Seed, Expand and Constrain

Ирина Понамарева

Higher School of Economics

1 марта 2019 г.

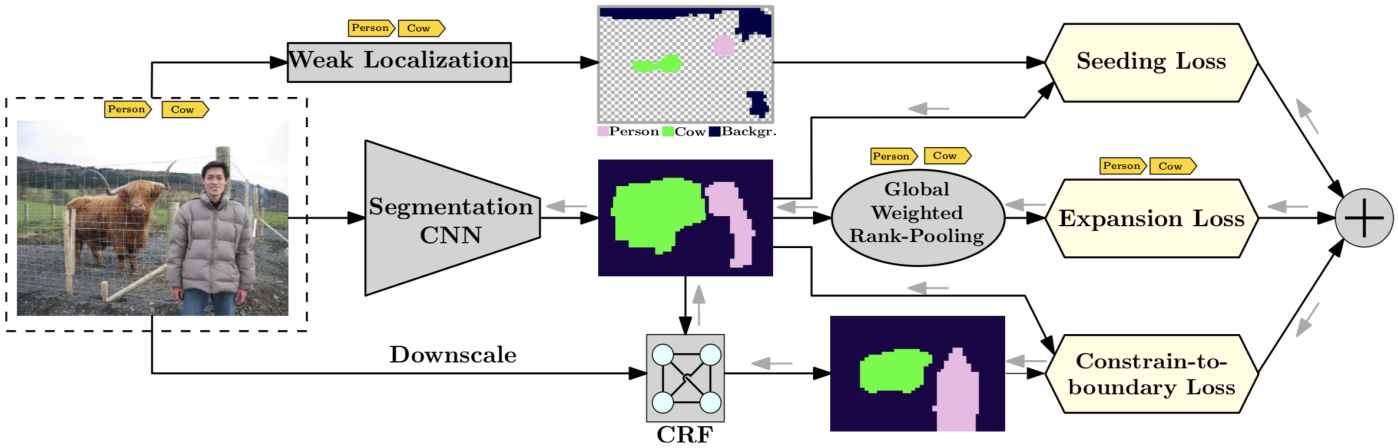
Overview

Формулировка задачи weakly-supervised segmentation

- $X \in \mathcal{X}$ — изображение
- $Y = (y_1, ..., y_n)$ — маска сегментации
- $y_i \in \mathcal{C} = \mathcal{C}' \cup \{c_{bg}\}$
- $\mathcal{D} = \{(X_i, T_i)\}_{i=1}^{N},\ X_i \in \mathcal{X},\ T_i \subset \mathcal{C}'$ — набор слабо аннотированных изображений.

Задача
$f(X, \theta)$ — deep CNN, моделирует
$f(X, \theta) = f_{u,c}(X, \theta) = p(y_u = c|X)$ — условную вероятность увидеть метку c на данной позиции

SEC loss



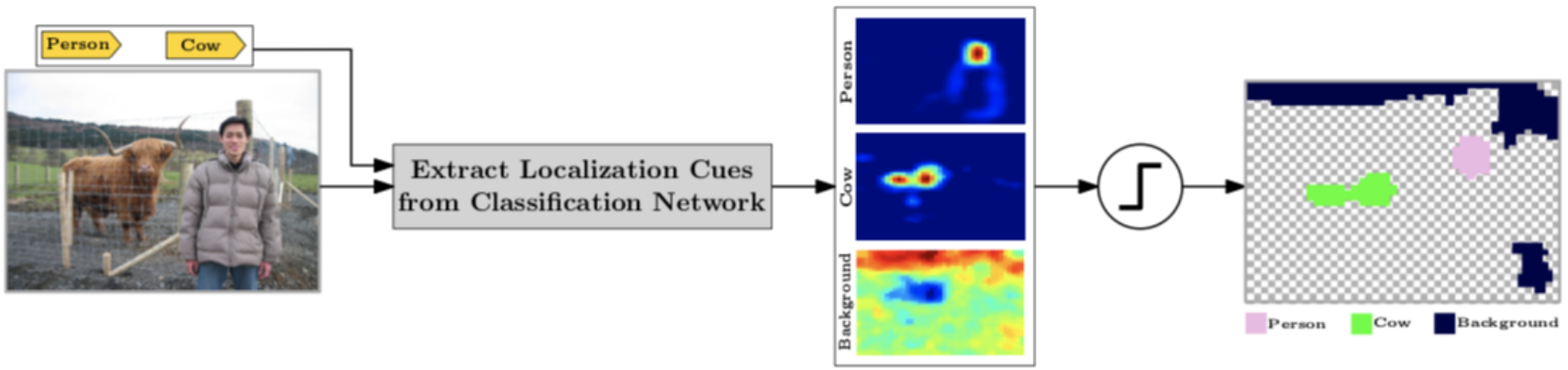
$$\min_{\theta} \sum_{(X,T)\in\mathcal{D}} \Big[L_{seed}(f(X;\theta), T) + L_{expand}(f(X;,\theta), T) + L_{constrain}(X, f(X;\theta)) \Big]$$

- L_{seed} — дает подсказки по локализации
- L_{expand} — штраф за слишом маленькие или неверные маски сегмантации
- $L_{constrain}$ — поощряет сегмантации, уважающие пространственную и цветовую структуру

Seeding loss

- Глубокие нейросети не умеют хорошо определять позицию, но могут давать подсказки по локализации
- Будем использовать эту информацию!
- S_c — набор точек, определенных как класс c слабой локализацией

$$L_{seed}(f(X), T, S_c) = -\frac{1}{\sum_{c \in T} |S_c|} \sum_{c \in T} \sum_{u \in S_c} \log f_{u,c}(X).$$



Expansion loss with global weighted rank pooling

- Idea: measure if a segmentation mask is consistent with the image-level labels
- Global average pooling or global max pooling?
- Global weighted rank pooling (GWRP)!

