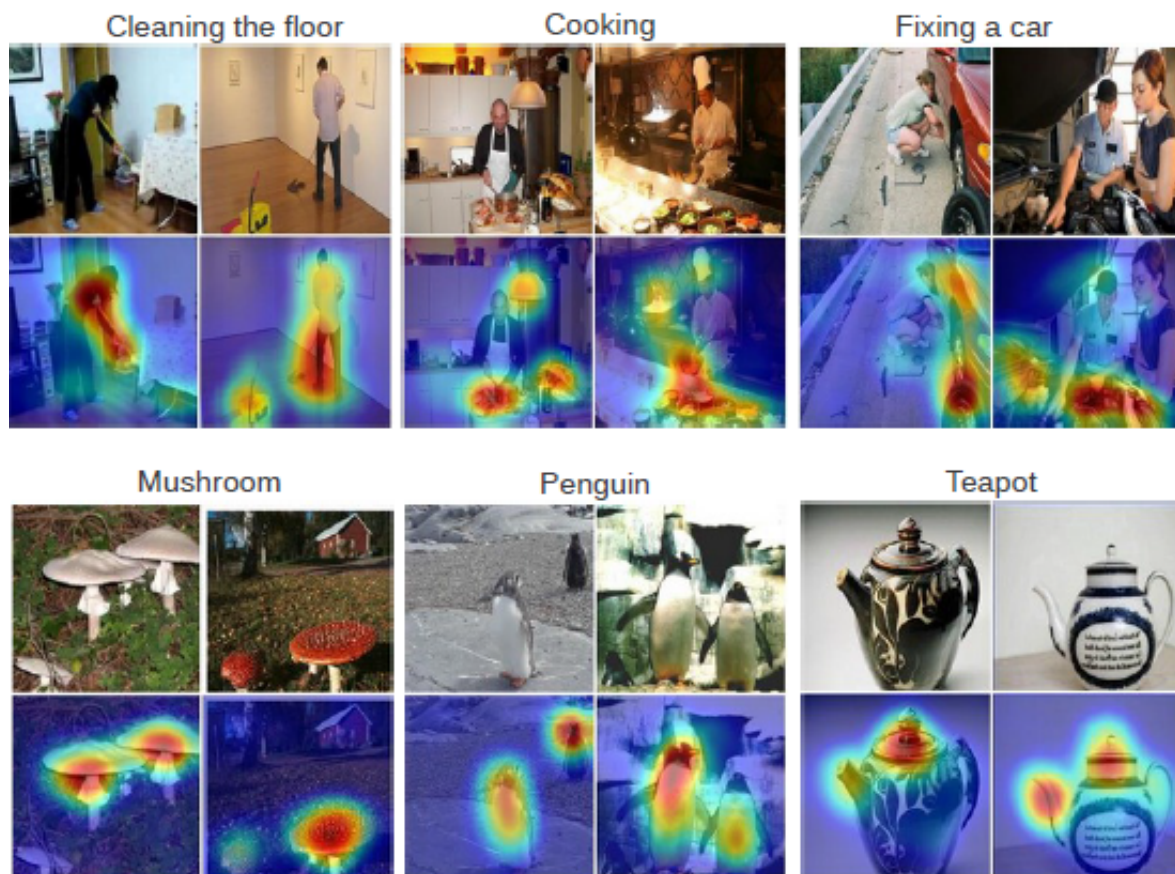
We define M_c as the class activation map for class c , where each spatial element is given by

$$M_c(x, y) = \sum_k w_k^c f_k(x, y). \quad (2)$$

Hence $M_c(x, y)$ directly indicates the importance of the activation at spatial grid (x, y) leading to the classification of an image to class c .

By simply upsampling the class activation map to the size of the input image, we can identify the image regions most relevant to the particular category.

