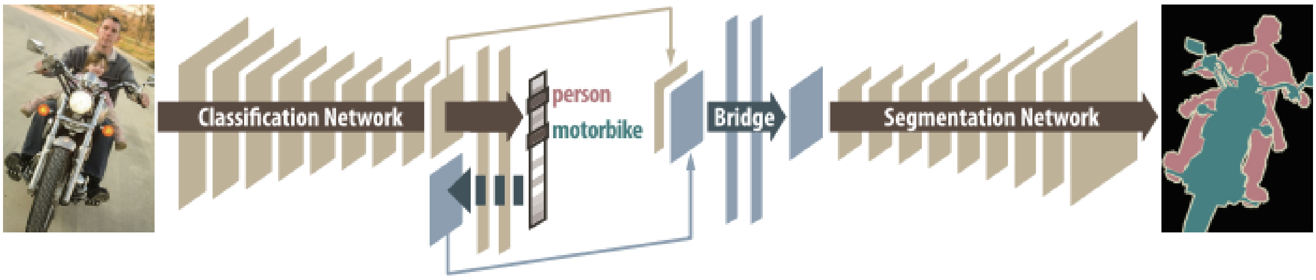
# DecoupledNet

**Decoupled Deep Neural Network for Semi-supervised Semantic Segmentation**

**NeurIPS 2015**

**227+ citations**

- Decouples classification and Segmentation
- First labels are identified by classification network and then binary segmentation is performed for each of the identified labels.
- Use of Heterogeneous annotations (strong annotations: segmentation masks and weak annotations: object class labels per image)
- Advantages:
  - Reduction in search space for segmentation by exploiting class-speicific activation maps.
- Common challenges in Semantic Segmentation:
  - Pose Variations.
  - Scale changes.
  - Occlusion
  - Background clutter
- **Semi and weakly-supervised approaches update the model of a supervised DNN by iteratively inferring and refining hypothetical segmentation labels.**
- These methods rely on ad-hoc procedures and there is no guarantee convergence; implementation may be tricky and results may be difficult to reproduce.
- **BRIDGING-LAYERS:**Deliver class-specific information; enable segmentation network to focus on the single label for segmentation (causing a reduction in search space.)
- **Note:**
  - Training is performed on each network separately. Classification is trained on image-level annotations and Segmentation is trained on pixel-wise annotations.
- Semi-Supervised learning bridges the gap between fully- and weakly-supervised learning approaches.
- Architecture:



  - **Classification Network:**

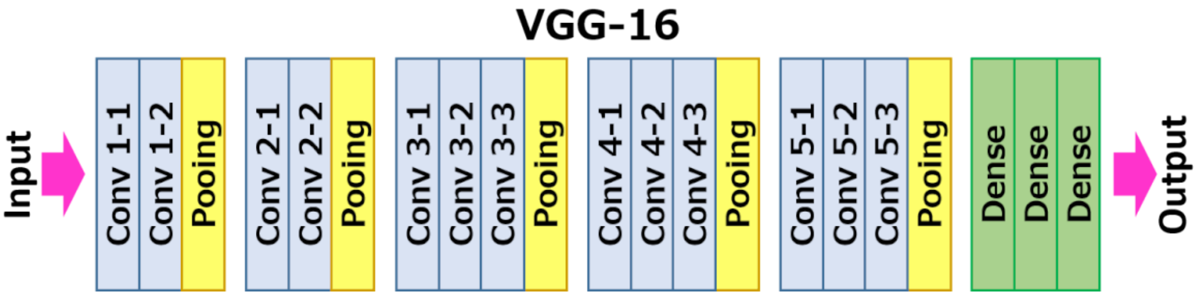
    - Takes an image $x$ as its input, and outputs a normalized score vector $S\left(\mathbf{x} ; \theta_c\right) \in \mathbf{R}^L$ representing scores of the input $x$ based on trained classification model $\theta_c$ for predefined $L$ categories.
    - The objective of the Classification Network is:

$$\min_{\theta_e} \sum_i e_c\left(\mathbf{y}_i, S\left(\mathbf{x}_i ; \theta_c\right)\right)$$

