Robust Object Detection Under Occlusion with Compositional Generative Networks.

Alan Yuille
Bloomberg Distinguished Professor
Depts. Cognitive Science and Computer Science
Johns Hopkins University

Motivation: We need tougher tests for Computer Vision Algorithms

- Standard Performance Measures (SPMs) test algorithms on finite-sized balanced annotated datasets (BAD). Half of the data is used for training and half for testing.
- ***But SPMs are problematic.*** Datasets are biased. They under-represent rare, but important, events. More generally, finite-sized datasets are unable to capture the combinatorial complexity of the real world.
- ***We need tougher tests that can quantify AI vision algorithms more accurately.*** Tougher tests include: (I) ***Out-of-DistributionTesting***, where the algorithms are tested from data that has different statistical properties than the training data. (II) ***Adversarial Examiners***, where the examiner selects images adaptively to search for the weaknesses of the algorithms.
- ***We need algorithms that can pass these tougher tests.***
- *Alan Yuille & Chenxi Liu: Deep Nets: What have they ever done for Vision? IJCV. 2021.*

Standard Performance Measures: Simple Formulation.

- SPMs assume there is an unknown distribution $P(x, y)$ and our task is to find an classifier $y = f(x : \theta)$ given a loss function $L(y, f(x, \theta))$.
- We have training samples $\mathcal{X}_{Train} = \{(x_i, y_i) : i = 1, \ldots, N\}$ and testing samples $\mathcal{X}_{Test}\{(x_a, y_a) : a = 1, \ldots, M\}$.
- We train the classifier to minimize $\sum_{(x,y)\in\mathcal{X}_{Train}} L(y, f(x,\theta))$. We evaluate our performance by $\sum_{(x,y)\in\mathcal{X}_{Test}} L(y, f(x,\theta))$.

- If there is sufficient training data, in terms of the complexity of the classifiers, then good performance on the training set will imply good performance on the testing set (must be checked to avoid overfitting).
- Mathematical analysis – VC/PAC theory – shows that the classifiers will almost certainly generalize to any data sampled from P(x,y).
- ***But, as we argued, SPMs are problematic and tougher tests are needed.***

Out-of-Distribution: Toy Example

- Suppose our training data is generated from $P(x, y)$. But our test data is generated from $(1 - \epsilon)P(x, y) + \epsilon Q(x, y)$, where $Q(x, y)$ is another distribution and ϵ is a constant.
- If ϵ is small, then classifiers trained on data from $P(x, y)$ may still have good performance on data from $(1 - \epsilon)P(x, y) + \epsilon Q(x, y)$.
- But performance will surely degrade badly if ϵ is large.
- How to address this? One solution is to use Bayesian generative models. Instead of learning the classifiers we learn probability distributions $P(x|y)$ and $P(y)$ from the training set. We estimate $Q(x|y)$ using other data. Then we can combine them to estimate y when the data comes from $(1 - \epsilon)P(x, y) + \epsilon Q(x, y)$.
- This talk describes a more complex model based on this strategy.

Compositional Generative Networks (CGNs)

- Compositional Generative networks (CGNs) are generative models of DN convolutional features. They have standard Deep Net backbones but replace the discriminative head by a generative model.
- *Why Generative? They have knowledge of the generation process: (I) Objects are seen from different viewpoints and have different spatial patterns for each viewpoint. (II) They know that parts of the object are invisible because they are occluded.*
- We test Deep Nets and CGNs for occluded objects (out-of-distribution testing) and then for robustness to patch attacks (Adversarial Examiner).
- A. Kortylewski et al. CVPR 2020, A. Wang et al. CVPR 2020, A. Kortylewski et al. IJCV 2020. A. Kortylewskii et al. CISS 2021.

Generalization to occlusion (out-of-distribution)

- In natural images objects are surrounded and partially occluded by other objects
- Occluders are highly variable in terms of shape and texture -> **exponential complexity**
- Vision systems must generalize in exponentially complex domains

Occlusion -- A Fundamental Limitation of Deep Nets?

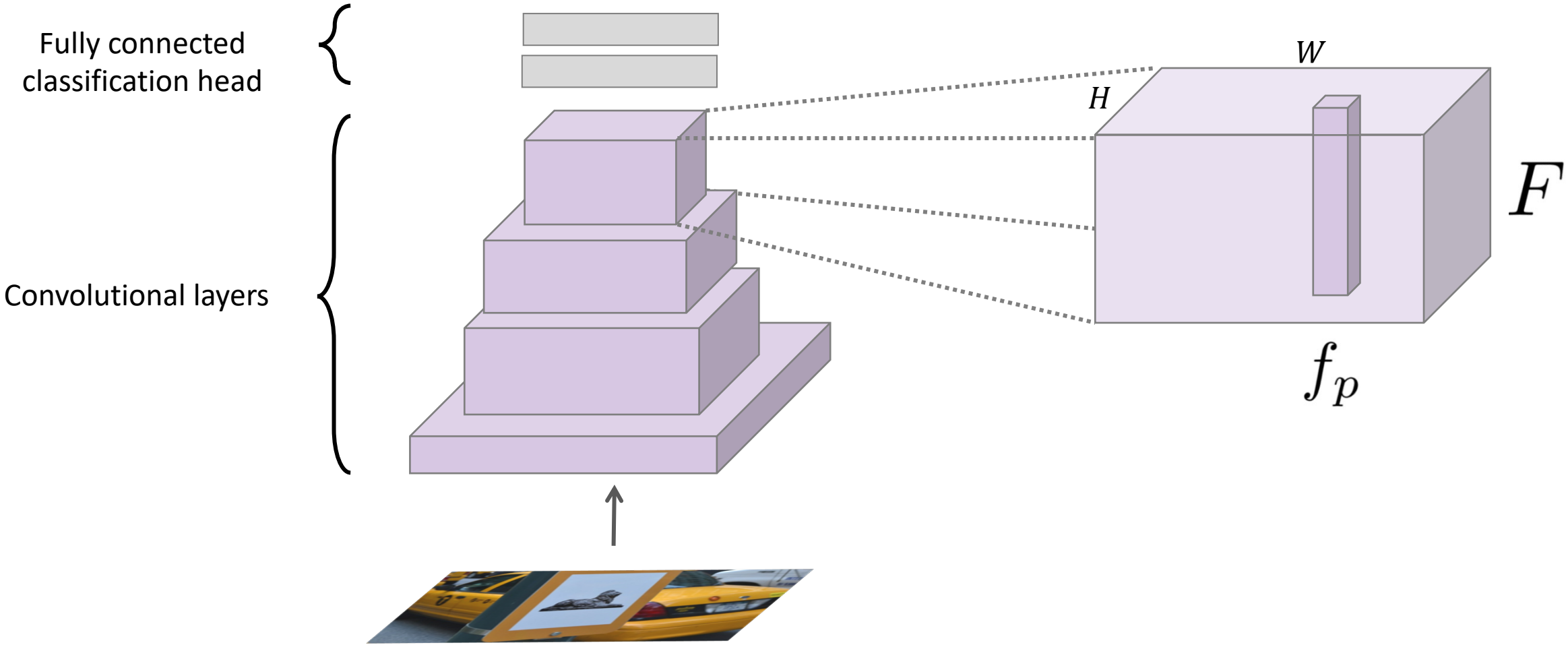
- DCNNs do not generalize when trained with non-occluded data

Occ. Area	**0%**	**30%**	**50%**	**70%**	Avg
VGG -16	99.1	88.7	78.8	63.0	82.4

- What if we train with lots of augmented data? Better, but still not good enough.