Expansion loss with global weighted rank pooling

- $I^c = \{i_1, ..., i_n\}$ — index set, i.e. $f_{i_1,c} \geq f_{i_2,c} \geq ... \geq f_{i_n,c}$
- $0 < d_c \leq 1$ — decay parameter. $d = 0$: GMP, $d = 1$: GAP

$$G_c(f(X); d_c) = \frac{1}{Z(d_c)} \sum_{j=1}^{n} (d_c)^{j-1} f_{i_j,c}(X),$$

$$Z(d_c) = \sum_{j=1}^{n} (d_c)^{j-1}$$

$$L_{expand}(f(X), T, S_c) = -\frac{1}{|T|} \sum_{c \in T} \log G_c(f(X); d_+)$$

$$-\frac{1}{\mathcal{C}' \setminus T} \sum_{c \in \mathcal{C}' \setminus T} \log(1 - G_c(f(X); d_-)) - \log G_{c^{bg}}(f(X); d_{bg})$$

Constrain-to-boundary loss

- Идея: уважать пространственную и цветовую структуру
- Конструируем полносвязный CRF $Q(X, f(X))$
- Ошибка — средняя дивергенция между выходом модели и CRF.

$$L_{constrain}(f(X), T, S_c) = \frac{1}{n} \sum_{u=1}^{n} \sum_{c \in \mathcal{C}} Q_{u,c}(X, f(X)) \log \frac{Q_{u,c}(X, f(X))}{f_{u,c}}$$

Inference at test stage

- mean Intersection over Union (mIoU)
- Сегментация: DeepLab-CRF-LargeFOV
- Локализация: finetuned VGG
- $d_- = 0$ (GMP), $d_+ = 0.996$, $d_0 = 0.999$

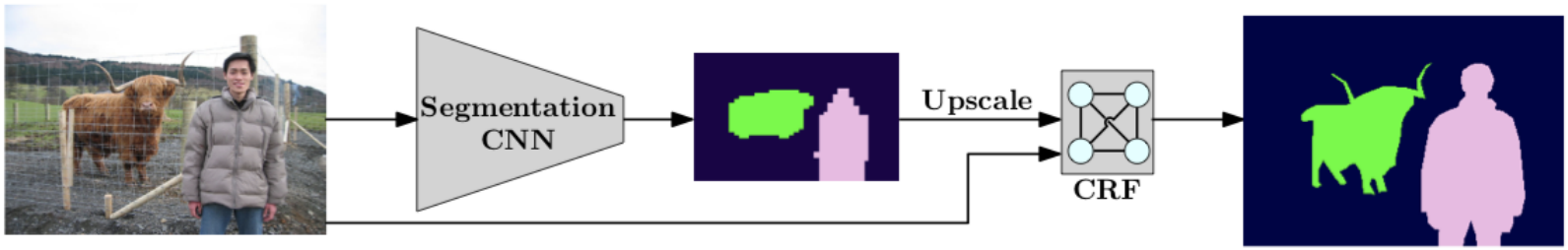
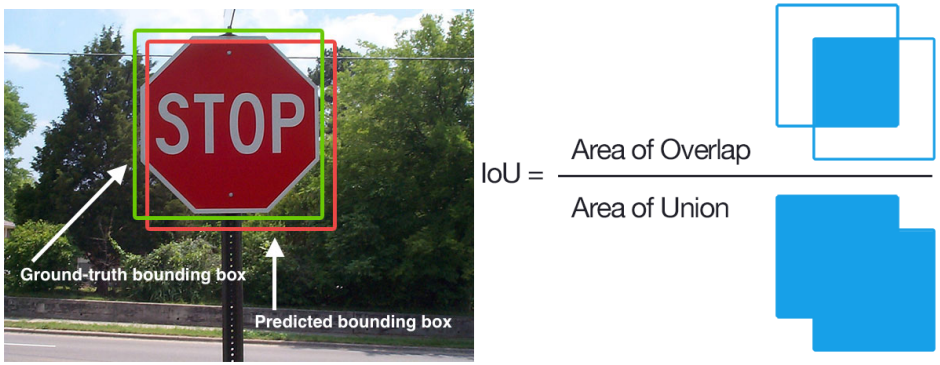
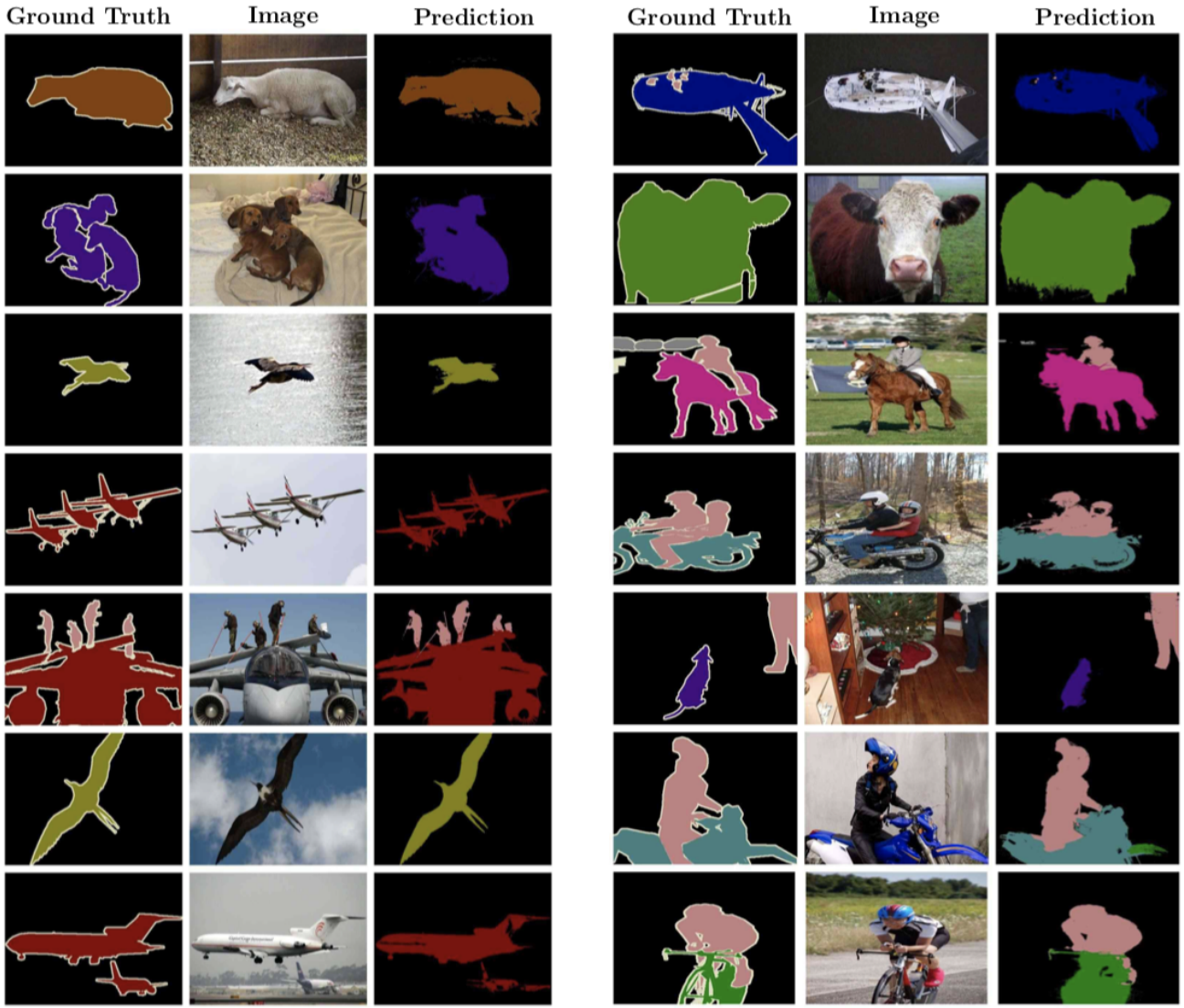