    - here $\mathbf{y}_i \in\{0,1\}^L$ denotes the ground-truth label vector of the $i$-th example; $e_c$ represents the error/loss.
    - VGG-16 is used for classifcation
    - The region in $x_i$ corresponding to each label $l \in \mathcal{L}_i$ is predicted by the Segmentation Network.
  - **Segmentation Network:**
    - Input: class-specific activation map $g_i^l$ of input image $x_i$, obtained from ***bridging layers***
    - Output: 2 channel class-specific segmentation map $M\left(\mathbf{g}_i^l ; \theta_s\right)$, $\theta_s$ is the model parameter for segmentation network. It has foreground channel and background channel represented as: $M_f\left(\mathbf{g}_i^l ; \theta_s\right)$ and $M_b\left(\mathbf{g}_i^l ; \theta_s\right)$ respectively.
    - Objective function:

$$\min_{\theta_s} \sum_i e_s\left(\mathbf{z}_i^l, M\left(\mathbf{g}_i^l ; \theta_s\right)\right)$$

    - It is formulated as per-pixel regression to ground-truth segmentation and minimized the above objective.
    - here $z_i^l$ represents the binary ground-truth segmentation mask for category $l$ of the $i$-th image $x_i$.
    - **Deconvolution network is used for segmentation, which uses a sequence of unpooling, deconvolution and rectification**.
    - Unpooling is implemented by importing the switch variable from every pooling layer in the classification network.
    - **SWITCH VARIABLE is nothing but the pooling indices of the pooling layer.**
    - The objective is to minimize the pixel-wise *binary* classification; it infers whether each pixel belongs to the given class $l$ or not.
    - Normally we try to classify each pixel to one of the $L$ predefined classes.
    - By decoupling classification and segmentation we have reduced the number of parameters to be optimized since it is a binary classification for segmenting each class label.
    - This decoupling is helpful since we have only 5-10 fully annotated images per class.
  - **Bridging Layers:**
    - To construct the class specific activation map $g_i^l$ for each identified label $l \in \mathcal{L}_i$.

- To encode spatial configuration of objects presented in image, we exploit outputs from and intermediate layer in the classification network.
- Outputs from the last pooling layer $pool5$ since the activation patterns of convolution and pooling layers often preserve spatial information effectively.
- Here we will represent the $pool5$ layer as $f_{spat}$.
- 

- Let $f^{(i)}$ be the output of the $i$-th layer ($i$=1,2,3..$M$) in the classification network. The relevance of activations in $f^{(i)}$ with respect to a specific class $l$ is computed by chain rule of partial derivative, which is similar to error back-propagation in optimization, as

$$\mathbf{f}^{l}_{\text{cls}} = \frac{\partial S_l}{\partial \mathbf{f}^{(k)}} = \frac{\partial \mathbf{f}^{(M)}}{\partial \mathbf{f}^{(M-1)}} \frac{\partial \mathbf{f}^{(M-1)}}{\partial \mathbf{f}^{(M-2)}} \cdots \frac{\partial \mathbf{f}^{(k+1)}}{\partial \mathbf{f}^{(k)}}$$

- where $f^{l}_{cls}$ denotes class-specific saliency map and $S_l$ is the classification score of class $l$.
- where $\mathbf{f}^{l}_{\text{cls}}$ denotes class-specific saliency map and $S_l$ is the classification score of class $l$.
- Intuitively, the above equation means that the values in $\mathbf{f}^{l}_{\text{cls}}$ depend on how much the activation in $f^{(k)}$ are relevant to class $l$; this is measured by computing the partial-derivative of class score $S_l$ with respect to the activations in $f^{(k)}$. We back-propagate the class-specific information until $pool5$ layer.
- **The class-specific activation map $g^l_i$ is obtained by combining both $f_{spat}$ and $f^l_{cls}$.**
- We first concatenate $\mathbf{f}_{\text{spat}}$ and $\mathbf{f}^{l}_{\text{cls}}$ in their channel direction, and forward-propagate it through the fully-connected bridging layers, which discover the optimal combination of $\mathbf{f}_{\text{spat}}$ and $\mathbf{f}^{l}_{\text{cls}}$ using the trained weights.
- The resultant class-specific activation map $g^l_i$ that contains both spatial and class-specific information is given to segmentation network to produce a class-specific segmentation map.
- **The changes in $g^l_i$ depend only on $f^l_{cls}$ since $f_{spat}$ is fixed for all classes in an input image.**