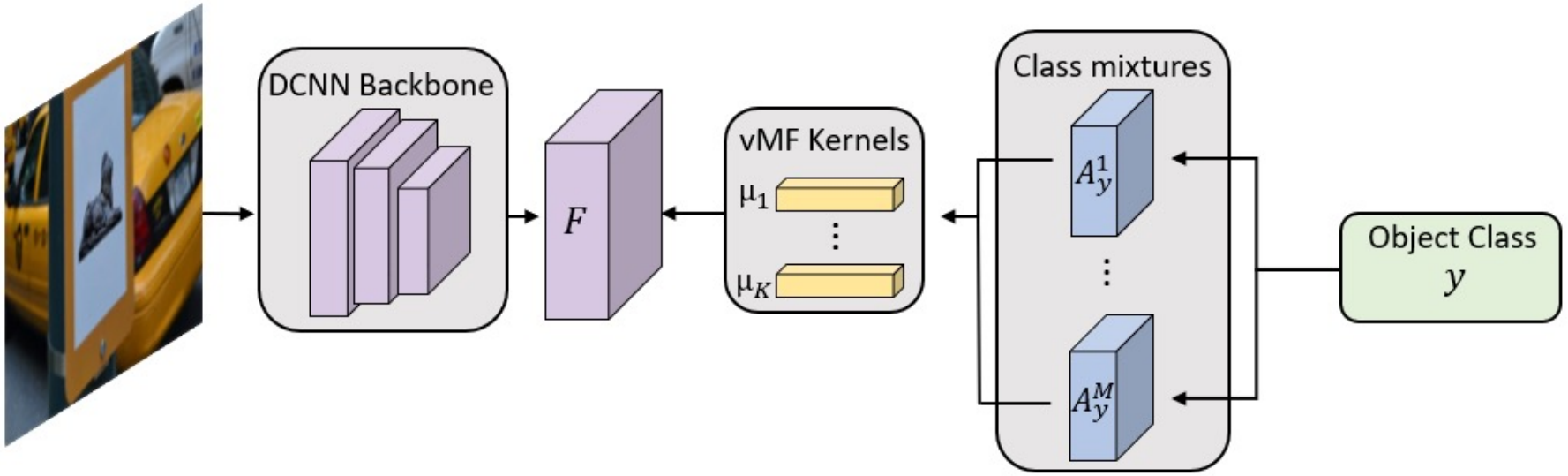
Occ. Area	**0%**	**30%**	**50%**	**70%**	Avg
VGG-16-Augmented	99.3	92.3	89.9	80.8	90.6

CompNets: A Generative Model of Neural Feature Activations

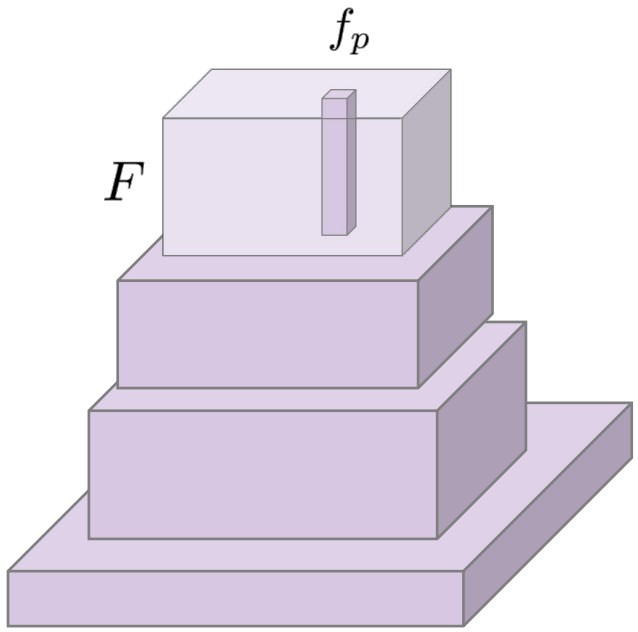


CompNets: The Big Picture

- Each objects is represented by a mixture of distributions (will approximately correspond to viewpoints). Each of these mixtures will generate a spatial pattern of features.
- This will require no additional annotations during training.



Mathematics of the CompNets.



$$p(F|\Theta_y) = \sum_m \nu^m p(F|\theta_y^m)$$

Class mixtures (~viewpoints) - m

$$p(F|\theta_y^m) = \prod_p p(f_p|\mathcal{A}_{p,y}^m, \Lambda)$$

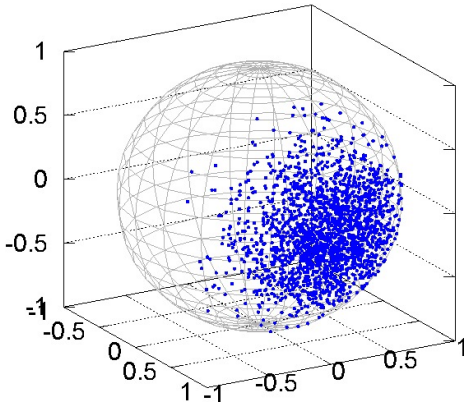
Factorizable in spatial position -- p

$$p(f_p|\mathcal{A}_{p,y}^m, \Lambda) = \sum_k \alpha_{p,k,y}^m p(f_p|\lambda_k), \quad \lambda_k = \{\mu_k, \sigma_k\}$$

At each p – mixture of vMF kernels (~parts)

$$p(f_p|\lambda_k) = \frac{e^{\sigma_k \mu_k^T f_p}}{Z(\sigma_k)}, \|f_p\| = 1, \|\mu_k\| = 1$$

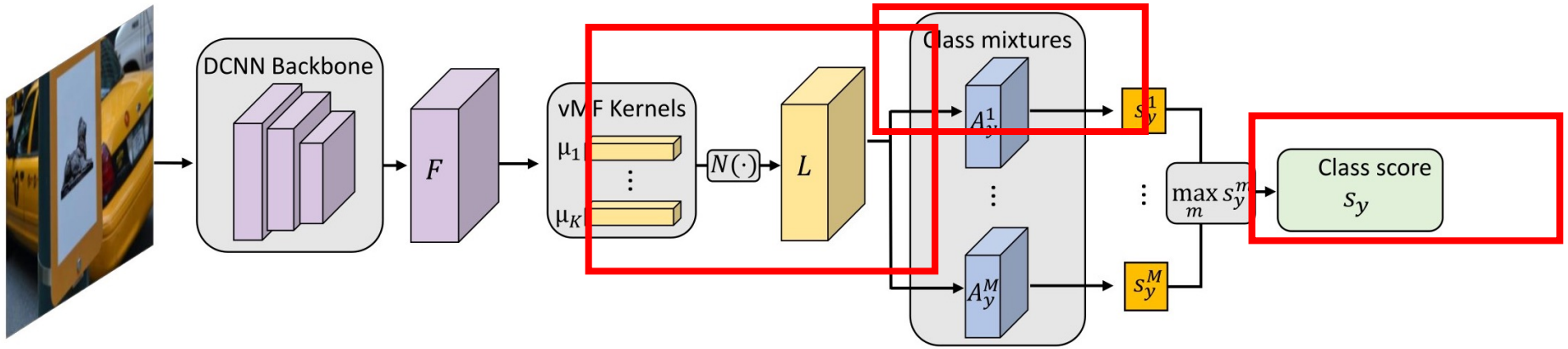
Each vMf kernel is Von-Mises Fisher -- Gaussian on a sphere



Y labels object class
P labels position in the image
fp are the feature vectors at p
m label the mixture (viewpoint)
alpha's,lambda's,mu's are parameters which are learnt.

$\Theta \ \mathcal{A}_{p,y}^m, \Lambda)$ are summary parameters

To classify the object we perform Bayesian Inference. This can be performed by a modified Feed-Forward Network



Class scores are the evidence for each object class.
Class mixture are (roughly) the viewpoints of each object.

vNF kernels are (roughly) object parts.

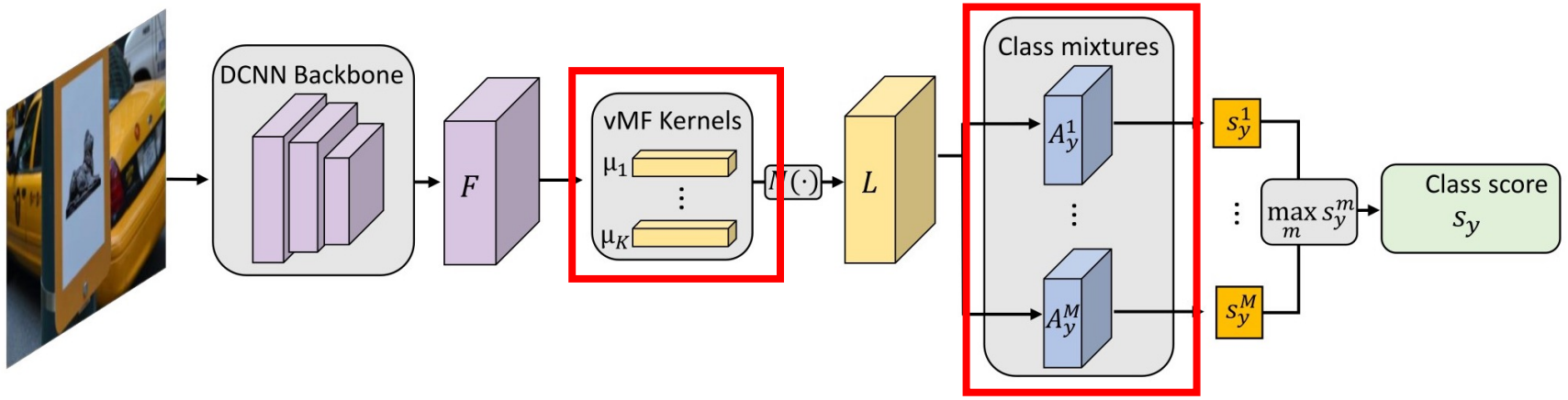
1. vMF likelihood: $$p(f_p|\lambda_k) = \frac{e^{\sigma_k \mu_k^T f_p}}{Z(\sigma_k)}, \|f_p\| = 1, \|\mu_k\| = 1$$

2. Mixture likelihoods: $$p(F|\theta_y^m) = \prod_p \sum_k \alpha_{p,k,y}^m p(f_p|\lambda_k)$$

3. Class score: $$p(F|\Theta_y) = \sum_m \nu^m p(F|\theta_y^m), \quad \nu^m \in \{0,1\}, \sum_m \nu^m = 1$$

The computations at the top levels of the network are slightly different from those in a standard DN.