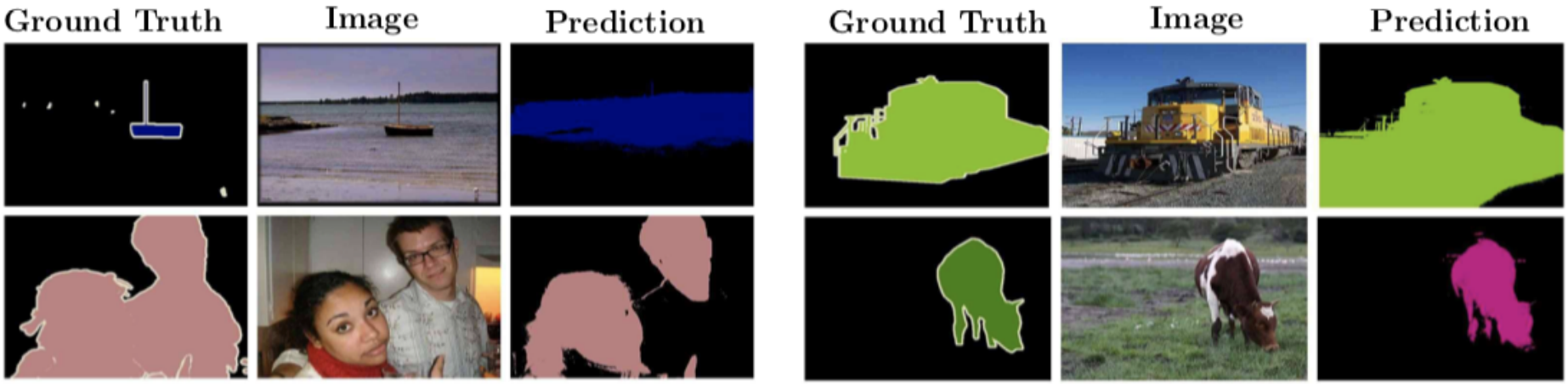
Fig. 3: The schematic illustration of our approach at test time.



Эксперименты: примеры удачных сегментаций



Эксперименты: ошибки модели



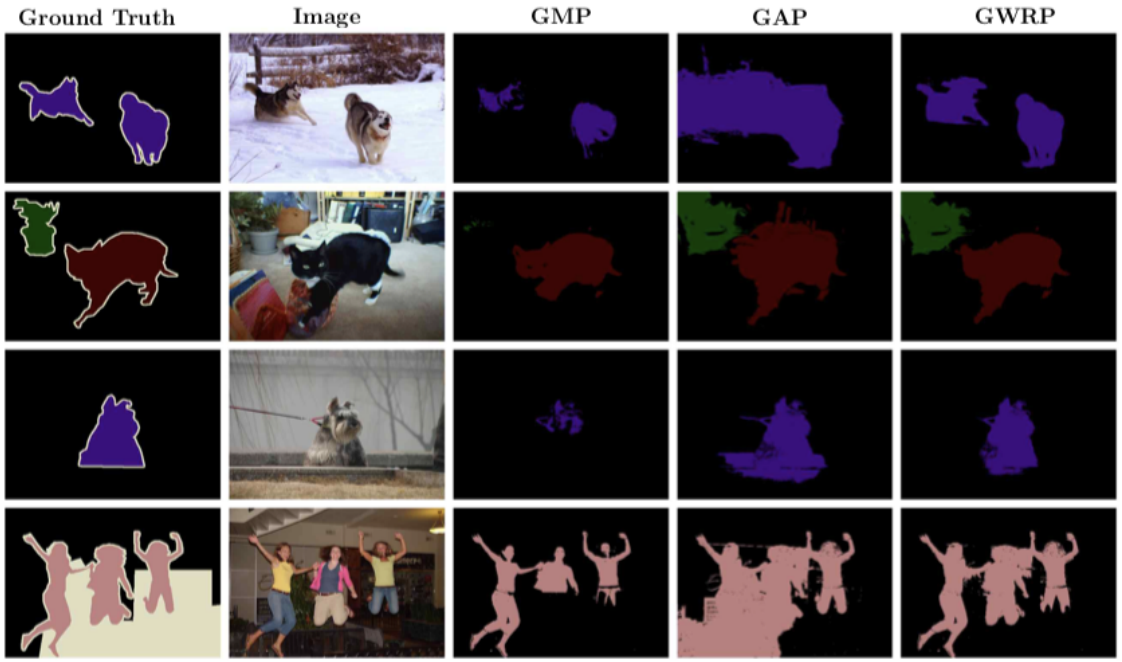
- Объект, постоянно присутствующий с определенным фоном
- Назван неверный класс (редко)
- Объект определен не полностью