Demo

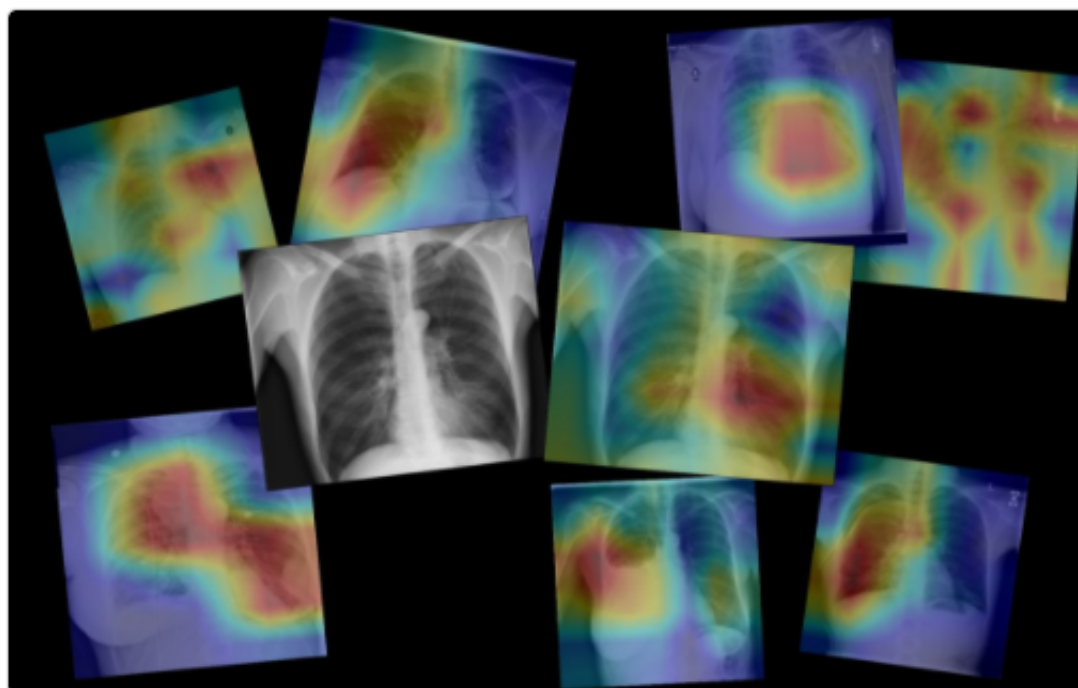


Andrew Ng @AndrewYNg · 11月16日

Our full paper on Deep Learning for pneumonia detection on Chest X-Rays.

@pranavrajpurkar @jeremy_irvin16 @mattlungrenMD arxiv.org/abs/1711.05225

翻译自英文





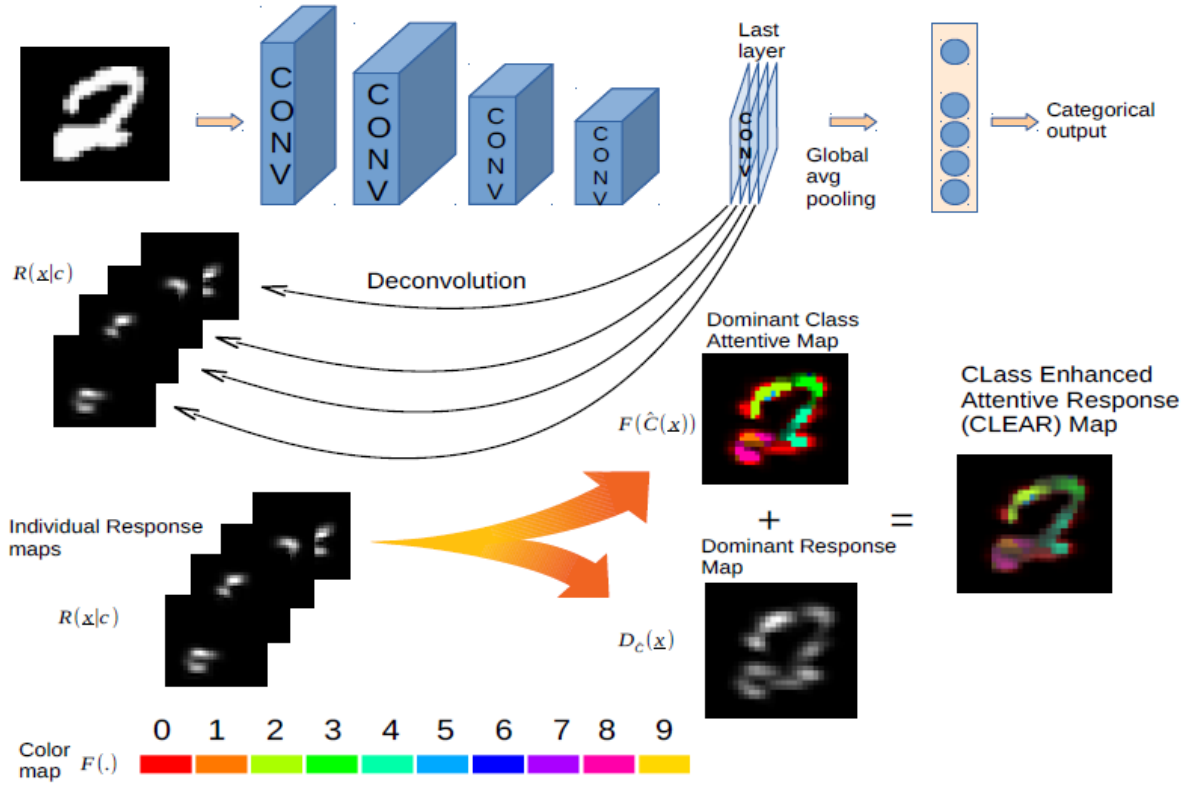
Stanford algorithm can diagnose pneumonia better than radiologists

