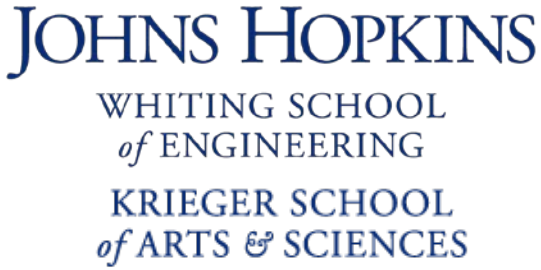


Deep Networks and Beyond: Vision and Machine Learning

Computational Cognition, Vision, and Learning

Alan Yuille

Departments of Cognitive Science and Computer Science

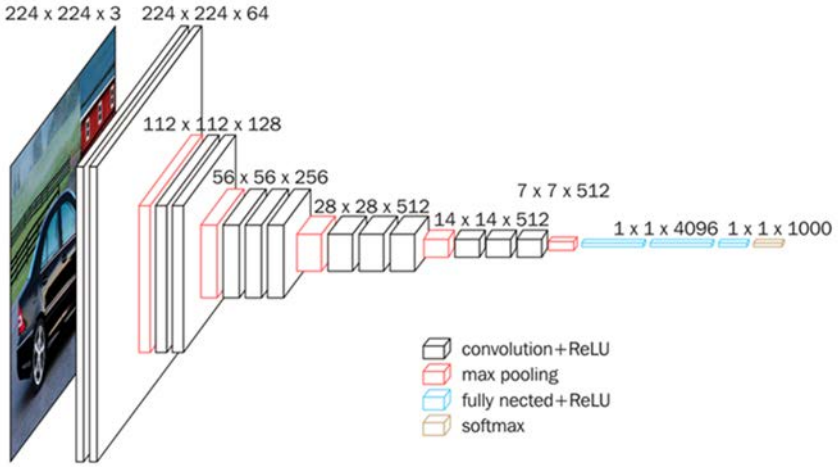
Johns Hopkins University

Deep Networks

- I have a love-hate relationship with Deep Networks.
- Their performance is extremely good for some visual tasks.
- Their lack of intepretability is worrying. Need to understand and diagnose them.
- They are a very rich and constantly evolving class of techniques.
- They are very useful, but are not sufficient to solve vision.

What can Deep Nets Do?

- Deep Nets, and other Machine Learning tools, have given huge progress for many vision tasks.
- Hierarchical Feature Representation.
- The output is a differentiable function of the input and the weights.
- This enables learning the weights by Stochastic gradient descent.



Examples: Face Recognition, Text Recognition, Medical Image Analysis

Several Parts

- Part 1. Example. The FELIX project.
- Part 2. Why are Deep Networks Deep? GUNN.
- Part 3. Combining Deep Networks with Random Forests
- Part 4. Few-Shot Learning
- Part 5. Unsupervised Deep Networks.
- Part 6. Attacking Deep Nets.
- Part 7. When is Big Data not enough?

Several Parts

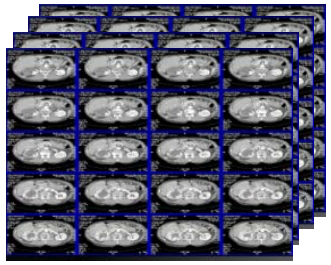
- **Part 1. Examples**
- Part 2. Why are Deep Networks Deep? GUNN.
- Part 3. Combining Deep Networks with Random Forests
- Part 4. Few-Shot Learning
- Part 5. Unsupervised Deep Networks.
- Part 6. Attacking Deep Nets
- Part 7. When is Big Data not enough?

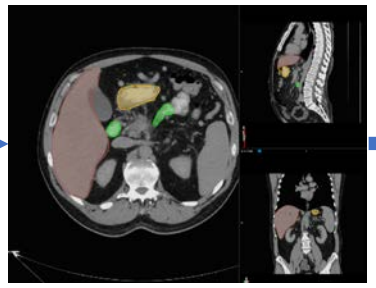
Part 1. Example

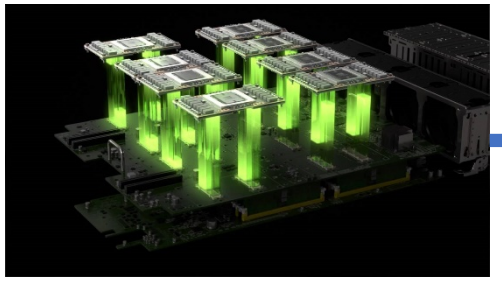
FELIX project: medical imaging CT cancer detection