Сравнение с другими моделями

PASCAL VOC 2012 *val* set	[4] (Img+Obj)	[14] (stage1)	EM-Adapt (re-impl. of [24])	CCNN [24]	MIL+ILP +SP-sppxl† [26]	SEC (proposed)	PASCAL VOC 2012 *test* set	MIL-FCN [25]	CCNN [24]	MIL+ILP +SP-sppxl† [26]	Region score pooling [18]	SEC (proposed)
background		71.7*	67.2	68.5	77.2	**82.4**	background		≈71‡	74.7	≈74‡	**83.5**
aeroplane		30.7*	29.2	25.5	37.3	**62.9**	aeroplane		24.2	38.8	33.1	**56.4**
bike		**30.5***	17.6	18.0	18.4	26.4	bike		19.9	19.8	21.7	**28.5**
bird		26.3*	28.6	25.4	25.4	**61.6**	bird		26.3	27.5	27.7	**64.1**
boat		20.0*	22.2	20.2	**28.2**	27.6	boat		18.6	21.7	17.7	**23.6**
bottle		24.2*	29.6	36.3	31.9	**38.1**	bottle		38.1	32.8	38.4	**46.5**
bus		39.2*	47.0	46.8	41.6	**66.6**	bus		51.7	40.0	55.8	**70.6**
car		33.7*	44.0	47.1	48.1	**62.7**	car		42.9	50.1	38.3	**58.5**
cat		50.2*	44.2	48.0	50.7	**75.2**	cat		48.2	47.1	57.9	**71.3**
chair		17.1*	14.6	15.8	12.7	**22.1**	chair		15.6	7.2	13.6	**23.2**
cow		29.7*	35.1	37.9	45.7	**53.5**	cow		37.2	44.8	37.4	**54.0**
diningtable		22.5*	24.9	21.0	14.6	**28.3**	diningtable		18.3	15.8	**29.2**	28.0
dog		41.3*	41.0	44.5	50.9	**65.8**	dog		43.0	49.4	43.9	**68.1**
horse		35.7*	34.8	34.5	44.1	**57.8**	horse		38.2	47.3	39.1	**62.1**
motorbike		43.0*	41.6	46.2	39.2	**62.3**	motorbike		52.2	36.6	52.4	**70.0**
person		36.0*	32.1	40.7	37.9	**52.5**	person		40.0	36.4	44.4	**55.0**
plant		29.0*	24.8	30.4	28.3	**32.5**	plant		33.8	24.3	30.2	**38.4**
sheep		34.9*	37.4	36.3	44.0	**62.6**	sheep		36.0	44.5	48.7	**58.0**
sofa		23.1*	24.0	22.2	19.6	**32.1**	sofa		21.6	21.0	26.4	**39.9**
train		33.2*	38.1	38.8	37.6	**45.4**	train		33.4	31.5	31.8	**38.4**
tv/monitor		33.2*	31.6	36.9	35.0	**45.3**	tv/monitor		38.3	41.3	36.3	**48.3**
average	32.2	33.6*	33.8	35.3	36.6	**50.7**	average	25.7	35.6	35.8	38.0	**51.7**

(*results from unpublished/not peer-reviewed manuscripts, †trained on ImageNet, ‡value inferred from average)

Discussion: выбор функции потерь и стратегии пуллинга

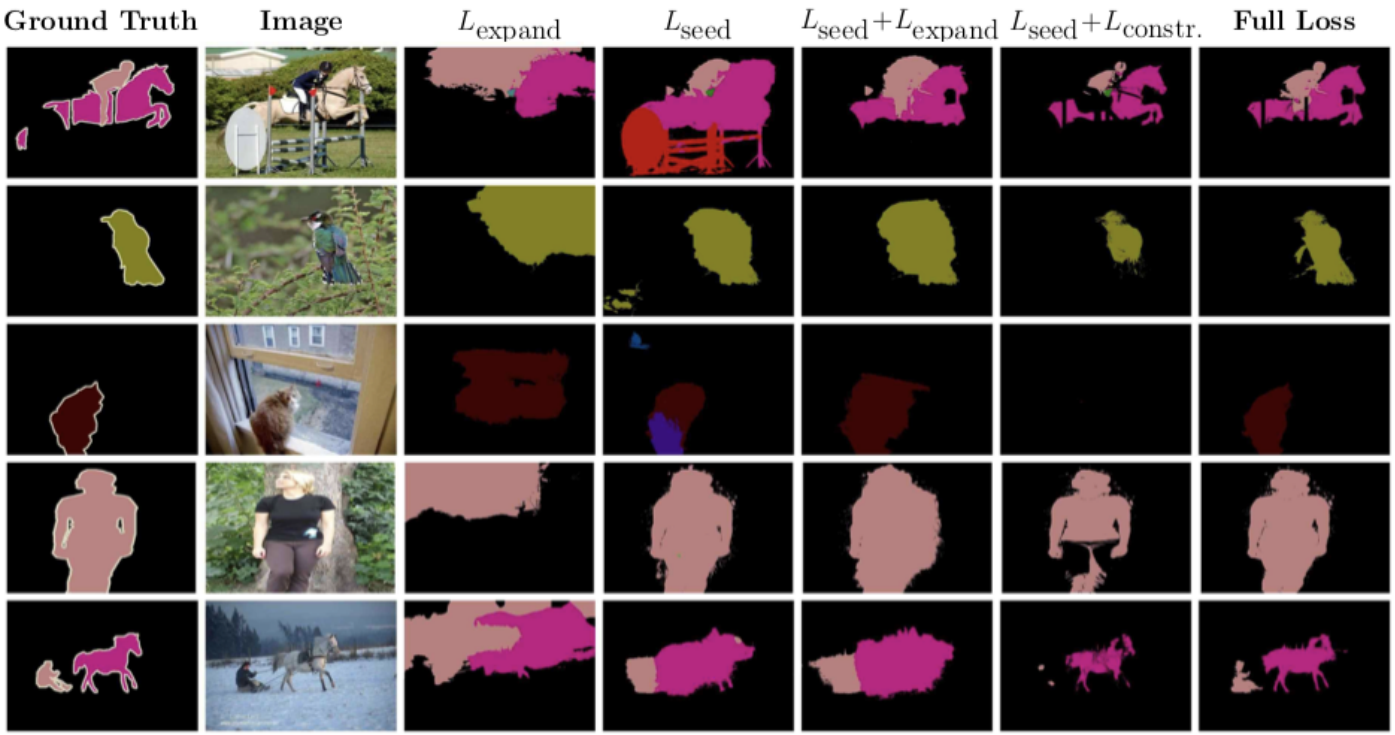


pooling method	fg fraction	mIoU *(val)*
GMP	20.4	46.5
GAP	35.6	45.7
GWRP	25.8	50.7
ground truth	26.7	–

- GMP недооценивает размер объектов
- GAP переоценивает размер объектов
- GWRP — то, что нужно!