Class Enhanced Attentive Response

contribution

- **The attentive region of interest in the image** responsible for the decision made by the DCNN
- **The attentive levels at these regions of interest** so that we understand their level of influence over the decision made by the DCNN
- **The dominant class associated with these attentive regions of interest** so that we can better understand **why** a decision was made

Procedure



A single layer of DNN: \hat{h}_l be the deconvolved output response of the single layer l with K kernel weights w . The deconvolution output response at layer l then can be then obtained by convolving each of the feature maps z_l with kernels w_l and summing them as:

$$\hat{h}_l = \sum_{k=1}^K z_{k,l} * w_{k,l}. \quad (1)$$

Here $*$ represents the convolution operation. For notational brevity, we can combine the convolution and summation operation for layer l into a single convolution matrix G_l . Hence the above equation can be denoted as: $\hat{h}_l = G_l z_l$

For multi-layered DCNNs, we can extend the above formulation by adding an additional un-pooling operation U . Thus, we can calculate the deconvolved output response from feature space to input space for any layer l in a multi-layer network as:

$$R_l = G_1 U_1 G_2 U_2 \dots G_{l-1} U_{l-1} G_l z_l \quad (2)$$

For CLEAR maps, we specifically calculate the output responses from individual kernels of the last layer of a network. Hence, given a network with last layer L containing

$K = N$ kernels, we can calculate the attentive response map; $R(\underline{x} | c)$ (where \underline{x} denotes the response back-projected to the input layer, and thus an array the same size as the input) for any class-specific kernel

$$c(1 \leq c \leq N)$$

in the last layer as:

$$R(\underline{x}|c) = |G_1 U_1 G_2 U_2 \dots G_{L-1} U_{L-1} G_L^c z_L|. \quad (3)$$

Given the set of individual attentive response maps, we then compute the dominant attentive class map, $\hat{C}(\underline{x})$, by finding the class at each pixel that maximizes the attentive response level, $R(\underline{x}|c)$, across all classe

$$\hat{C}(\underline{x}) = \underset{c}{\operatorname{argmax}} R(\underline{x}|c). \quad (4)$$













Given the dominant attentive class map, $\hat{C}(\underline{x})$, we can now compute the dominant attentive response map, $D^{\hat{C}}(\underline{x})$, by selecting the attentive response level at each pixel based on the identified dominant class, which can be expressed as follows:

$$D_{\hat{C}}(\underline{x}) = R(\underline{x}|\hat{C}). \quad (5)$$

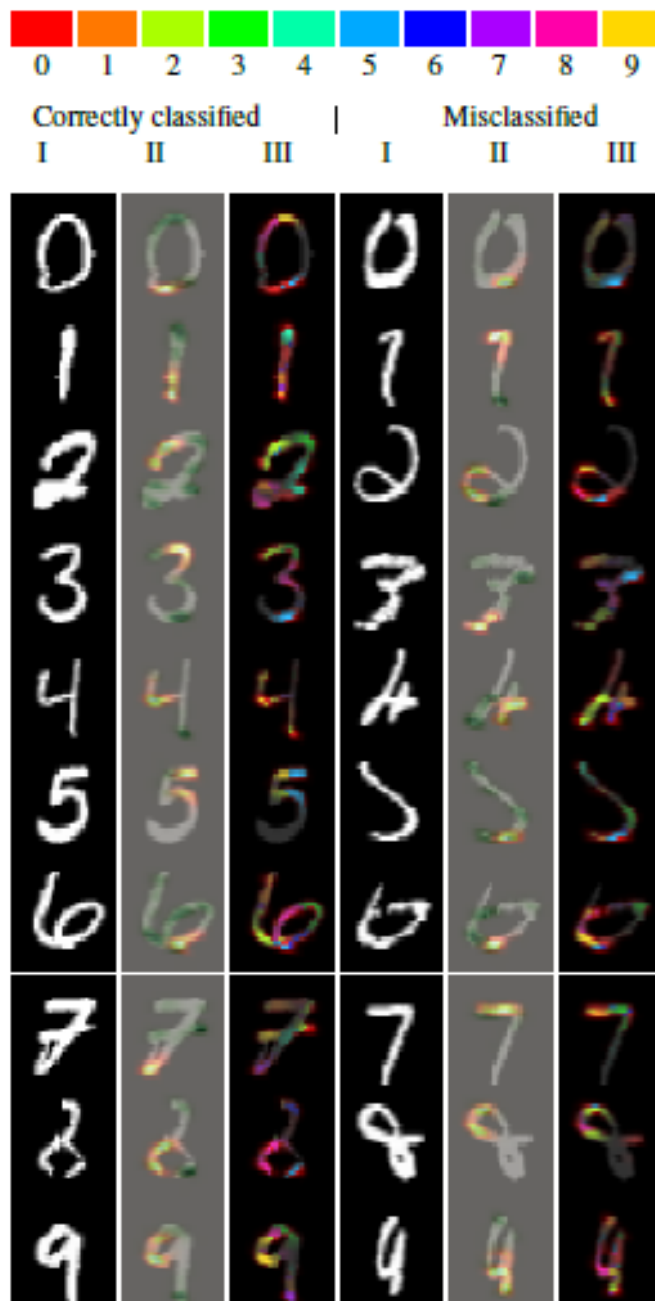
To form the final CLEAR map, we map the dominant class attentive map and the dominant attentive response map in the HSV color space as follows, then transform back into the RGB color space

$$\begin{aligned} H &= F(\hat{C}(\underline{x})), \\ S &= 1, \\ V &= D_{\hat{C}}(\underline{x}). \end{aligned} \quad (6)$$

Result in Mnist

Input	Output	Heatmap	Interpretation	CLEAR map	Interpretation
	 → 3 ✓		Focuses on right areas: Looks correct!		Major part of the positive focus is of 3
	 → 2 ✗		Focuses on wrong part, curve might be two; but why not 3 or 5 or 6?		Major part of the positive focus represents 2
	 → 3 ✗		Probably focuses on correct part, but why 3?		Major part of negative focus is 3; higher activation than any other class





ICLR2018 Interpretable Computer Aided Diagnosis of Diabetic Retinopathy

published:10/20/2017

Material

Kaggle diabetic retinopathy dataset five grades of diabetic retinopathy are as follows:
0: Negative, 1: Mild, 2: Moderate, 3: Severe, and 4: Proliferative. [^4]

Approach

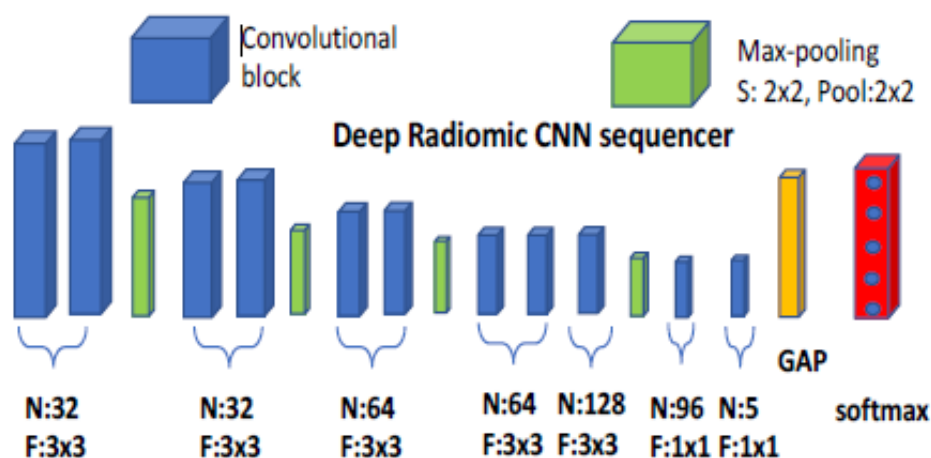
- Select retinal fundus images for one eye (right) only and performed an

automatic selective cropping to remove the background information. The use of a single eye led to 53,354 images in total.