Training

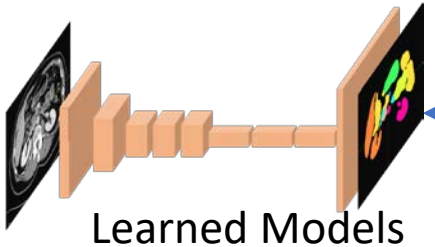

Image Scan


Medical Image Data


Annotated Data


Computing resource

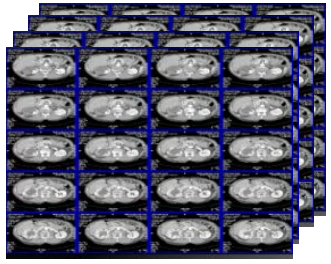

Professional Knowledge

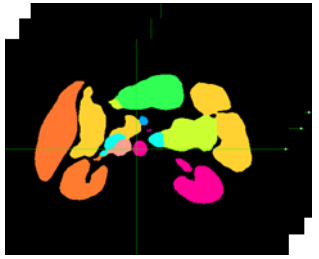

Learned Models

Deep Learning

Testing


Image Scan

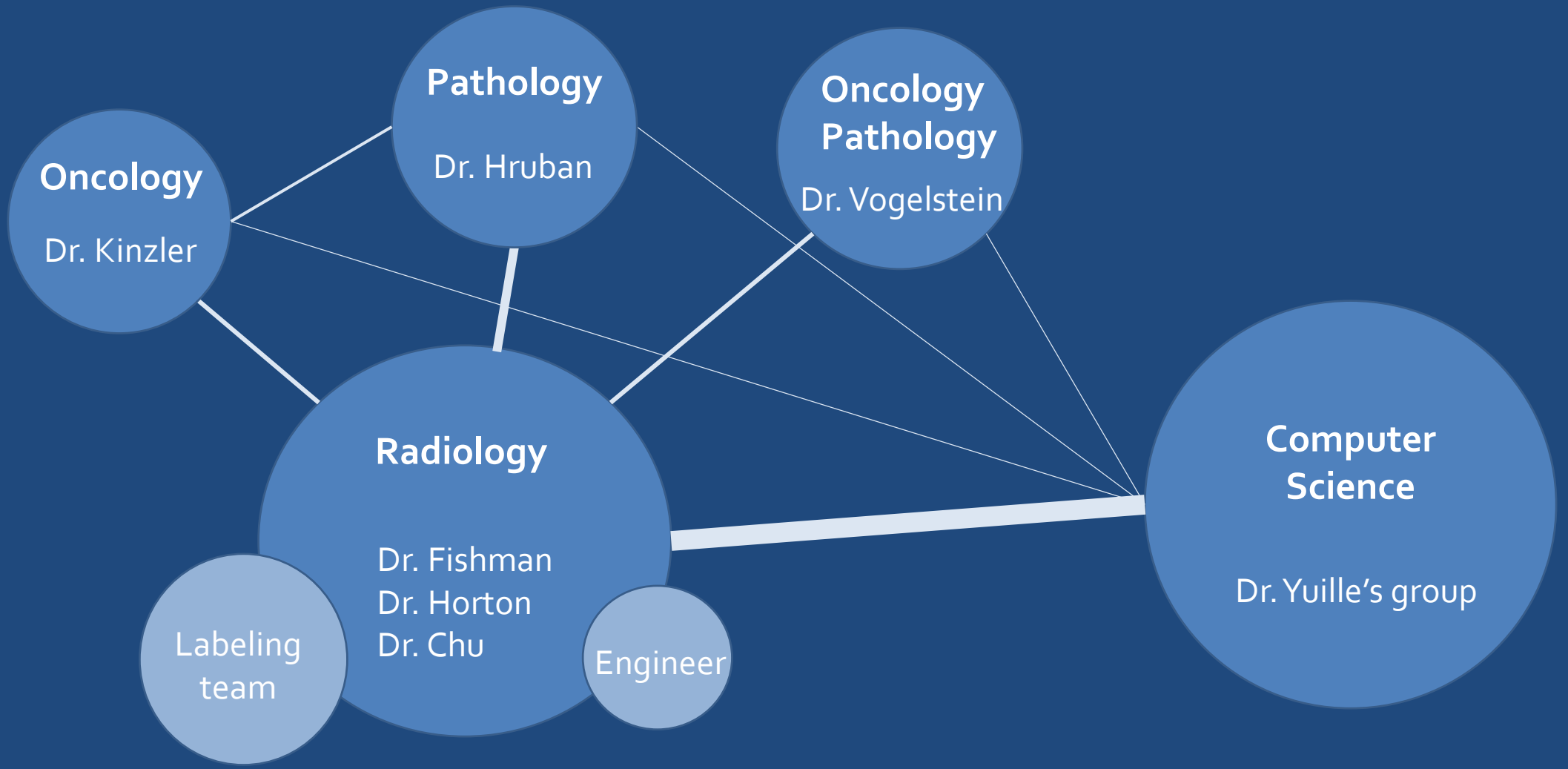

Medical Image Data


Testing Results


Professional Diagnosis

Part 1. Example

The Team



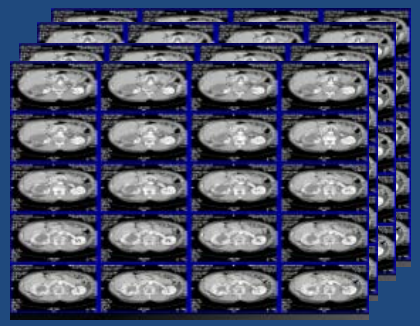
Medical School

Engineering School

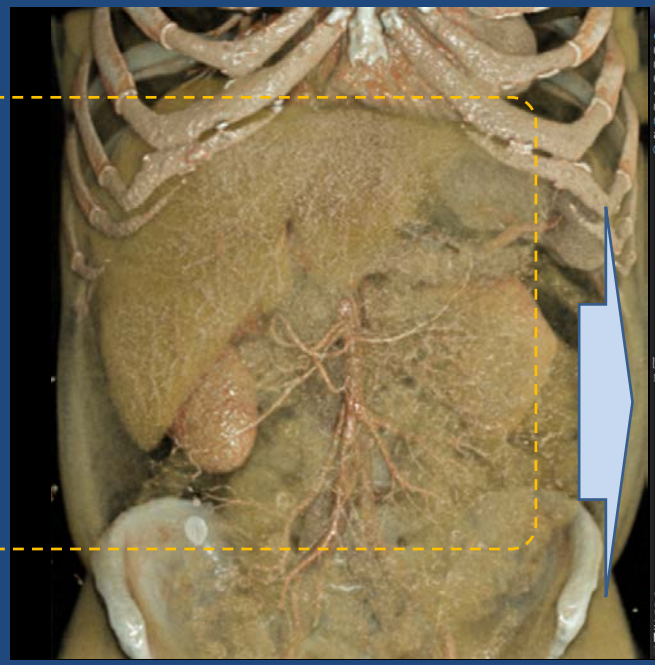
Part 1. Example

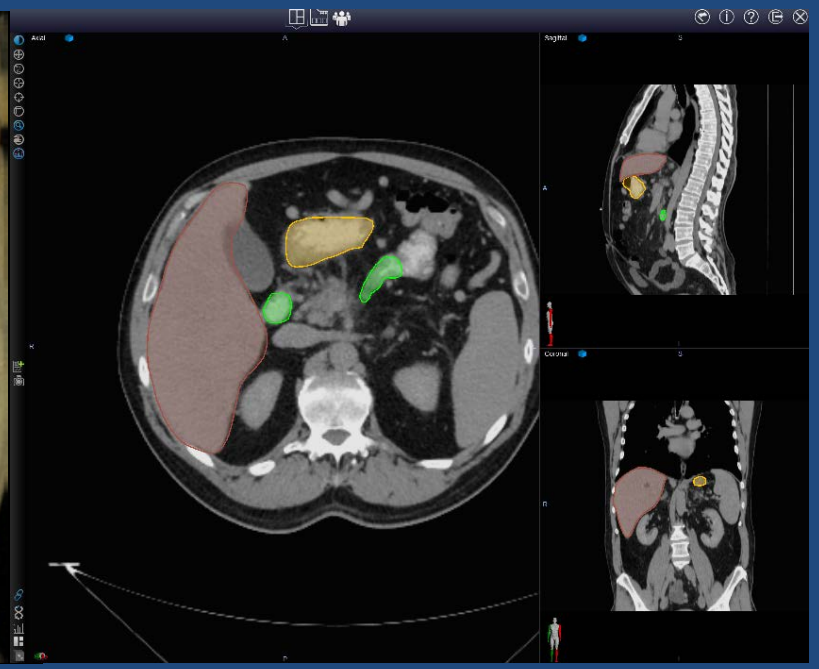
Data collection/annotation