Discussion: выбор функции потерь и стратегии пуллинга

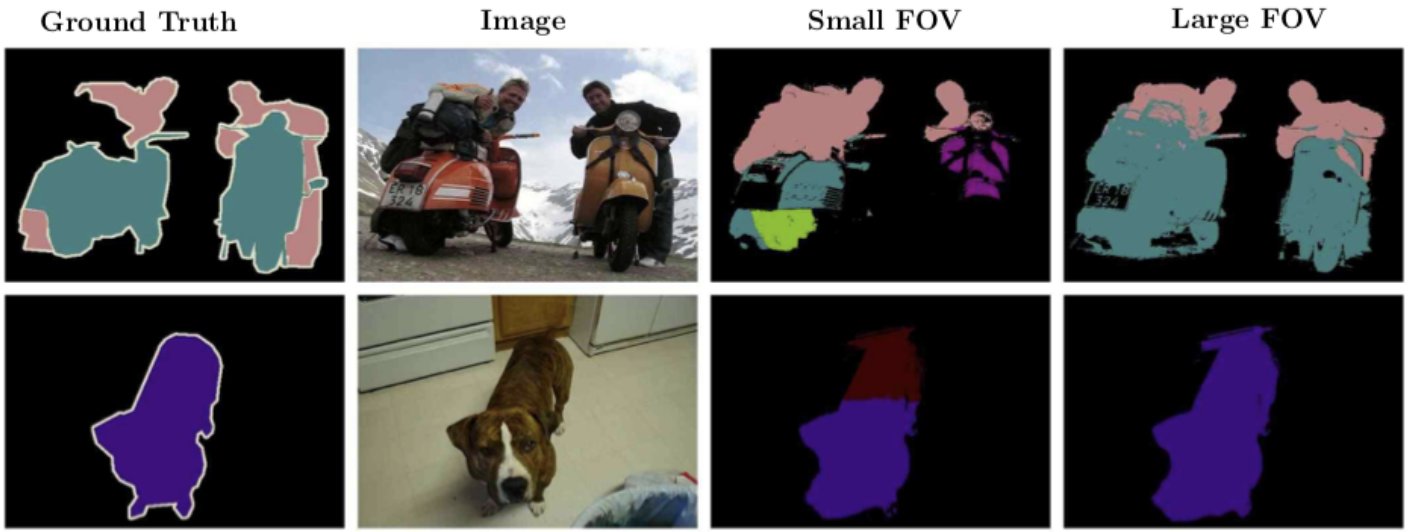
- Как разные компоненты функции потерь влияют на качество?
- Без seed все очень плохо
- Без expand сложно определить сложные объекты (люди в одежде...)
- Без constrain размер увеличен, но сложно найти границы



loss function	mIoU *(val)*
L_{expand}	20.6
L_{seed}	45.4
$L_{\text{seed}} + L_{\text{expand}}$	44.3
$L_{\text{seed}} + L_{\text{constrain}}$	50.4
all terms	50.7

Discussion: field of view

- Как field of view влияет на качество?
- Судя по всему, не стоит его уменьшать. Поэтому важно использовать seed!



field of view	mIoU *(val)*
211x211	38.1
378x378	50.7

Заключение

- Seed — правильное расположение сидов
- Expand — увеличить сиды до разумных пределов
- Constrain — учесть границы объектов

References

- **Kolesnikov, Alexander, and Christoph H. Lampert. "Seed, expand and constrain: Three principles for weakly-supervised image segmentation."European Conference on Computer Vision. Springer, Cham, 2016.**
- Pathak, Deepak, et al. "Fully convolutional multi-class multiple instance learning."arXiv preprint arXiv:1412.7144 (2014).
- He, Xuming, Richard S. Zemel, and Miguel Á. Carreira-Perpiñán. "Multiscale conditional random fields for image labeling."Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.. Vol. 2. IEEE, 2004.
- Krähenbühl, Philipp, and Vladlen Koltun. "Efficient inference in fully connected crfs with gaussian edge potentials."Advances in neural information processing systems. 2011.
- Pathak, Deepak, Philipp Krahenbuhl, and Trevor Darrell. "Constrained convolutional neural networks for weakly supervised segmentation."Proceedings of the IEEE international conference on computer vision. 2015.

The End

Additional slides: CRF

- What is Conditional Random Field?
- It models conditional probability of labels given images (posterior distribution as a Gibbs field)
- CRF (I, X) is characterized by Gibbs distribution

$$P(X, I) = \frac{1}{Z(I)} \exp\Big(- \sum_{c \in \mathcal{C}_{\mathcal{G}}} \phi_c(X_c|I)\Big), G - \text{graph on X}$$

- Two forms of feature functions:
  - state feature functions of a label l at a site i + image
  - transition feature functions of an image and a label at site i + neighboring site j

Additional slides: FCN CRF

$$P(X, I) = \frac{1}{Z(I)} \exp\Big(- \sum_{c \in \mathcal{C}_\mathcal{G}} \phi_c(X_c|I)\Big), G - \text{graph on X}$$

- Gibbs energy of labelling x:

$$E(x|I) = \sum_{c \in \mathcal{C}_\mathcal{G}} \phi_c(x_c|I)$$

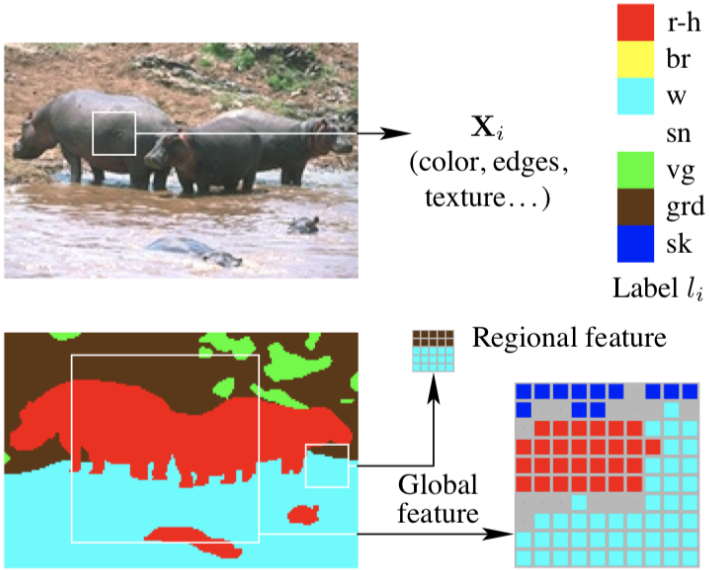
$$x^* = \arg\max_x P(x|I) \quad \textit{(MAP labelling)}$$