The Parameters of the CompNet are learnt by minimzing a loss function by Backpropagation & Clustering



$$\mathcal{L} = \mathcal{L}_{class}(y, y') + \gamma_1 \mathcal{L}_{weight}(W) + \gamma_2 \mathcal{L}_{vmf}(F, \Lambda) + \gamma_3 \mathcal{L}_{mix}(F, \mathcal{A}_y)$$

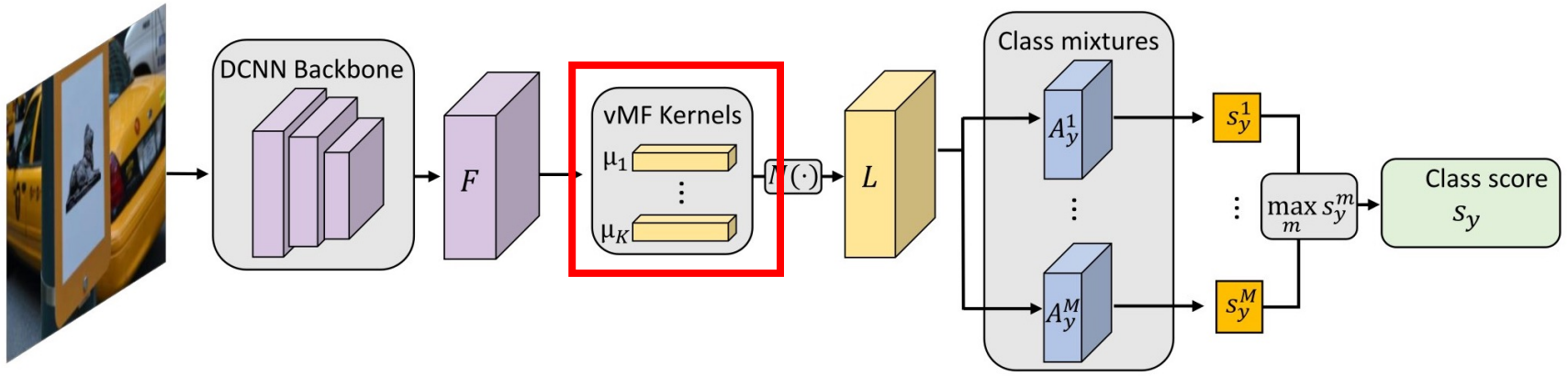
$$\mathcal{L}_{vmf}(F, \Lambda) = -\sum_p \max_k \log p(f_p|\mu_k) = C \sum_p \min_k \mu_k^T f_p$$

$$\mathcal{L}_{mix}(F, \mathcal{A}_y) = -\sum_p \log \left[\sum_k \alpha_{p,k,y}^m p(f_p|\mu_k) \right]$$

Clustering is needed to get the vMF Kernels (~parts) and the class mixtures (~viewpoints).

The clustering is done by using spectral clustering to initialize von-Mises-Fisher clustering (similar to mixture of Gaussians but on a sphere).

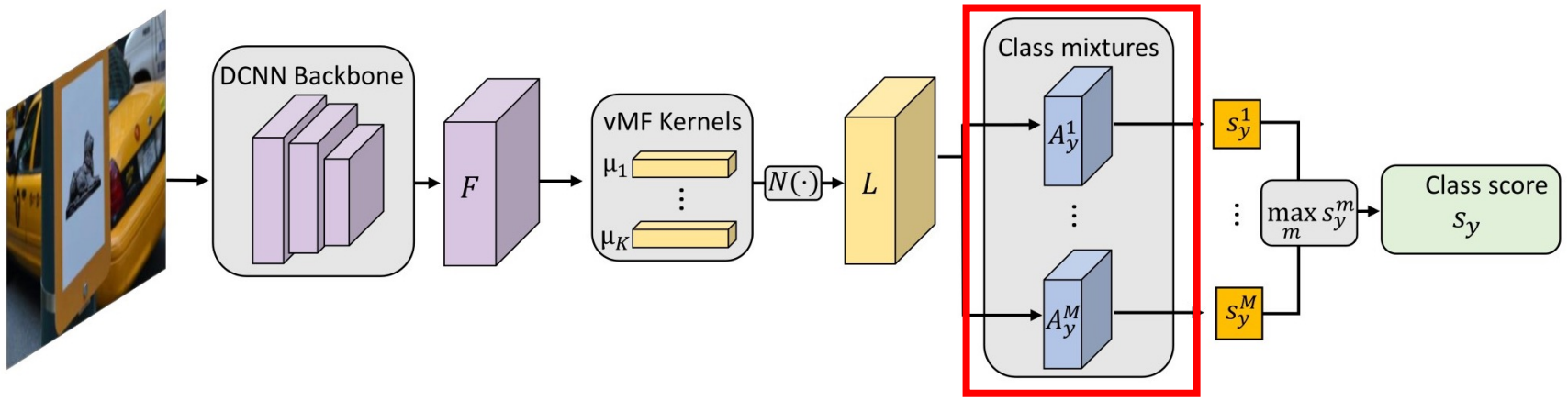
Explainability - vMF Kernels resemble „part detectors“



- Image patterns with highest likelihood:

Note: the vMF Kernels can be quantified because we hand-annotated parts of these objects. Not just nice pictures. Quantification is not perfect but better than unsupervised alternatives.

Explainability – Class mixtures are similar to object viewpoint



- Images with highest likelihood for mixture components:



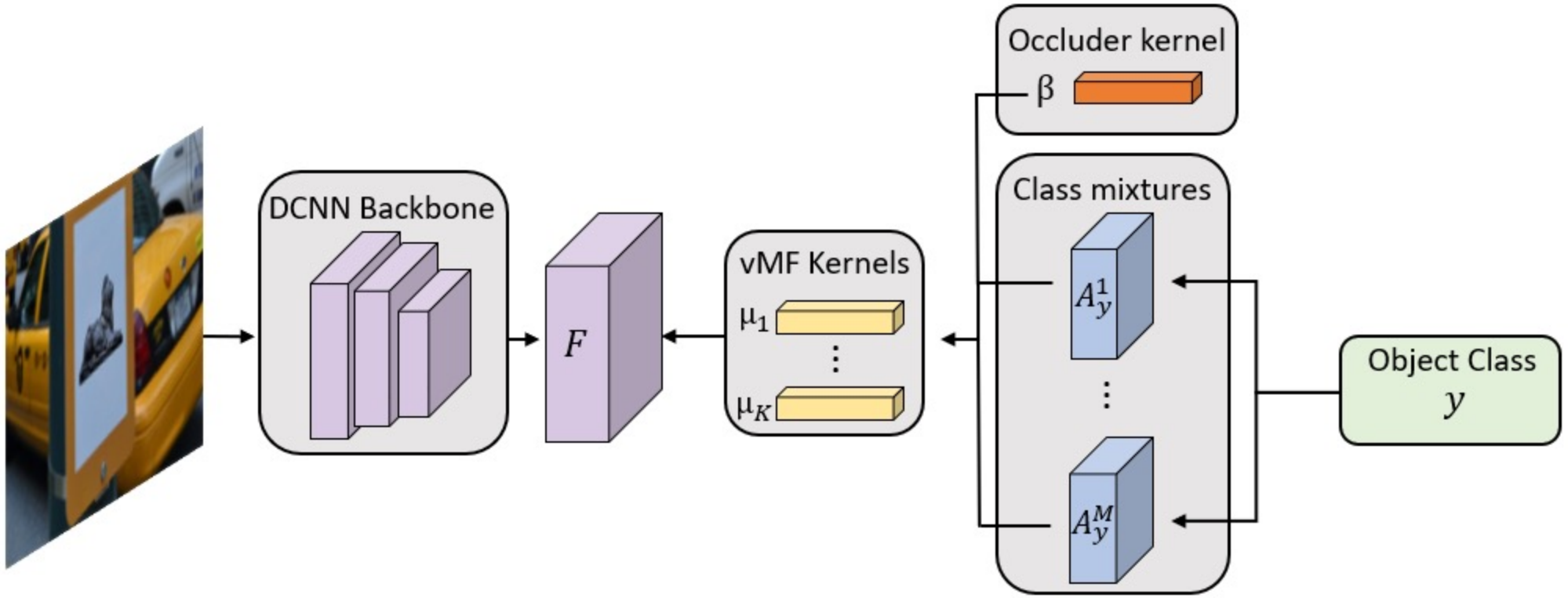

Quantitative studies show that class mixtures typically correspond to object viewpoints, but can correspond to frequently occurring spatial patterns (tandem bikes).