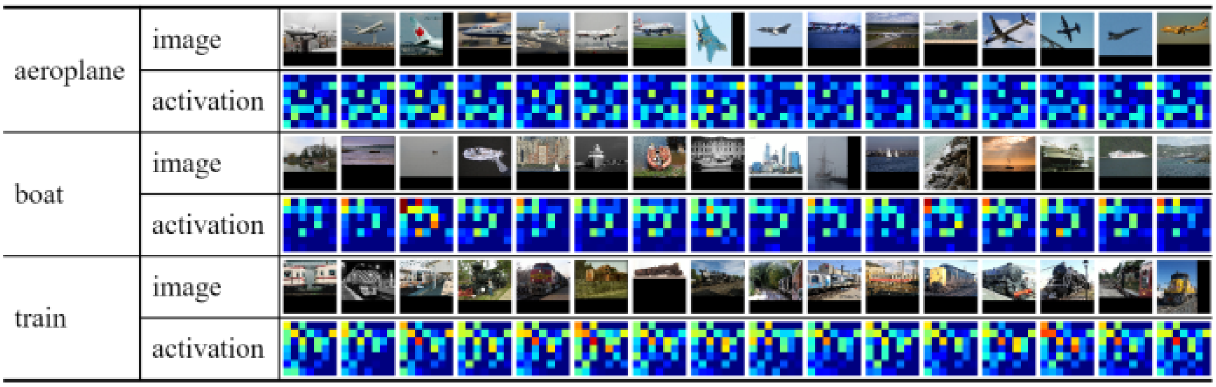
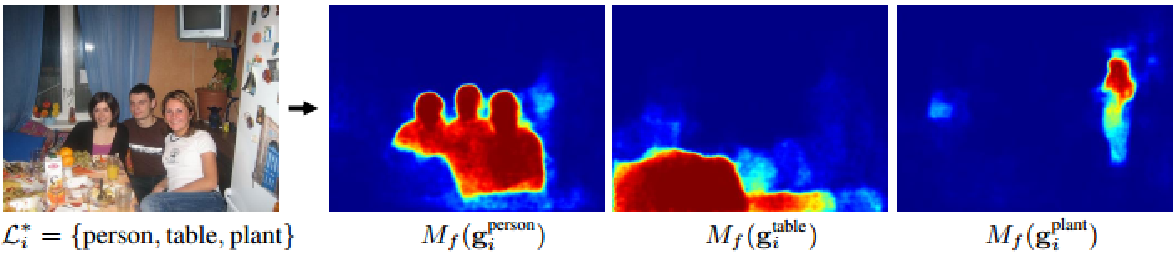


Figure 2: Examples of class-specific activation maps (output of bridging layers). We show the most representative channel for visualization. Despite significant variations in input images, the class-specific activation maps share similar properties.

- 
- The activations from the images in the same class share similar patterns despite substantial appearance variations, which shows that the output of bridging layers capture **class-specific informations** effectively.
- It also reduces the variations of input distributions for segmentation network, which allows to achieve good generaliation performance in segmentation even with a small number of training examples.

- **Inference:**
  - We compute a class-specific activation map $g_i^l$ for each identified label $l \in L_i$ and obtaion class-specific segmenation maps $\left\{M\left(\mathbf{g}_i^l;\theta_s\right)\right\}_{\forall l\in\mathcal{L}_i}$.
  - We also obtain $M\left(\mathbf{g}_i^*;\theta_s\right)$, where $g_i^*$ is the activation map from the bridging layers for all identified labels.
  - The final label estimation is given by identifying the label with the maximum score in each pixel out of $\left\{M_f\left(\mathbf{g}_i^l;\theta_s\right)\right\}_{\forall l\in\mathcal{L}_l}$ and $M_b\left(\mathbf{g}_i^*;\theta_s\right)$.


$\mathcal{L}_i^* = \{\text{person, table, plant}\}$
$M_f(\mathbf{g}_i^{\text{person}})$
$M_f(\mathbf{g}_i^{\text{table}})$
$M_f(\mathbf{g}_i^{\text{plant}})$

- Figure 3: Input image (left) and its segmentation maps (right) of individual classes.
- The above image illustrates the output segmentation map of each $g_i^l$ for $x_i$, where each map identifies high response area given $g_i^l$ successfully.