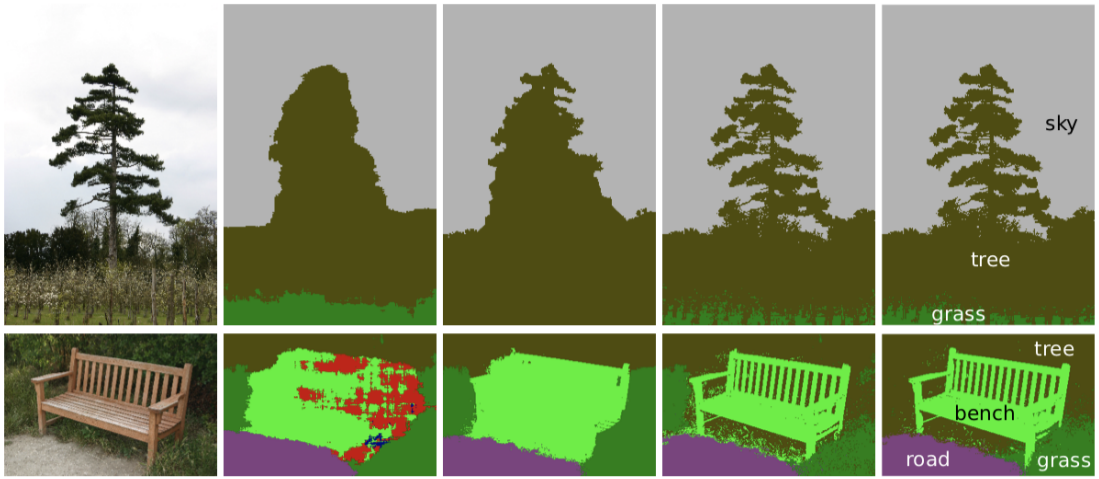
- In fully connected CRF $\mathcal{G}$ is a complete graph on X and $\mathcal{C}_\mathcal{G}$ is the set of all unary and pairwise cliques

$$E(x) = \sum_i \phi_u(x_i) + \sum_{i<j} \phi_p(x_i, x_j)$$

Additional slides: FCN CRF

- CRF: illustrations

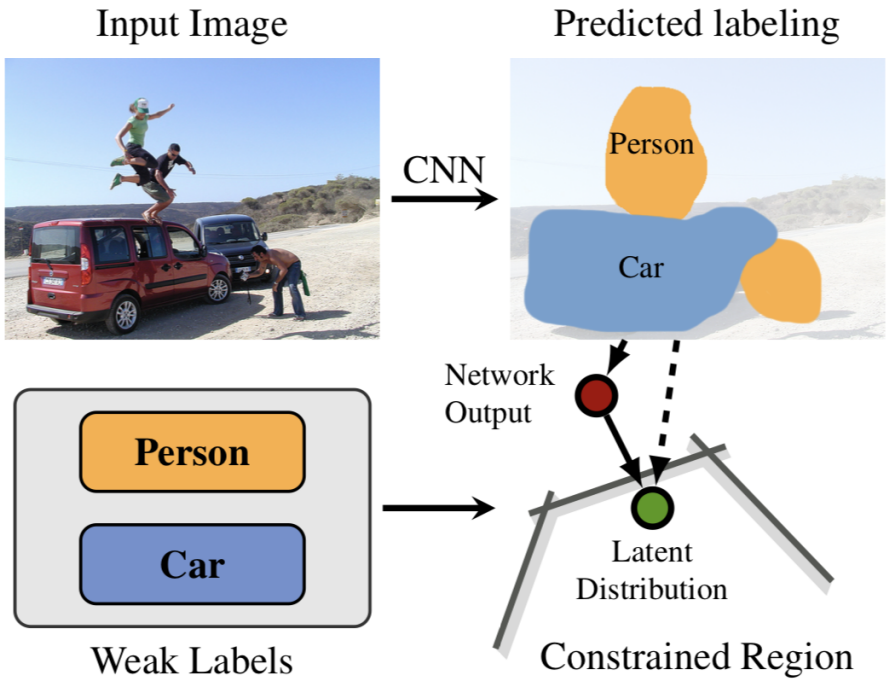


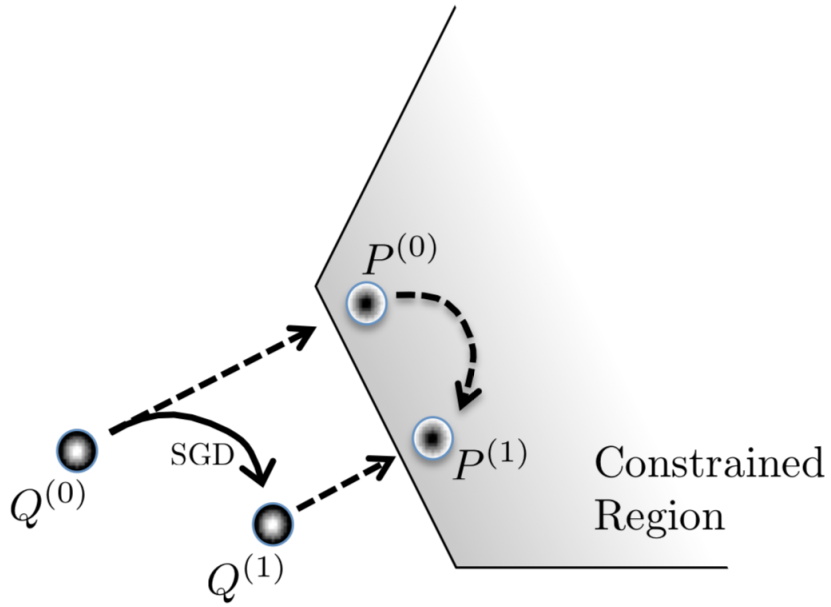


(a) Image (b) Unary classifiers (c) Robust P^n CRF (d) Fully connected CRF, MCMC inference, 36 hrs (e) Fully connected CRF, our approach, 0.2 seconds