Out-of-distribution: Occlusion modeling using an outlier model

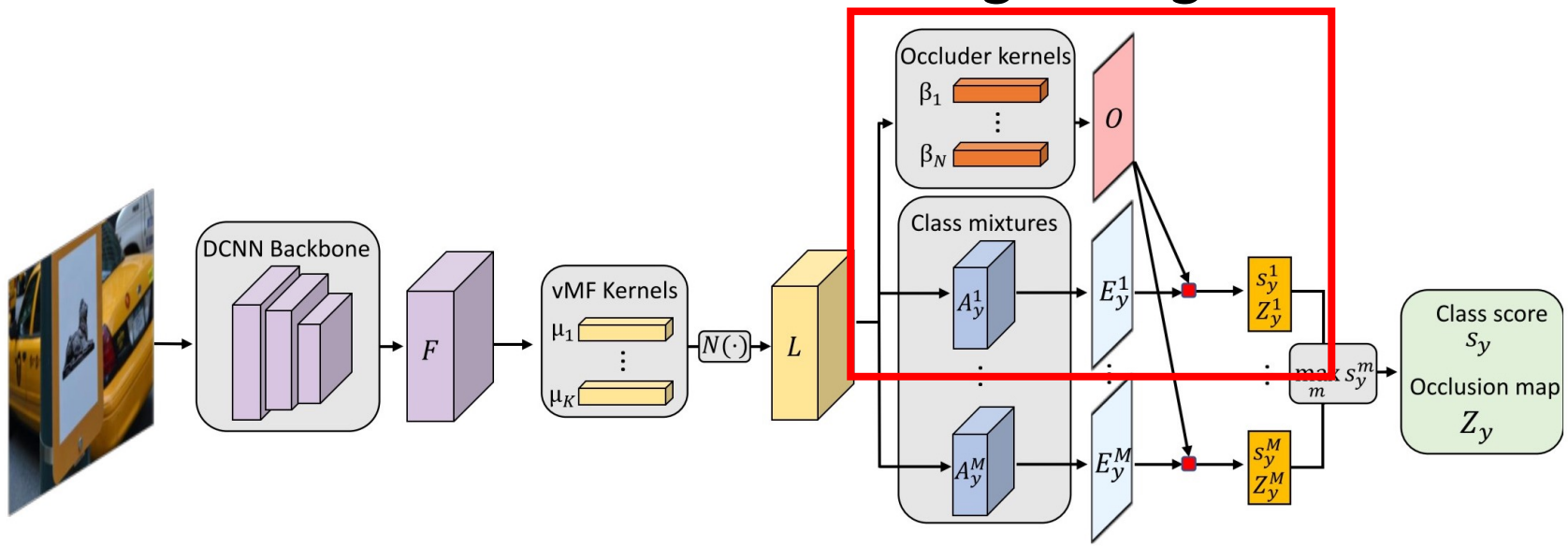
- Recall our earlier "toy" example. Training samples for the object models (unoccluded) are drawn from $P(x, y)$ and we use them to learn $P(x, y)$.
- But the test data is drawn from $(1 - \epsilon)P(x, y) + \epsilon Q(x, y)$, where $Q(x, y)$ is analogous to occluded.
- We learn the model $Q(x, y)$ separately. Then we can use Bayesian methods to combine it with $P(x, y)$.
- Note: this is a simplification but it captures the main ideas. The full model is more complex and is presented in the next few slides.

Introduce an Outlier (Occluder) Process

- Some of the features vectors are generated by one of the object class mixtures models, but others are generated by occluders.



Out-of-distribution: Occlusion modeling using an outlier model



- We introduce an outlier model:

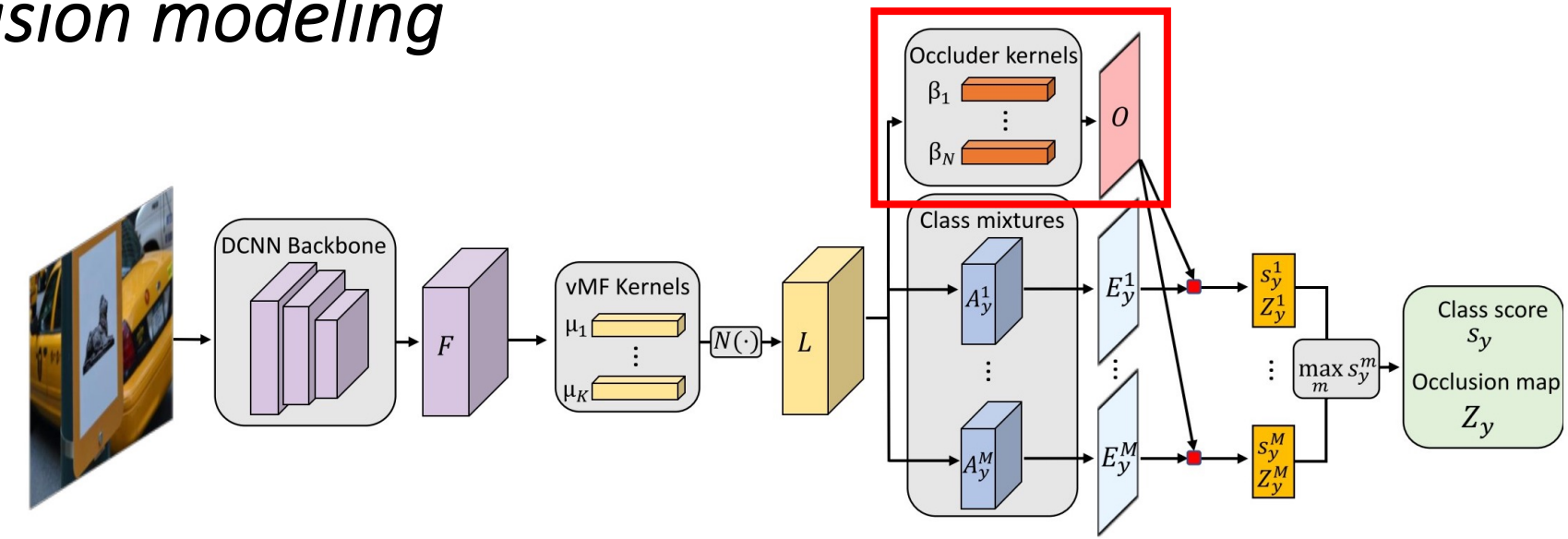
$$p(F|\theta_y^m, \beta) = \prod_p p(f_p, z_p^m = 0)^{1-z_p^m} p(f_p, z_p^m = 1)^{z_p^m}, \quad \mathcal{Z}^m = \{z_p^m \in \{0,1\} | p \in \mathcal{P}\}$$

$$p(f_p, z_p^m = 1) = p(f_p|\beta, \Lambda)\ p(z_p^m = 1),$$

$$p(f_p, z_p^m = 0) = p(f_p|\mathcal{A}_{p,y}^m, \Lambda)\ (1 - p(z_p^m = 1)).$$

Z is estimated by the algorithm during inference.
If z-p = 0, then feature at p is generated by object
If z-p = 1, the feature at p is generated by occlude.