CT Images



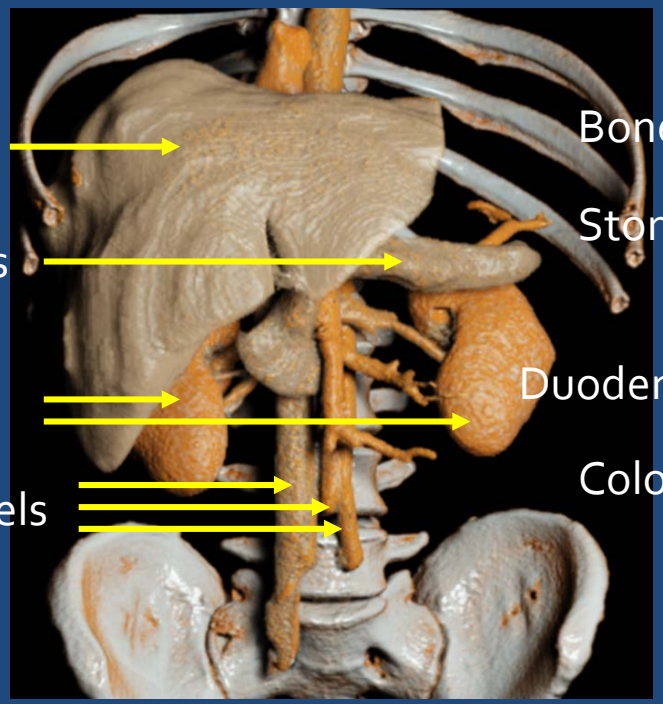
Abdominal region

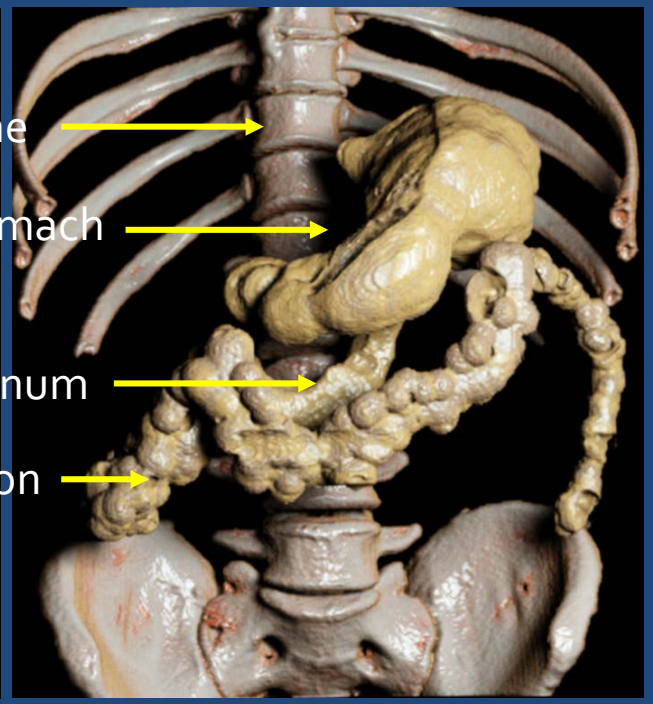


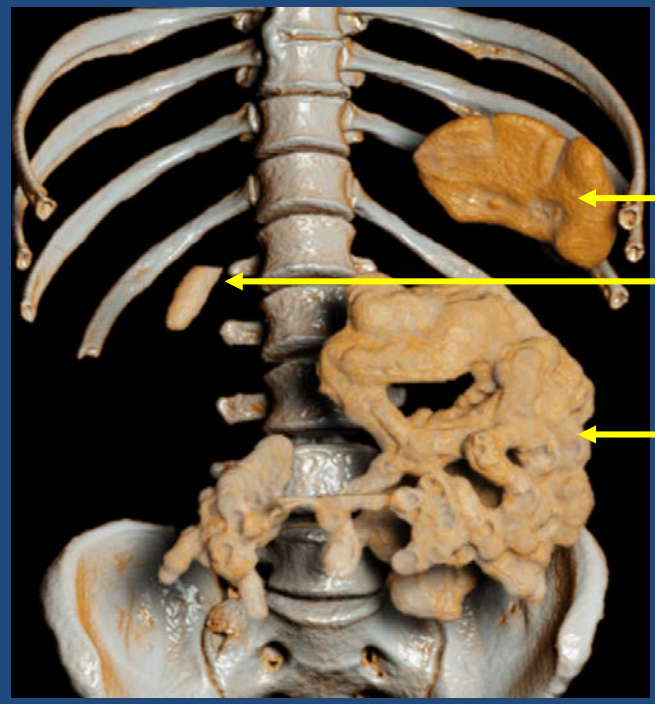


Manual segmentation

Annotated abdominal organs

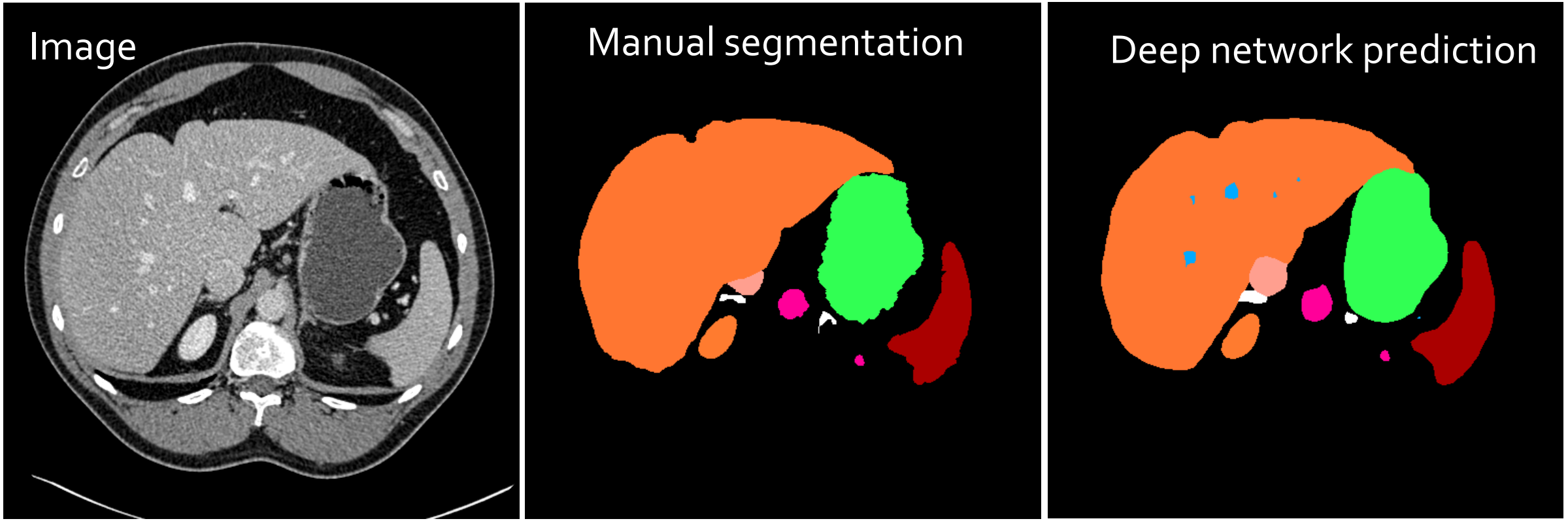






Part 1. Example

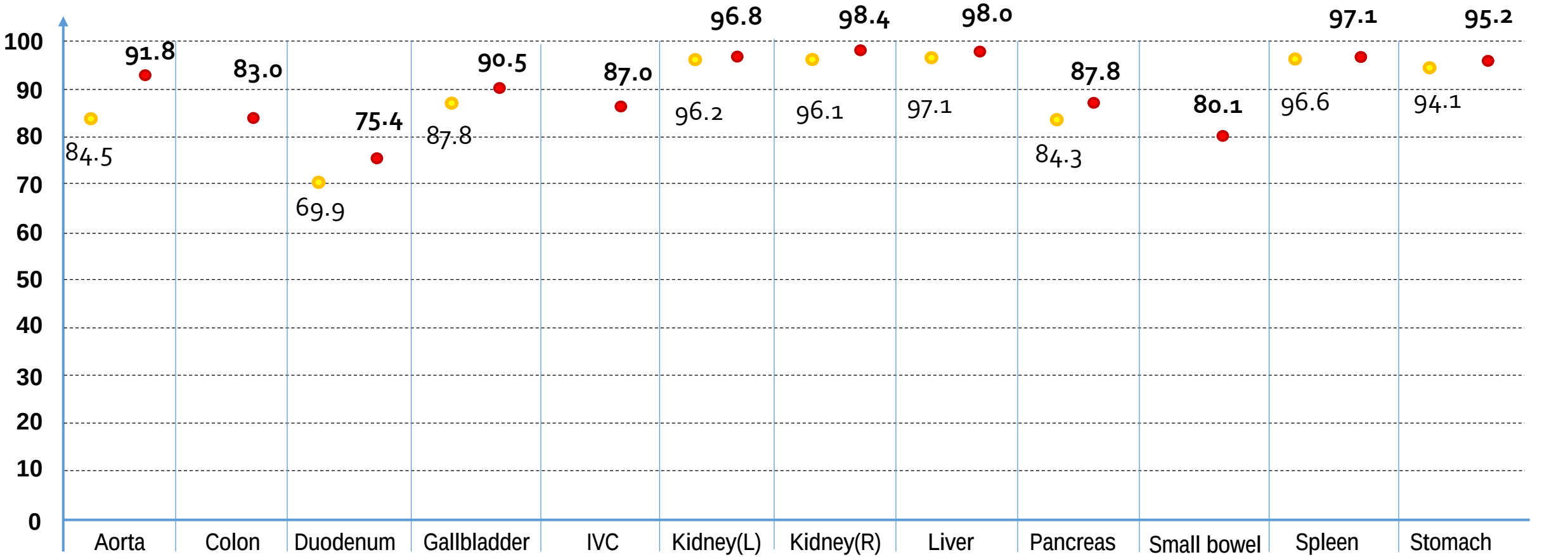
FELIX Multi-organ segmentation



Aorta
Adrenal Gland
Celiac AA
Colon
Duodenum
Gallbladder
IVC
Kidney L
Kidney R
Liver
Pancreas
SMA
Small Bowel
Spleen
Stomach
Veins

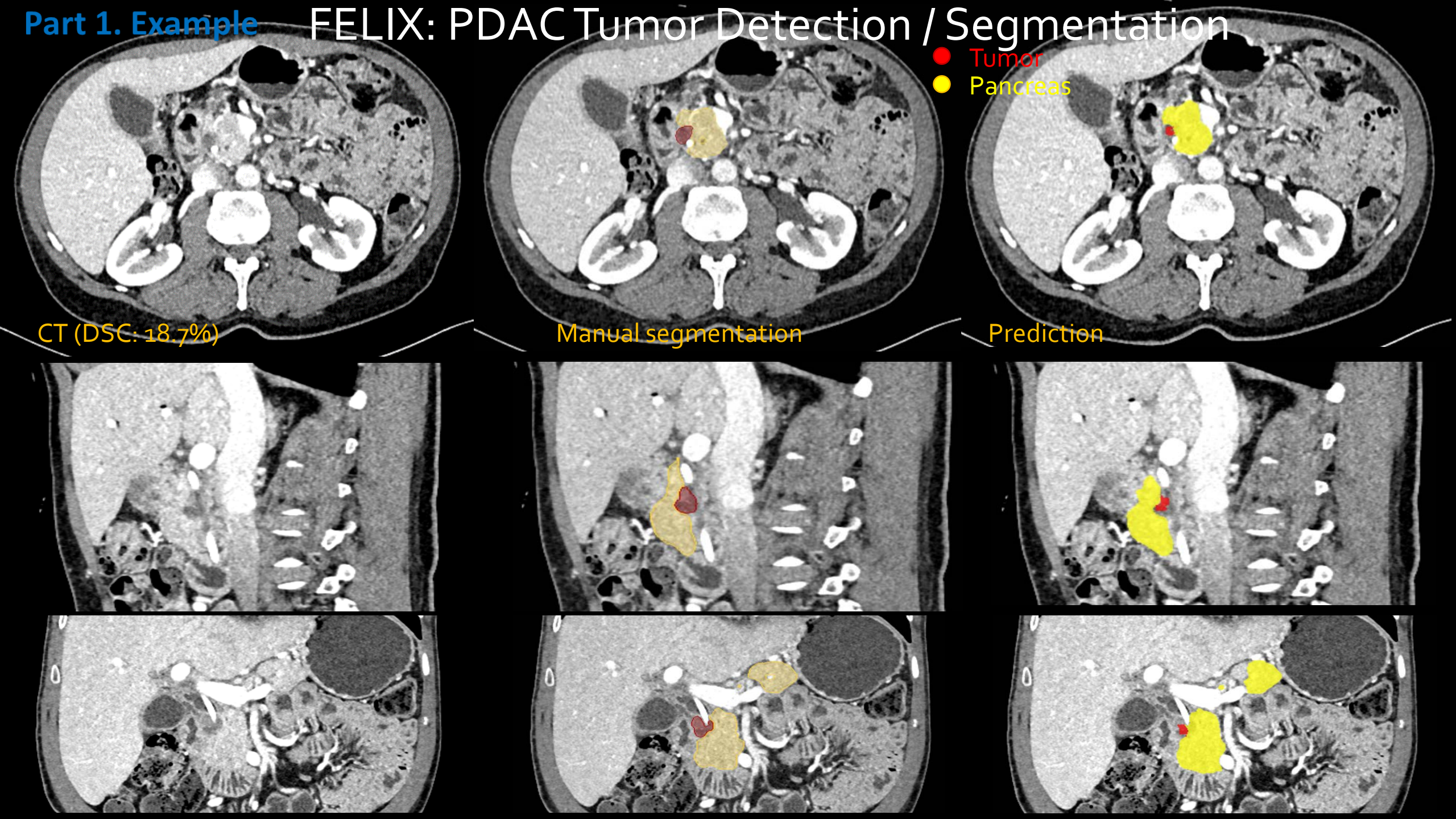
Part 1. Example

FELIX Multi-organ segmentation



Part 1. Example

FELIX: PDAC Tumor Detection / Segmentation



FELIX: Detecting Cancer Tumors PDACs

- Strategy: Detection by Segmentation.
- Segment the healthy Pancreas, the tumors (PDACs), the dilated ducts. Makes predictions for where the Radiologist should pay attention.
- Quality of segmentation decreases for tumors and dilated ducts – DICE scores 60-70% for tumors, 50-60% for dilated ducts.
- But classification performance has extremely high sensitivity and specificity.
- False positives: “focal fat”, lack of accuracy in detecting dilated ducts.
- False negatives: PDACs on the borders of the dataset (e.g., very small, location).