Additional slides: Constrained CNN

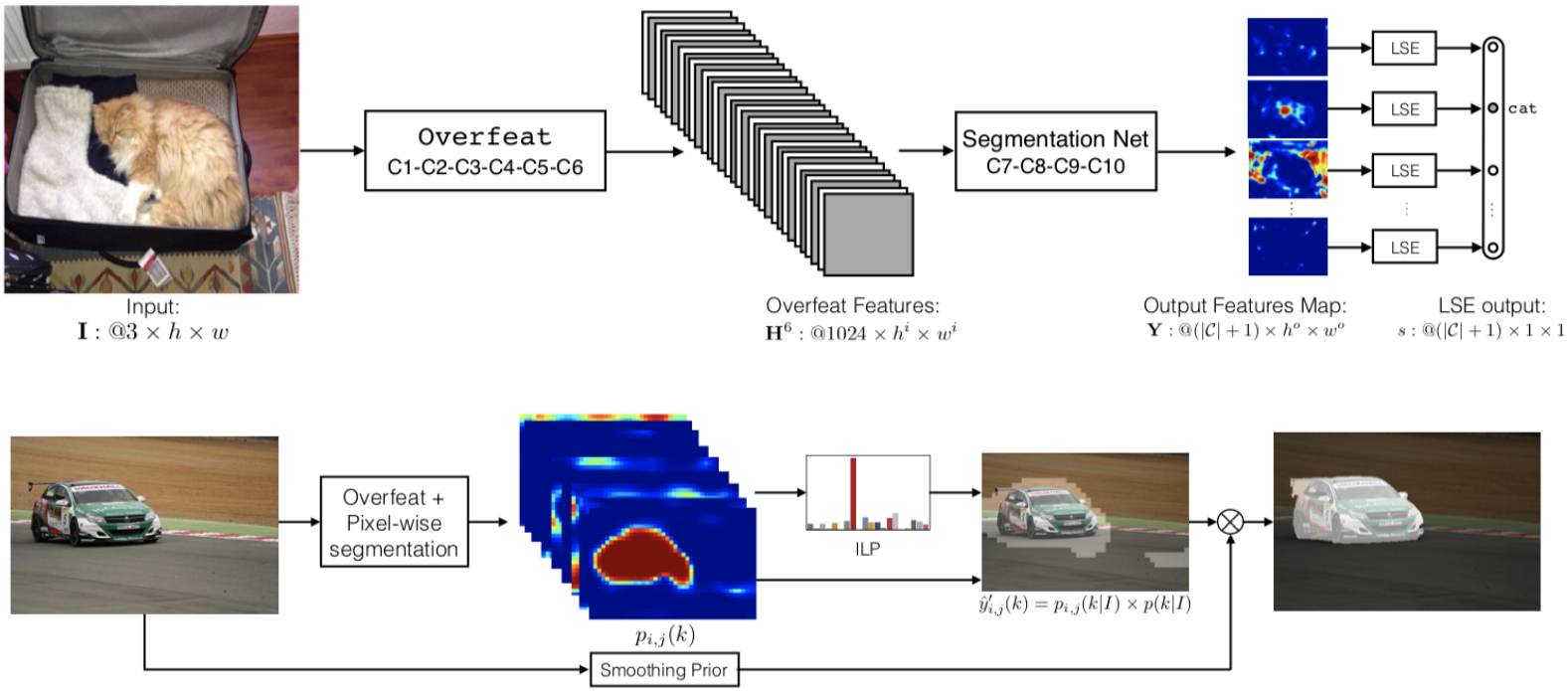
- Each image-level tag imposes constraints on the output labeling of a CNN classifier.
- In semantic segmentation, such constraints can describe the existence and expected distribution of labels from image-level tags.





Additional slides: MIL+ILP

- MIL+ILP (Multiple Instance Learning + Image-Level Prior)
- Use pixel-level labels, aggregate them to image-level ones
- Use simple post-processing technique: ILP and Smoothing Prior



Additional slides: Сравнение с другими моделями

Table 2: Summary results (mIoU %) for other methods on PASCAL VOC 2012. Note: the values in this table are not directly comparable to Table 1, as they were obtained under different experimental conditions.

method	*val*	*test*	comments
DeepLab [6]	67.6	70.3	fully supervised training
STC [42]	49.8*	51.2*	trained on Flickr
TransferNet [12]	52.1	51.2	trained on MS COCO; additional supervision: from segmentation mask of other classes
[4] (1Point)	42.7	–	additional supervision: 1 click per class
[4] (AllPoints-weighted)	43.4	–	additional supervision: 1 click per instance
[4] (squiggle)	49.1	–	additional supervision: 1 squiggle per class
EM-Adapt [23]	38.2	39.6	uses weak labels of multiple image crops
SN_B [41]	41.9	43.2	uses MCG region proposals (see text)
MIP+ILP+SP-seg [26]	42.0	40.6	trained on ImageNet, MCG proposals (see text)
MIL+ILP+SP-bb [26]	37.8	37.0	trained on ImageNet, BING proposals (see text)

(*results from manuscripts that are currently unpublished/not peer-reviewed)

Additional slides: Class Activation Maps

- We use a network architecture which largely consists of convolutional layers, and just 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). Given this simple connectivity structure, we can identify the importance of the image regions by **projecting back** the weights of the output layer on to the convolutional feature maps, a technique we call class activation mapping.

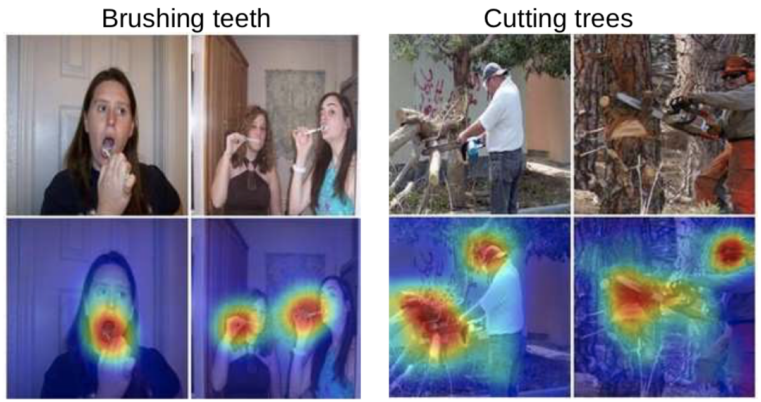


Figure 1. A simple modification of the global average pooling layer combined with our class activation mapping (CAM) technique allows the classification-trained CNN to both classify the image and localize class-specific image regions in a single forward-pass e.g., the toothbrush for *brushing teeth* and the chainsaw for *cutting trees*.