Occlusion modeling

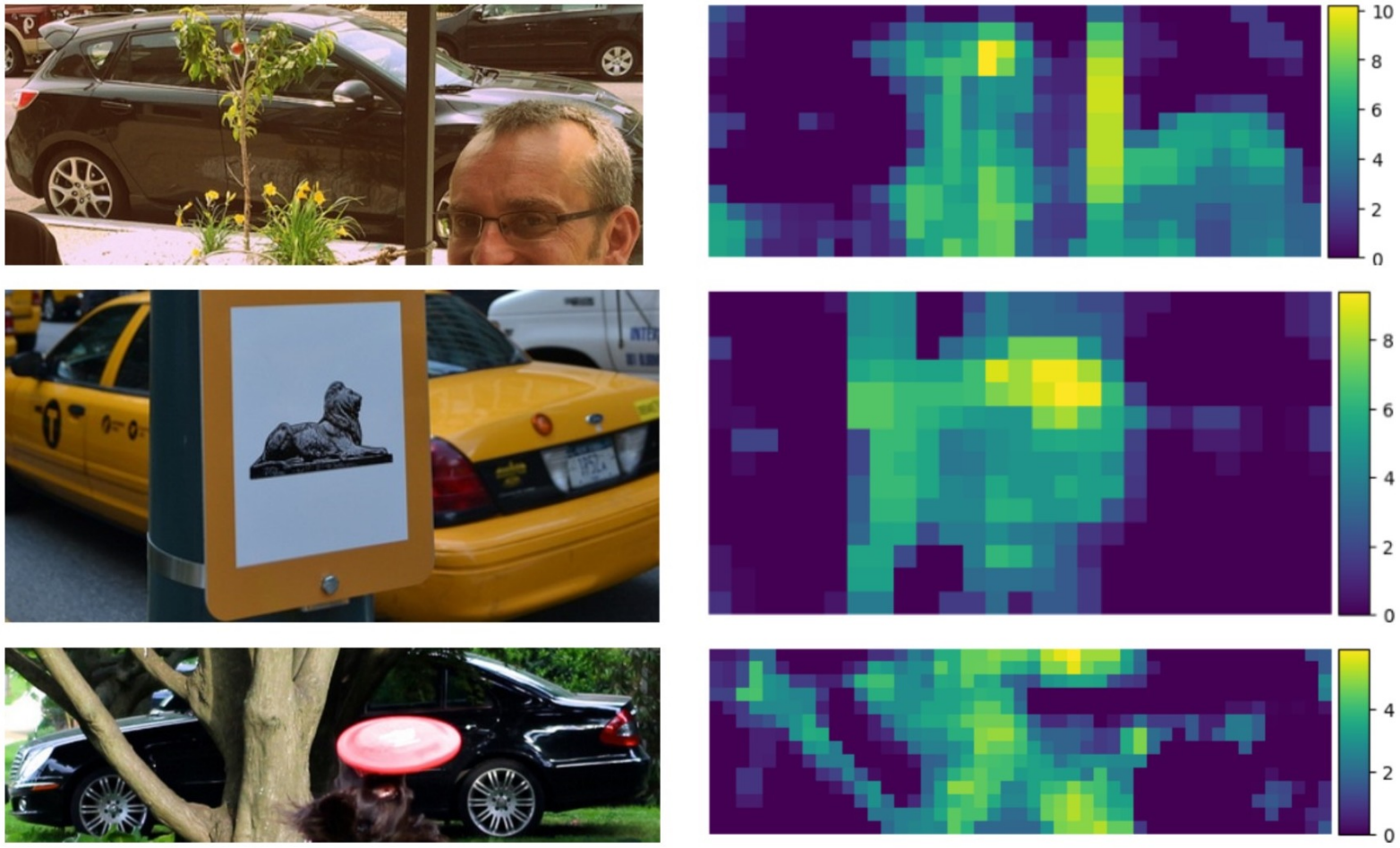


- We introduce an outlier model:

$$p(F|\theta_y^m, \beta) = \prod_p p(f_p, z_p^m = 0)^{1-z_p^m} p(f_p, z_p^m = 1)^{z_p^m}, \quad \mathcal{Z}^m = \{z_p^m \in \{0,1\} | p \in \mathcal{P}\}$$

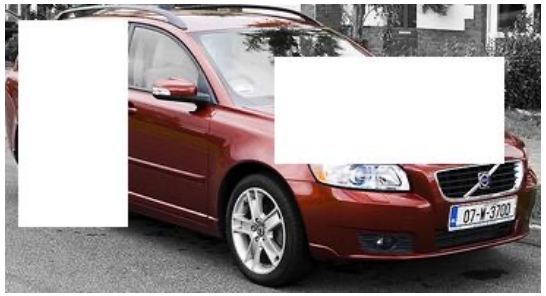
- A simple model of how the object does not look like: learnt seperately.

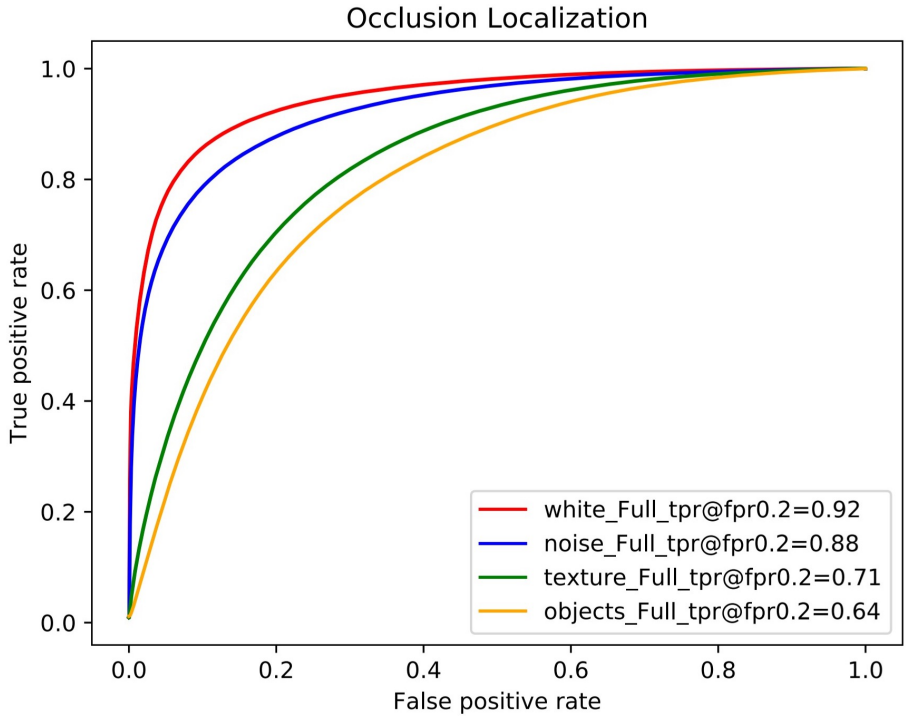
Competition between object and outlier model



We plot P(z-p =1) as a function of p.
Yellow and green indicate the highest probability of occlusion

Quantitative Evaluation of Occluder Localization



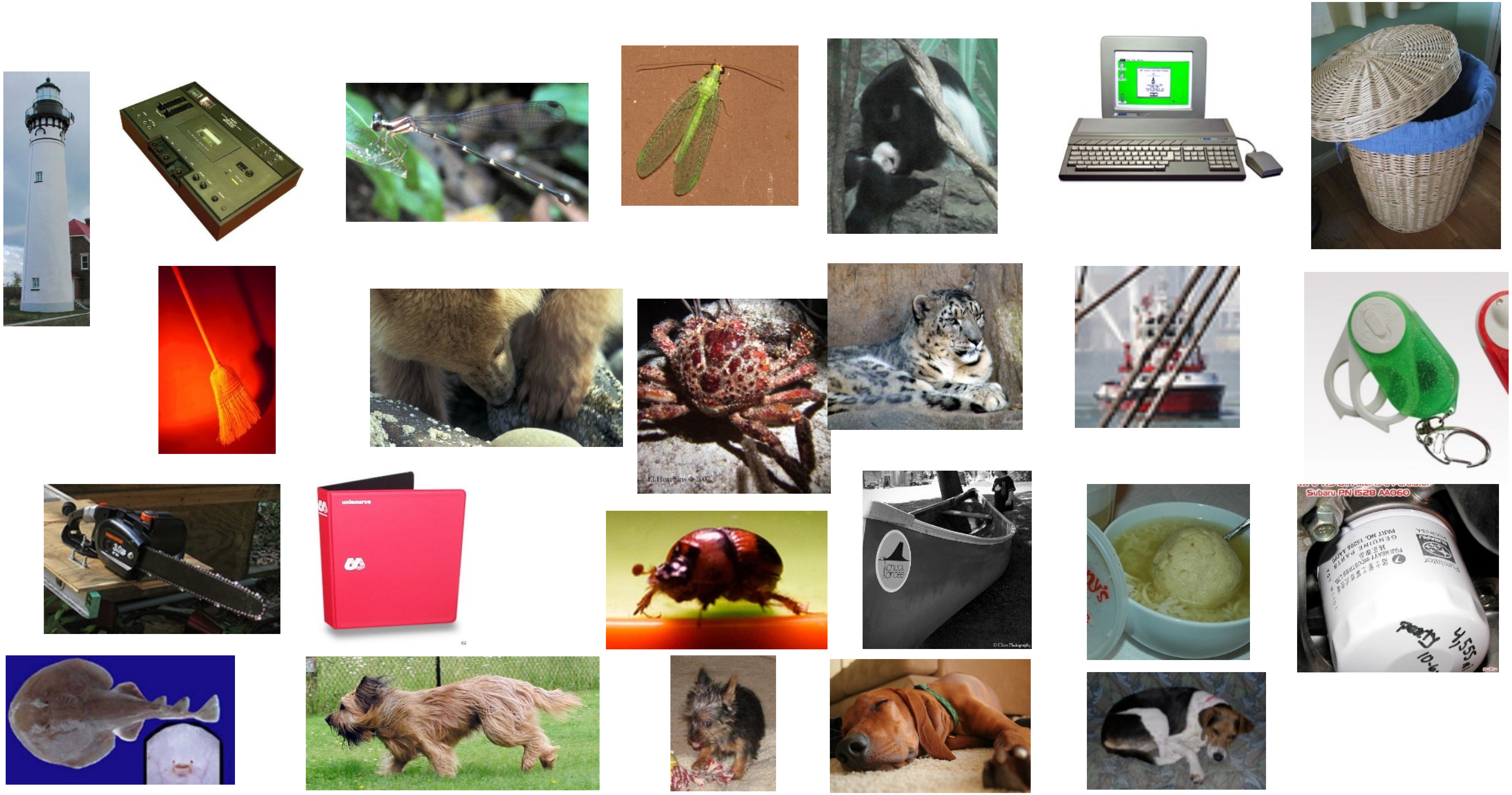
White occluders are easiest to detect and localize
Objects are the hardest to detect and localize.