Several Parts

- Part 1. Example
- Part 2. Why are Deep Networks Deep? GUNN.
- Part 3. Combining Deep Networks with Random Forests
- Part 4. Few-Shot Learning
- Part 5. Unsupervised Deep Networks.
- Part 6. Attacking Deep Nets.
- Part 7. When is Big Data not enough?

Gradually Updated Neural Networks (GUNN)

- Deep Networks keep getting deeper? Why?
  - From 5 layers in AlexNet to 1001 layers in ResNet.

Siyuan Qiao

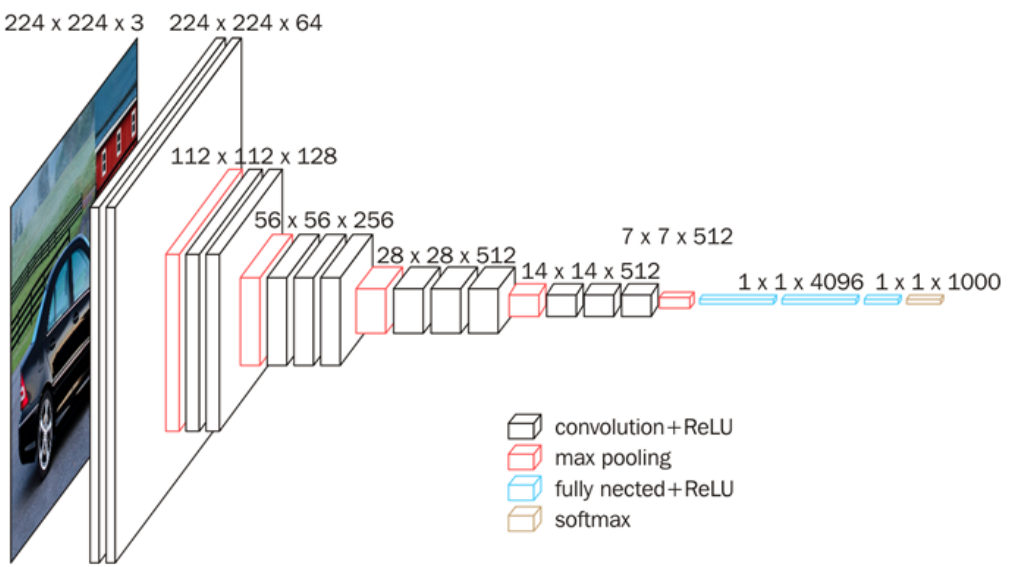
- Why are more layers good? Why is depth good?

- **Intuition:** many layers are good because each new layer introduces more nonlinearities (e.g., ReLu's) and increases the receptive field size, which improves performance.

*S. Qiao, Z. Zhang, W. Shen, B. Wang, and A. Yuille. ICML 2018.

The Standard Deep Net Design

- Deep Nets, e.g., the VGG-network (right) consist of a cascade of convolutional layers.
- Each layer consists of a set of channels.
- All channels in a layer are updated simultaneously based on input from the channels in the previous layer.
- Neurons in all channels have same number of non-linearities (e.g., one) and same receptive field sizes.



Why use the Standard Design?

- Suppose that the goal of Deep Nets is to have large "effective depth" with many nonlinearities and a large range of receptive field sizes.
- **Can we get large effective depth without needing many layers?**
- *GUNN presents an alternative way to get large effective depth, with only a small number of layers, by ordering the channels and updating them gradually, so that channels can receive input from their own layer.*
  - This greatly increases the number of nonlinearities and the sizes of the receptive fields when we add a new layer.

Gradually Updated Neural Networks

- Standard DNN: All channels are updated simultaneously.
  - All channels have the same number of non-linearities and receptive field sizes.
- GUNN: The later channels receive input from the earlier updated channels
  - They have more non-linearities and bigger receptive fields.

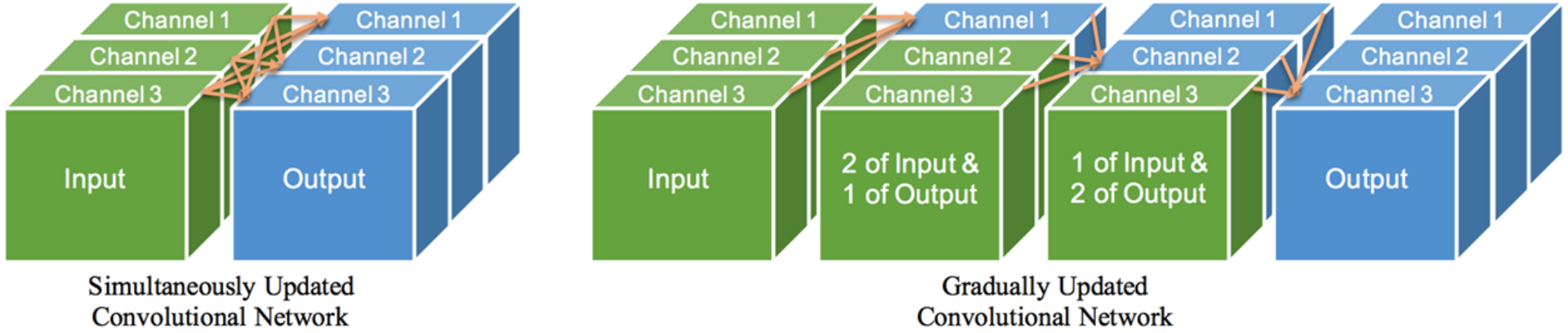


Figure 2. Comparing Simultaneously Updated Convolutional Network and Gradually Updated Convolutional Network. Left is a traditional convolutional network with three channels in both the input and the output. Right is our proposed convolutional network which decomposes the original computation into three sequential channel-wise convolutional operations. In our proposed GUNN-based architectures, the updates are done by *residual learning* (He et al., 2016a), which we do not show in this figure.

GUNN reduces symmetry and overlap singularities

- Deep Nets have hidden symmetries.
  - I.e., there are many equivalent ways to represent the same input output function.
  - **This is wasteful. Maybe Deep Nets are badly designed?**

- *These hidden symmetries makes their energy landscape complex, giving many equivalent minima which are separated by saddle points.*
  - *In particular, there are overlap singularities where two neurons "collapse" (compute the same function) which causes delays in learning.*

- By contrast, GUNN has much fewer symmetries:
  - Because the channels have different non-linearities, and it can be mathematically proven (see paper) that overlap singularities are greatly reduced because it is much harder for two neurons in GUNN to collapse.

GUNN reduces overlap singularities

- This leads to improved learning

➔Faster convergence. Better performance.