Additional slides: Class Activation Maps

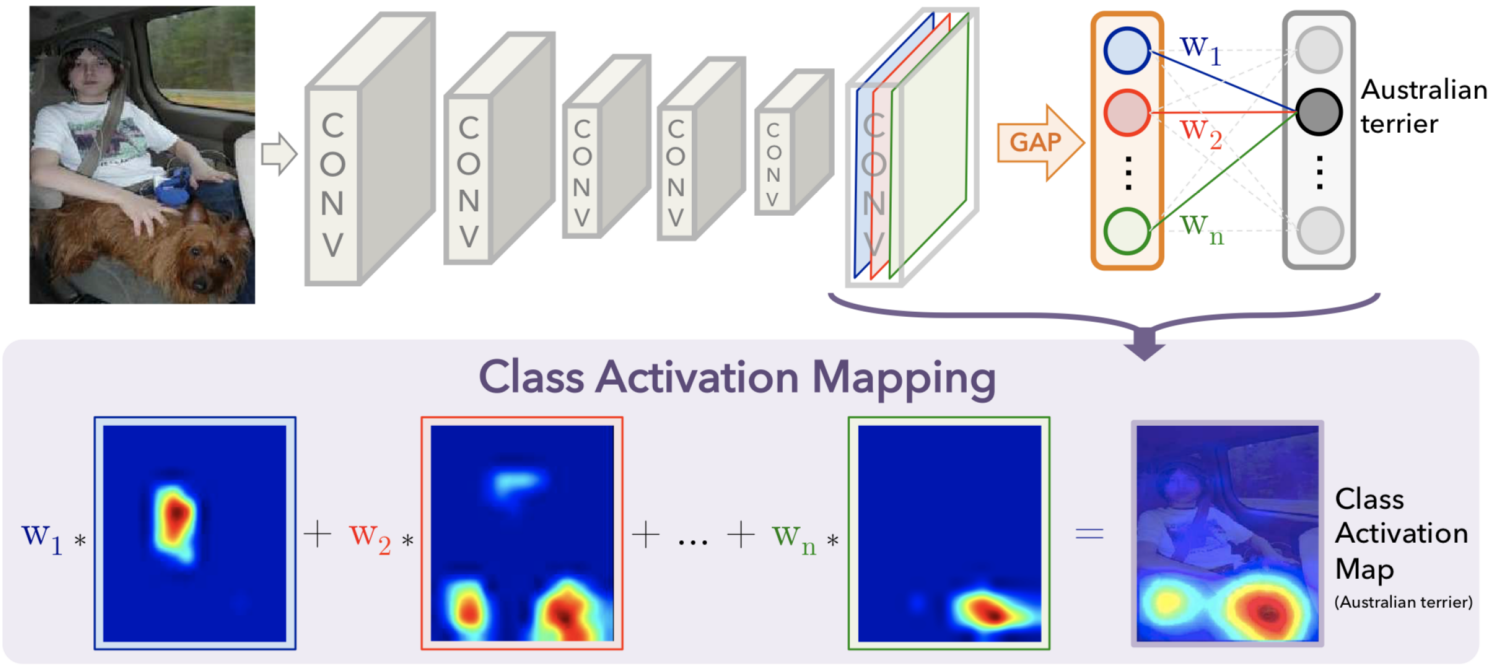
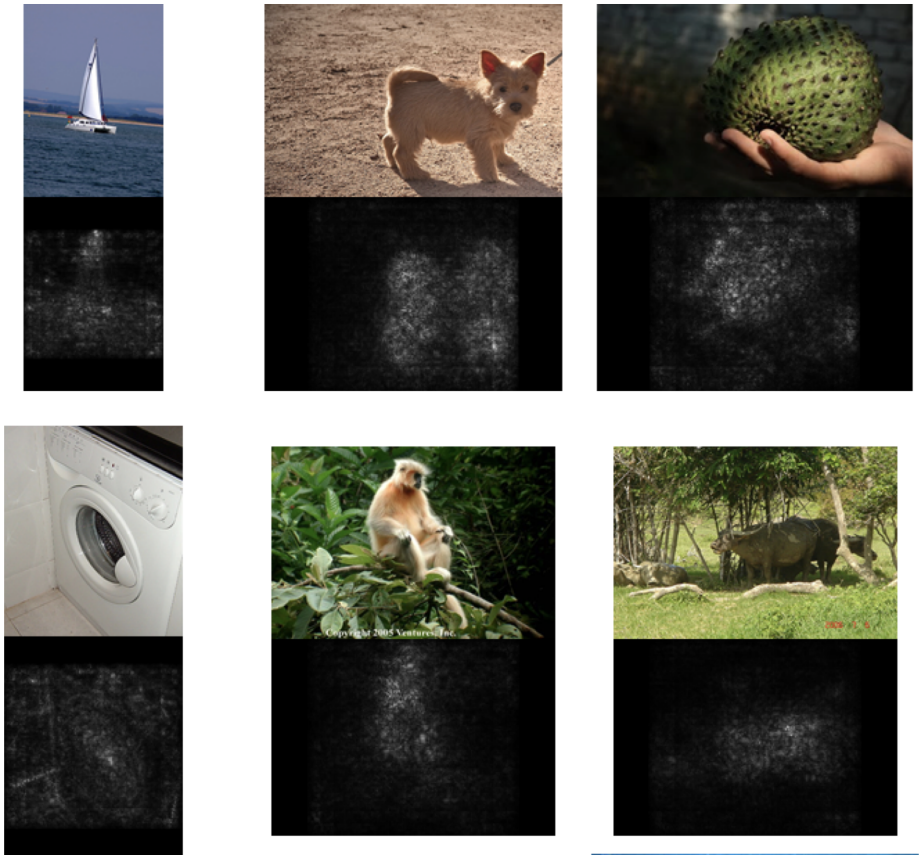


Figure 2. Class Activation Mapping: the predicted class score is mapped back to the previous convolutional layer to generate the class activation maps (CAMs). The CAM highlights the class-specific discriminative regions.

Additional slides: Saliency Maps



Additional slides: Summary других моделей

- **Graph-based models** infer labels for segments or superpixels based on their similarity within or between images
- Variants of **multiple instance learning** train with a per-image loss function, while internally maintaining a spatial representation of the image that can be used to produce segmentation masks
- Methods in the tradition of **self-training** train a fully- supervised model but create the necessary pixel-level annotation using the model itself in an EM-like procedure