CompNets can classify partially occluded vehicles robustly

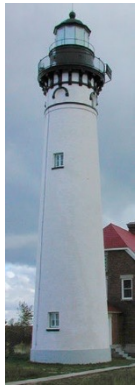



Occ. Area	**L0**	**L1**	**L2**	**L3**	Avg
VGG	97.8	86.8	79.1	60.3	81.0
ResNet50	98.5	89.6	84.9	71.2	86.1
ResNext	98.7	90.7	85.9	75.3	87.7

Scaling up to 100 Object Categories. A. Kortylewski et al. IJCV 2020.



Scaling up to 100 Object Categories.













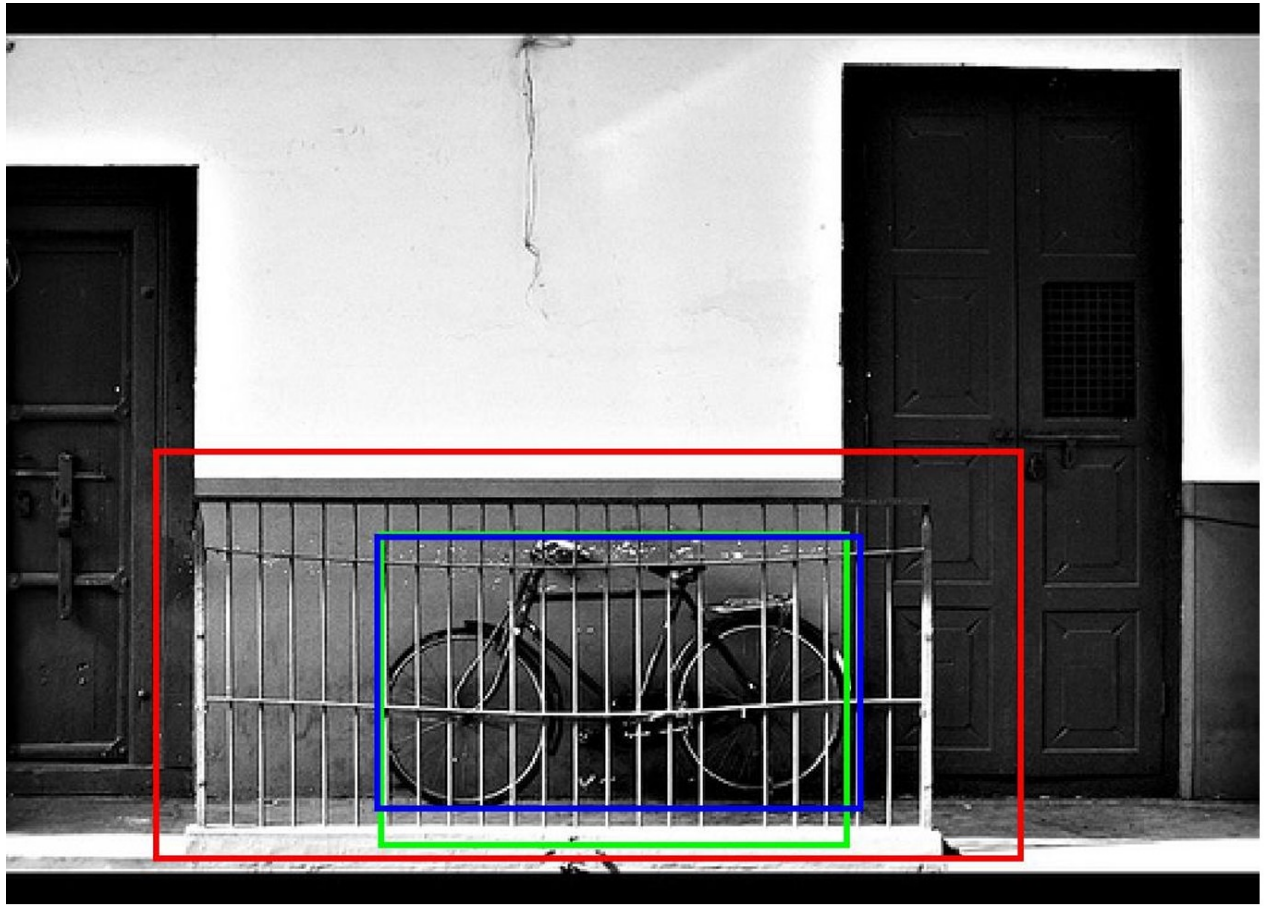


ImageNet under Occlusion

Occ. Area	0%	30%	50%	70%	Avg
ResNext	**98.4**	69.3	48.7	31	61.9
CompNet-ResNext	96.3	**76.6**	**60.1**	**45.5**	**69.6**

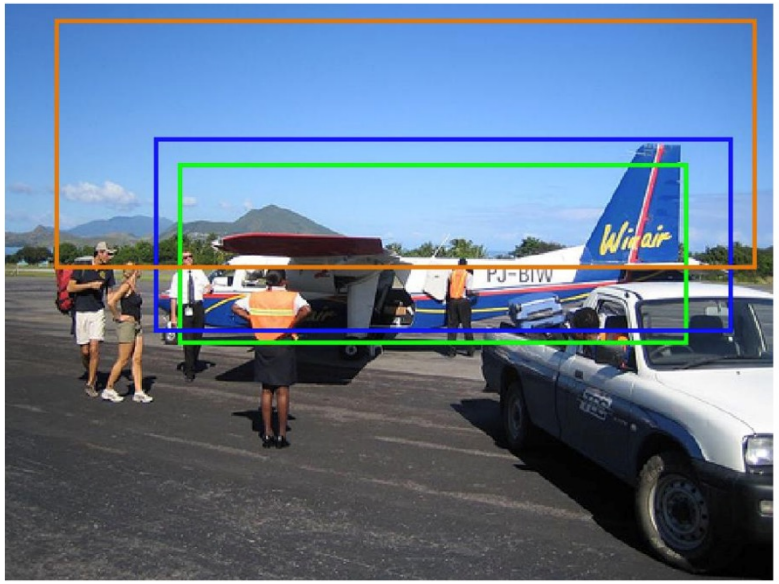
Performance degrades as the number of object classes increases. This is probably because 2D CGNs assume that objects can be represented by four viewpoints (which we can estimate). This is a good approximation for vehicles but a terrible one for animals and boats.

CGNs for Object Detection: (A. Wang et al. CVPR 2020).
DCNNs for object detection also do not generalize well to occlusion



Background context has too much influence when object is occluded

Deep Nets exploit background context too inflexibly. They learn that airplanes are often in blue sky. So if the objects are occluded then the influence of blue sky can become too big.