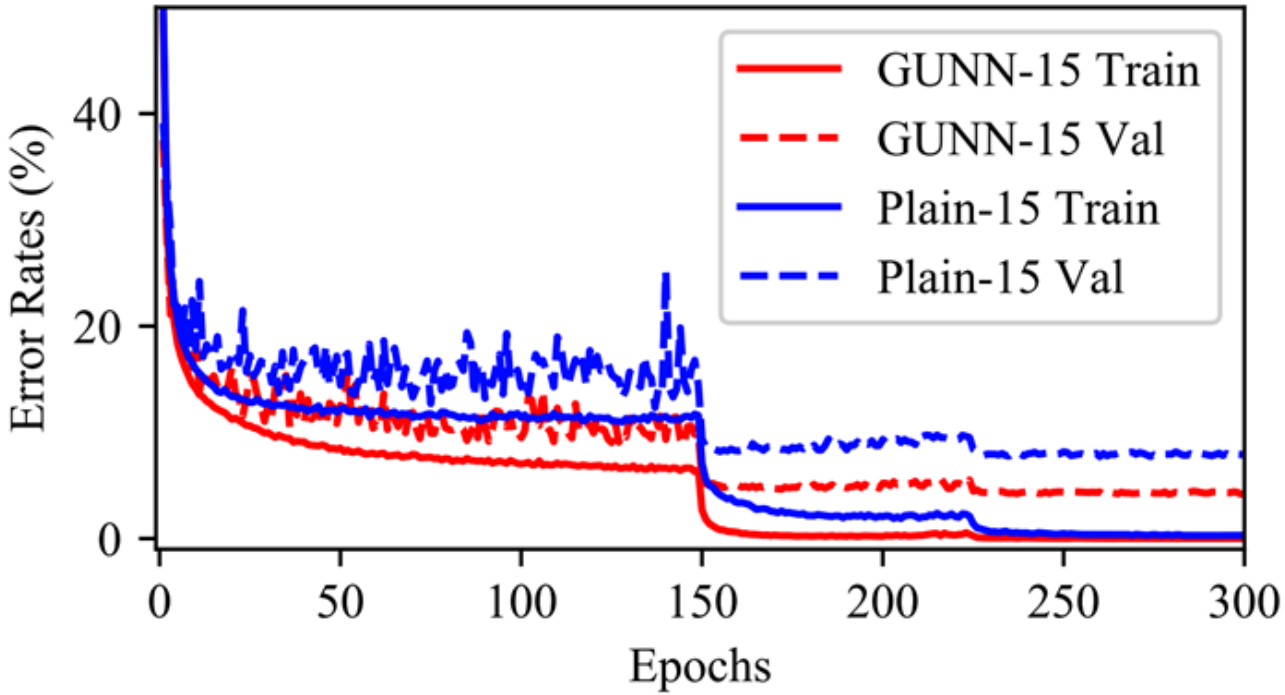


Figure 3. Training dynamics on CIFAR-10 dataset.

GUNN: Results on ImageNet

- Single-crop classification errors (%) on the ImageNet validation set. The test size of all the methods is 224 × 224. (*indicates GUNN.)

Method	# layers	# params	top-1	top-5
VGG-16 (Simonyan & Zisserman, 2014)	16	138M	28.5	9.9
ResNet-50 (He et al., 2016a)	50	25.6M	24.0	7.0
ResNeXt-50 (Xie et al., 2017)	50	25.0M	22.2	6.0
DenseNet-264 (Huang et al., 2017b)	264	33.3M	22.15	6.12
SUNN-18*	18	28.9M	26.16	8.48
GUNN-18*	18	28.9M	**21.65**	**5.87**
ResNet-101 (He et al., 2016a)	101	44.5M	22.0	6.0
ResNeXt-101 (Xie et al., 2017)	101	44.1M	21.2	5.6
DPN-98 (Chen et al., 2017)	98	37.7M	20.73	5.37
SE-ResNeXt-101 (Hu et al., 2017)	101	49.0M	20.70	**5.01**
Wide GUNN-18*	18	45.6M	**20.59**	5.52

Several Parts

- Part 1. Examples
- Part 2. Why are Deep Networks Deep? GUNN
- **Part 3. Combining Deep Networks with Random Forests**
- Part 4. Few-Shot Learning
- Part 5. Unsupervised Deep Networks
- Part 4. Attacking Deep Nets
- Part 5. When is Big Data not enough?

Part 3: Random Forests and Deep Networks

- Random Forests are a very successful for classification.
  - This is like the game of twenty questions.
  - You ask a sequence of questions to get the answer.
  - Particularly useful if the data is inhomogeneous.

Wei Shen

- But what are the questions for an image?
- We can use deep networks to ask the questions.
  - Application: Age estimation, but can be applied to survival analysis in medicine, and many others.

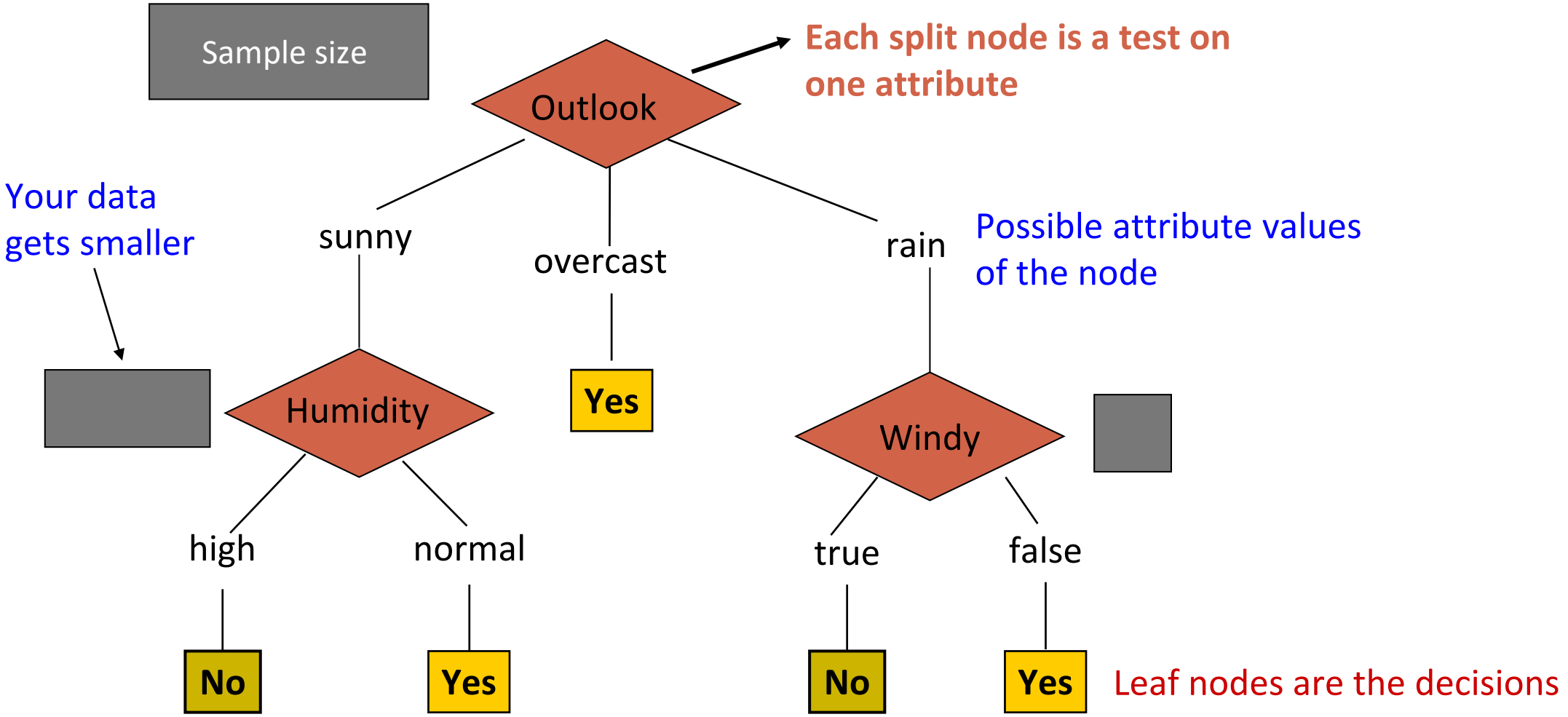
- Deep Networks can be combined with random forests.