- Divide the dataset into 90% and 10% of the dataset for training and testing respectively.
- Perform horizontal and vertical flipping along with channel-wise normalization for the whole dataset as data augmentation

Model

- Deep Radiomic CNN sequencer



Accuracy

73.2% overall

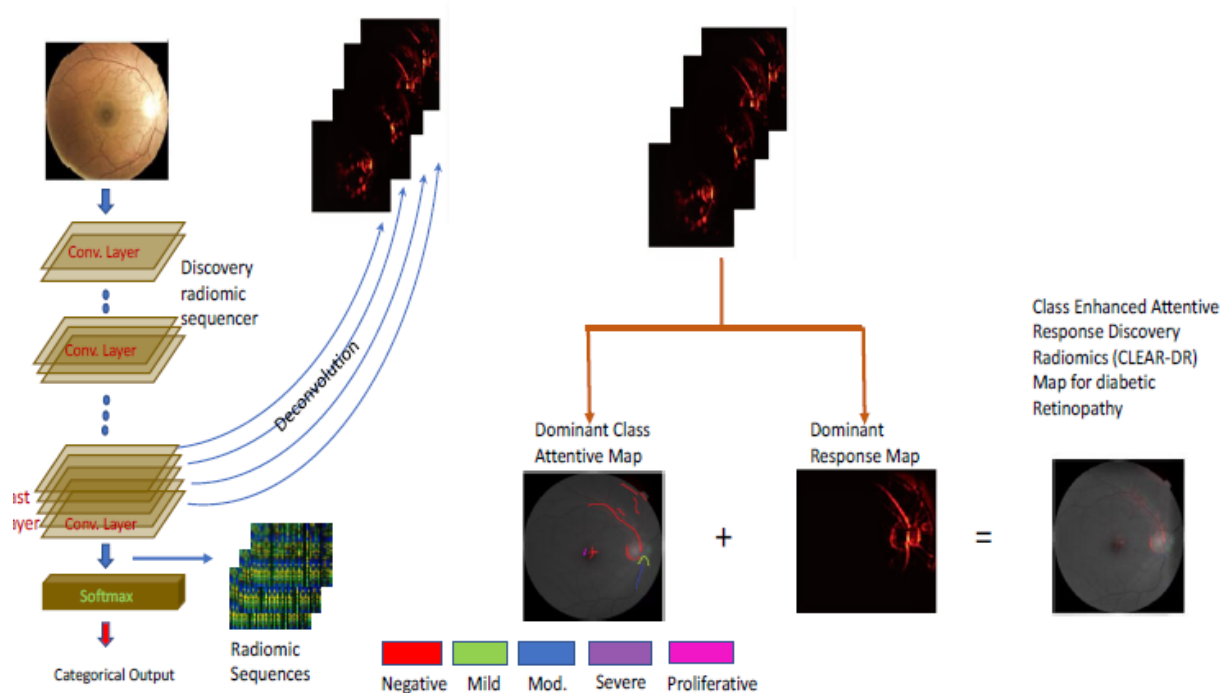
HighLight

Not only generates discriminating radiomic sequences for diabetic retinopathy grading **but also** provides a mechanism to **visually interpret its decision making process**

* individual attentive response maps are computed for each kernel associated with a grade by back-projecting activations from the output layer

- Compute two different types of maps
 - a. A dominant attentive response map, which shows the dominant attentive level for each location in the image;
 - b. A domainant grade involed in the desicion-making progress at each location

c. the dominant attentive response map and the dominant attentive grade map are combined visually by using color and intensity to produce the final CLEAR-DR map



Experiment

Correctly (a) and **Mis-classified** (b) examples for all diabetic retinopathy grades. Each color represents a single grade, as identified by the color map at the bottom of the figure. As well, the red box indicates the most attentive region used for grade prediction. It can be observed that the attentive regions used by the deep radiomic sequencer for **making correct decisions corresponds to medically relevant landmarks**, thus providing additional evidence for the proposed prediction. Best viewed in color and zoomed in.