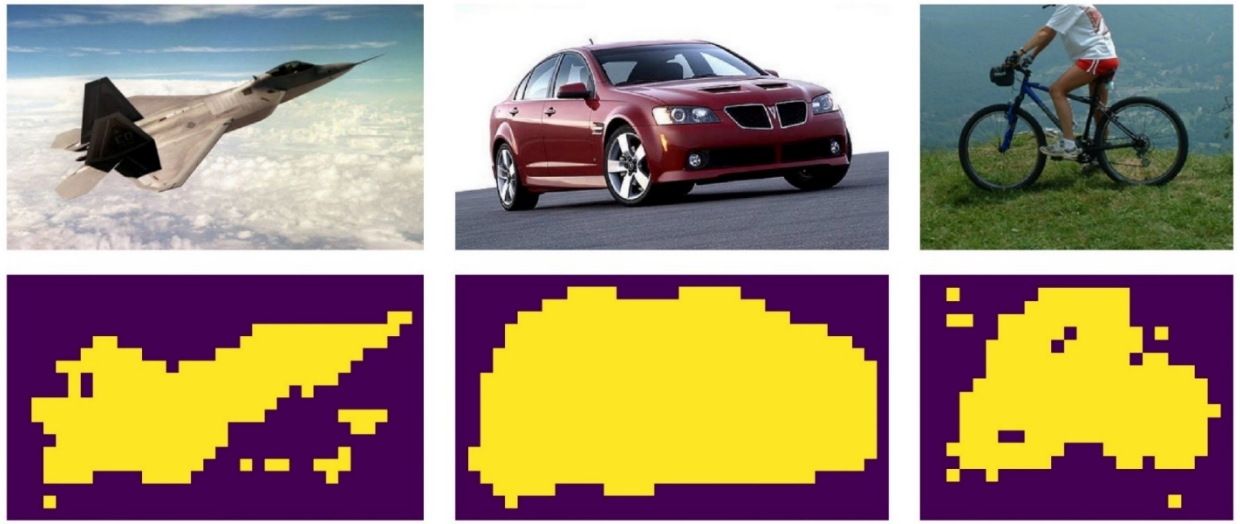
Seperate the representation of background context and object: Easy to do for generative models.

We can learn a model of the background context, similar to how we learnt the model for the occluder. This enables us to partially segment the object.

- We introduce a context-aware object model:

$$p(f_p|\mathcal{A}^m_{p,y}, \chi^m_{p,y}, \Lambda) = \omega\ p(f_p|\chi^m_{p,y}, \Lambda) + (1-\omega)p(f_p|\mathcal{A}^m_{p,y}, \Lambda)$$

- Segment the image during training:

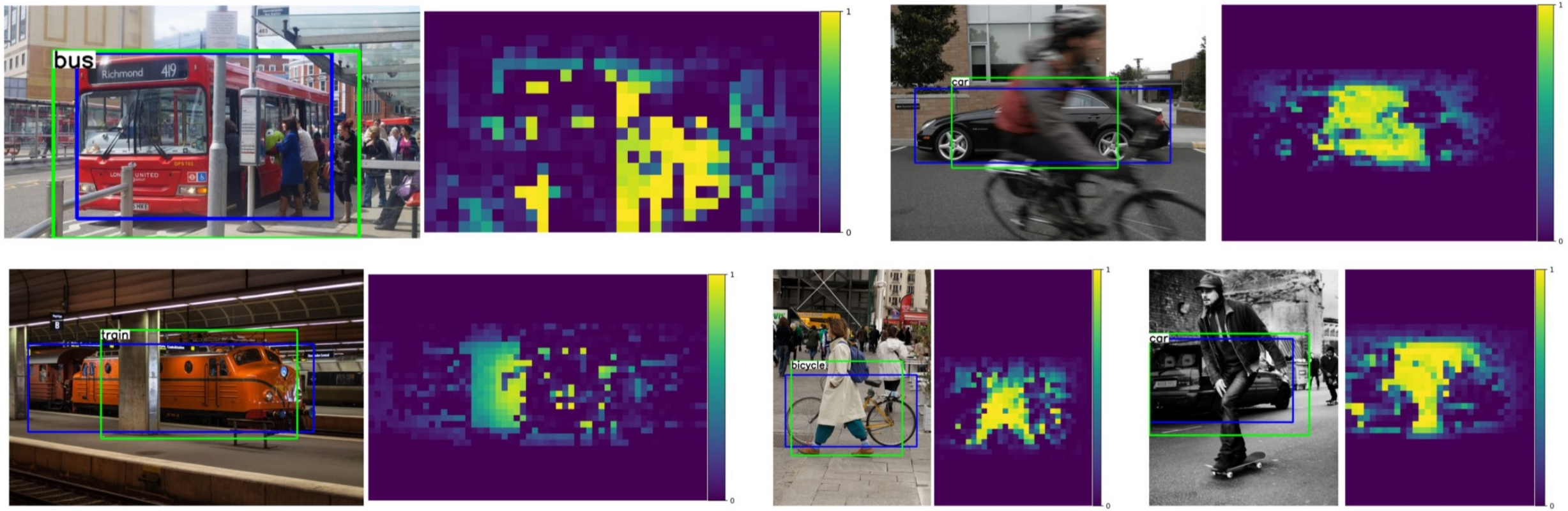


Intuitively, the background context model for an airplane will mostly be sky. So background pixels near the airplane are likely to be classified as non-airplane.

Context-awareness Improves Localization

The context-aware model means that we discount the evidence of the local background and pay more attention to the evidence of the object.

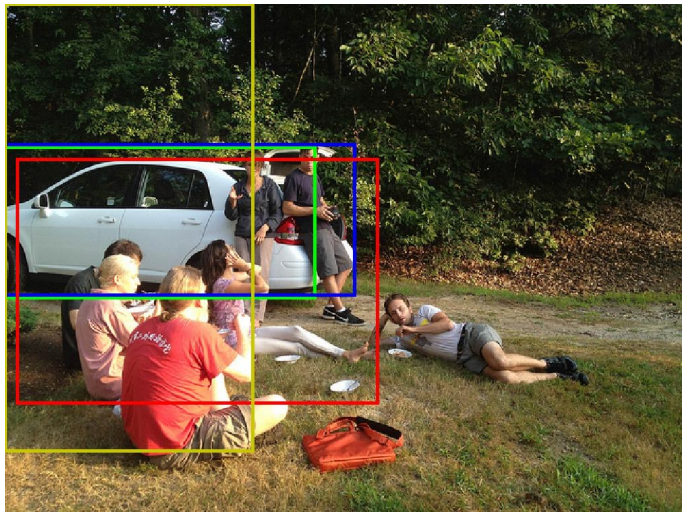
Explainability- Occluder localization in Object detection

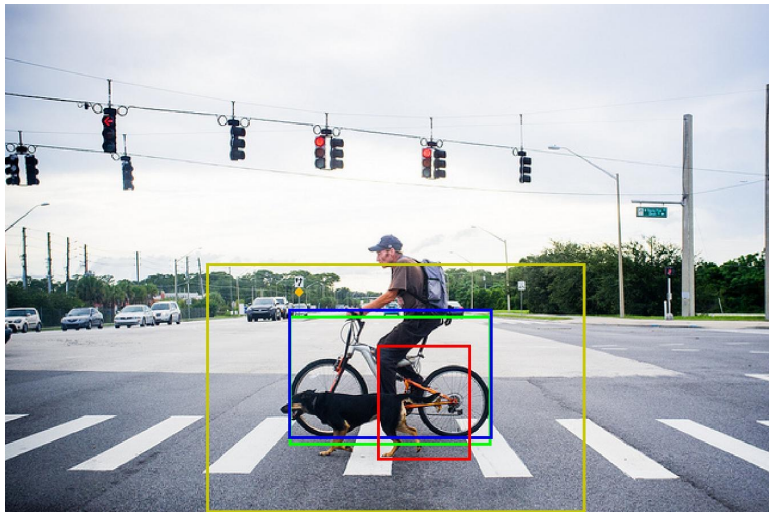


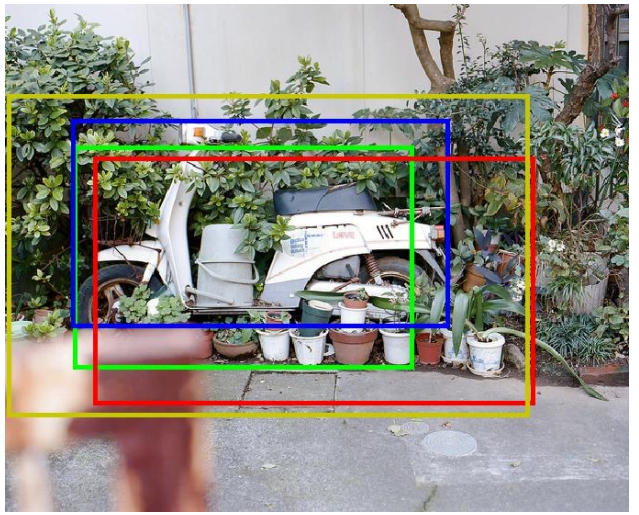
We can detect, and classify, the objects, and detect and localize the occluders.