[1] W Shen et al. Label Distribution Learning Forests. NIPS. 2017.

[2] W. Shen et al. Deep Regression Forests for Age Estimation. CVPR 2018.

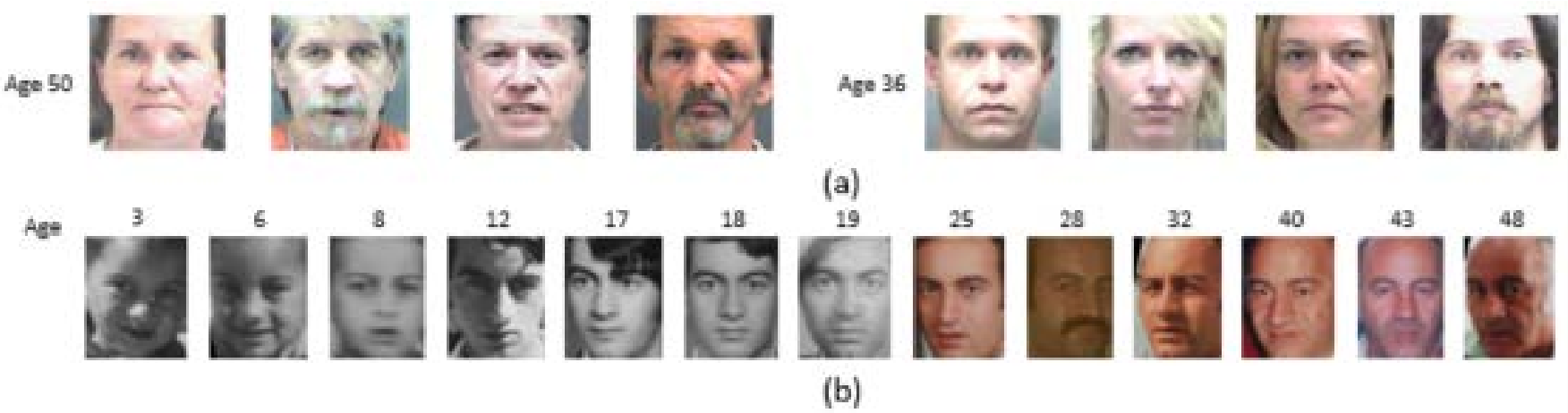
A random forest is made from Decision Trees

- **A Decision Tree for To 'play tennis' or not?**



Application: Age Regression

- How old are they? Both are 40!



- It is hard to estimate people's age from facial images.
- Inhomogeneous Data: large variations in facial appearance between people of same age. Human faces mature at different rates: bone growth, skin wrinkles, etc.
- We combine deep networks with random regression forests.

Deep Regression Forests (DRFs)

- We propose Deep Regression Forests (DRFs) – a novel regression algorithm inspired by differentiable decision trees (Kontschnieder et al. 2015).
  - Differentiable regression forests partitions the data so that each leaf node only has to learn a simple regression function.

- This enables the Deep Regression Forest to learn complex regression functions.

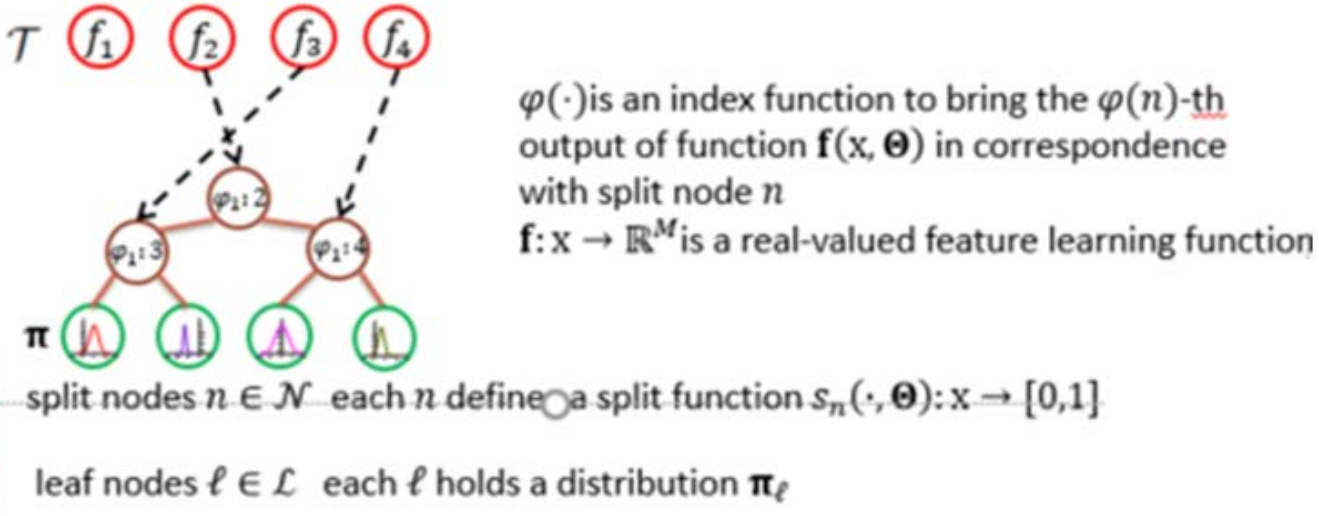
- The partition is "soft" because the decision trees (→ the random forests) are differentiable w.r.t. the parameters of the trees.

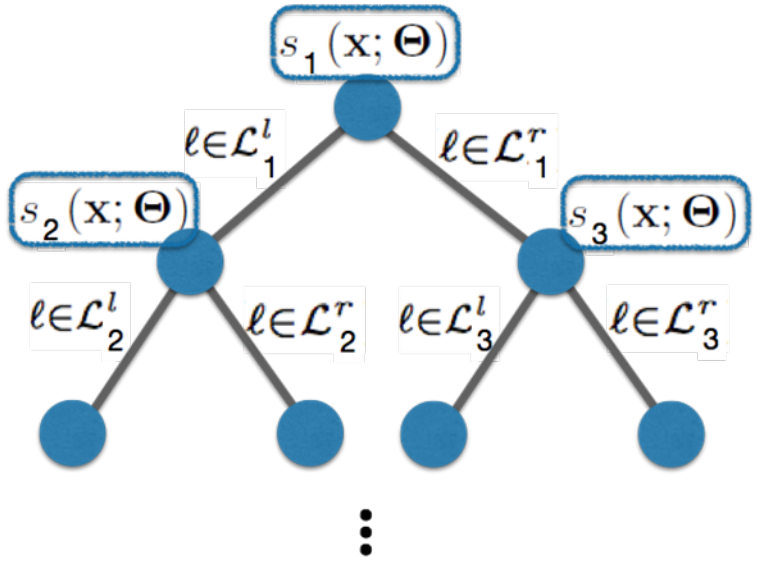
- The differentiable decision trees (→random forests) can be learnt together with the convolutional layers.