Detection Results

method	light occ.	heavy occ.
Faster R-CNN	73.8	55.2
Faster R-CNN with reg.	74.4	56.3
Faster R-CNN with occ.	77.6	62.4
CA-CompNet via BBV $\omega = 0.5$	78.6	76.2
CA-CompNet via BBV$\omega = 0.2$	**87.9**	**78.2**
CA-CompNet via BBV $\omega = 0$	85.6	75.9





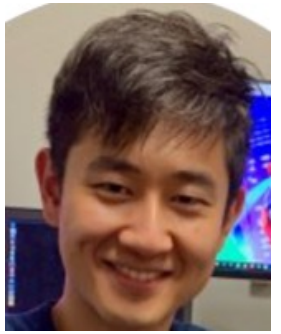


Caveat: in this paper we assume that the range of object sizes are small.

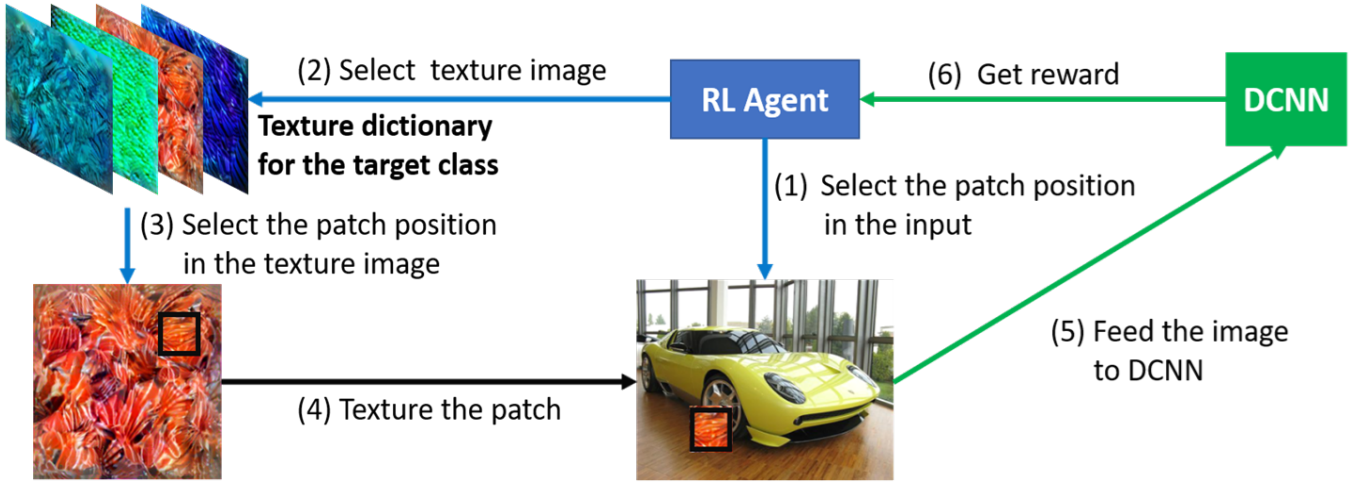
Adversarial Examiners: Robustness out of the box

- Background – standard computer vision & machine learing practice is to evaluate algorithms by average case performance on a finite-sized balanced annotated dataset (BAD).
- We argue that it is better to evaluate algorithms by trying to identify their weak points by dynamic testing, ***i.e. modifying the input images adaptively to cause the algorithms to fail – Adversarial Examiner.***
- In previous work, see next slide, we developed patch-attacks which could fool Deep Nets by adding a few small patches to images. The patches are their locations were chosen by a search strategy with feedback from the algorithm.
- These blackbox targeted attacks had over 90% success rate on advanced Deep Nets. Suggests that Deep Nets lack knowledge of the spatial structures of objects.
- *Chenglin Yang et al. ECCV. 2020.*

Adversarial Examiners: Patch Attacks



- Learn an Attack Policy by reinforcement learning.



Network	Attack	T_acc. (%)	Avg_area (%)	Avg_qry
ResNet50	–	0.10	–	–
	HPA	23.20	71.54	50000
	MPA_RGB	25.90	18.45	28361
	TPA_N10_2%	97.60	7.80	15728
	TPA_N10_4%	99.70	9.97	8643
	TPA_N10_10%	100.00	15.36	3747

- A black box targeted attack which is almost 100% effective.
- Suggests Deep Nets have little knowledge of the global structure of objects. They are just “bags of patches”. They only pay “attention” to small regions of the image. They find that synthetic texture patches (obtained by a surrogate deep network) look more like an object than the object itself.

Robustness out of the Box.

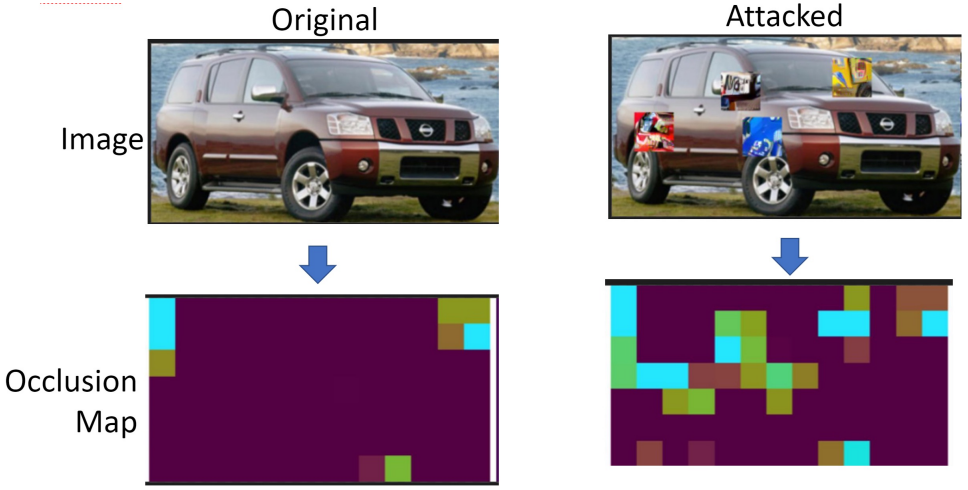
- We conjectured that CompNets would do better than Deep Nets because they have knowledge of the spatial structure of objects and their outlier process may enable them to reject the attacking patches.
- We are correct. Patch-attacks (and related attacks) are less successful on CompNets by an order of magnitude. CompNets also have some ability to detect and localize the patch attacks.
- This gives more evidence that CompNets are much more robust than Deep Nets. Datatset: PASCAL+PatchAttacks.

CompNets are robust to Patch Attacks.

- CompNets ***are*** robust against targeted patch attacks.

Model	Accuracy (%)	Attack success rate (%)	
		PatchAttack[1] (TPA) 4 patches	Sparse-RS[2] 1 patch
CompNet (vgg16 backbone)	98.5	12.6	0.9
vgg16	98.6	98.8	92.3

CompNets can localize attacks.



- C. Cosgrove et al. Robustness out of the box. Arxiv. 2020.
- A. Kortylewski et al. CISS. 2021.

Robusteness out of the Box

- CompNets are much more robust than Deep Nets to patch-attacks without needing any modifications.
- Performance degrades for fine-detail tasks (can be fixed by engineering tricks in the short term).
- This shows a limitation of our current Approximate Analysis by Synthesis. The CGNs only use high level convolutional features, which ignore fine-details. They are a good starting point but will need to be modified to include lower-level features and more precise spatial structure.

Summary

- Compositional Generative Networks (CGNs) perform as well as standard Deep Nets but are also robust to occluders and patch attacks. They show the power of Bayesian methods.
- Standard performance measures (SPMs) are problematic for evaluating vision algorithms. Particularly if we want to develop AI vision algorithms which are as robust and adaptive as the human visual system.
- We need tougher measures like out-of-distribution testing and adversarial examiners. And we need algorithms that can perform well on these measures.

References:

- C Cosgrove, A Kortylewski, C Yang, A Yuille. Robustness Out of the Box. arXiv. 2020.
- A. Kortylewski , J. He, Q. Liu, A. Yuille. Compositional Convolutional Nets, CVPR 2020.
- A. Kortylewski, Q. Liu, A. Wang, Y. Sun, A. Yuille. Compositional Convolutional Neural Networks. IJCV. 2020.
- A. Kortylewski, J. He, Q. Liu, C. Cosgrove, C. Yang, A. Yuille. Compositional Generative Networks and Robustness to Perceptible Image Changes. CISS. 2021.
- A. Wang, Y. Sun, A. Kortylewski, A. Yuille. Robust Object Detection under Occlusion with Context-Aware Compositional Nets. CVPR 2020.
- C. Yang, A. Kortylewski, C. Xie, Y. Cao, A. Yuille. Patch Attack. ECCV. 2020.
- A.L. Yuille & C. Liu. Deep Nets: What have they ever done for Vision. IJCV. 2021.