JOHNS HOPKINS UNIVERSITY

Learning a Differentiable Regression Tree

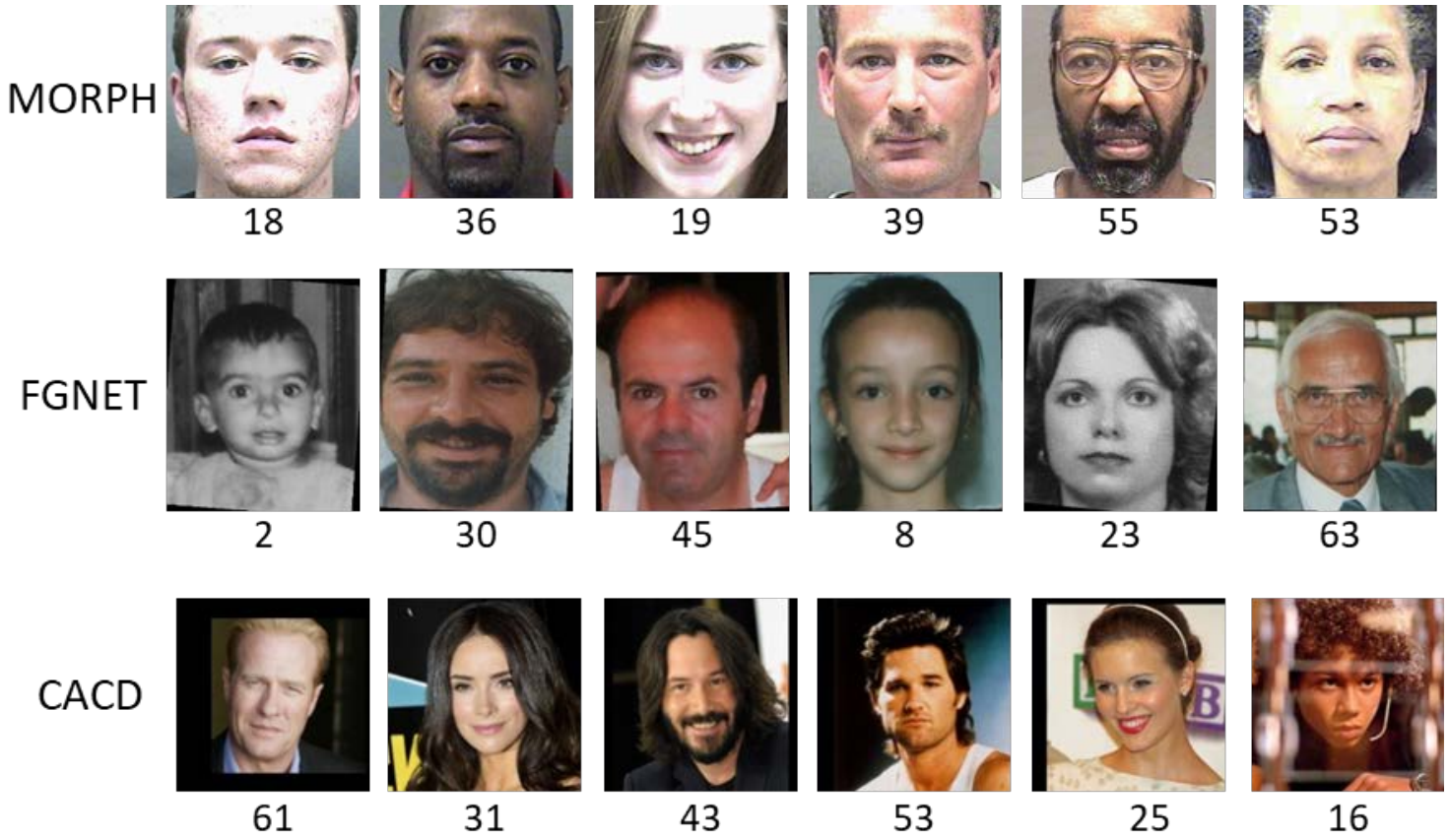
- Our goal is to learn a probability distribution $p(\mathbf{y}|\mathbf{x};\mathcal{T}) = \sum_{\ell \in \mathcal{L}} P(\ell|\mathbf{x};\Theta)\pi_\ell(\mathbf{y})$
- $P(\ell|\mathbf{x};\Theta)$ is the probability of reaching leaf node l.
- It depends on parameters Θ that are the weights of deep networks and determine the probability of splits of the differentiable decision tree.
- $\pi_\ell(\mathbf{y})$ is the regression probability at each leaf node.
- We learn Θ and $\pi_\ell(\mathbf{y})$.





Experimental Results

- Dataset
  - Morph: 55,000 images from about 13,000 people
  - FGNET: 1002 facial images of 82 individuals
  - CACD: 160,000 facial images of 2,000 celebrities



Summary Deep Regression Forests

- Deep nets can be combined with random forests for regression, thereby combining the strengths of both methods.

- This builds on differentiable decision trees (Kontscheider et al. 2015).
  - *Technically, their work can be extended to this richer class of problems – label distribution learning and regression – by observing that their update algorithm is a special case of variational bounding.*

- This leads to state-of-the-art results on age regression and can be applied to many other problems.

Several Parts

- Part 1. Examples
- Part 2. Why are Deep Networks Deep? GUNN
- Part 3. Combining Deep Networks with Random Forests
- **Part 4. Few-Shot Learning**
- Part 5. Unsupervised Deep Networks
- Part 6. Attacking Deep Nets
- Part 7. When is Big Data not enough?

Part 4. Few-Shot Learning

- What if there is too little data available for training? This is called few-shot learning.

- Exploit many-shot learning (standard learning) on a big dataset and few-shot on small dataset.
- For example, we know how to recognize many objects – pug, jay, hen – and then we want to learn a new object – snail, corgi (Queen of England's dogs) from a few examples (1 or 5).

- This is probably how children learn.
  - They take a lot of time to start learning.
  - But after they have learnt a critical amount, they learn very fast from very few examples.

*Siyuan Qiao et al. Few-Shot Image Recognition…. CVPR. 2018.

Few-Shot Learning for Recognizing Objects

- Training the weights (parameters) for many-shot object categories – e.g., pug, jay, hen-- in a large dataset $\mathcal{D}_{\text{large}}$ by Deep Networks.

- It is hard to train the weights for the few-shot categories -- snail, corgi -- in the small dataset $\mathcal{D}_{\text{few}}$.

- We cannot learn the weights directly because there is not enough data.

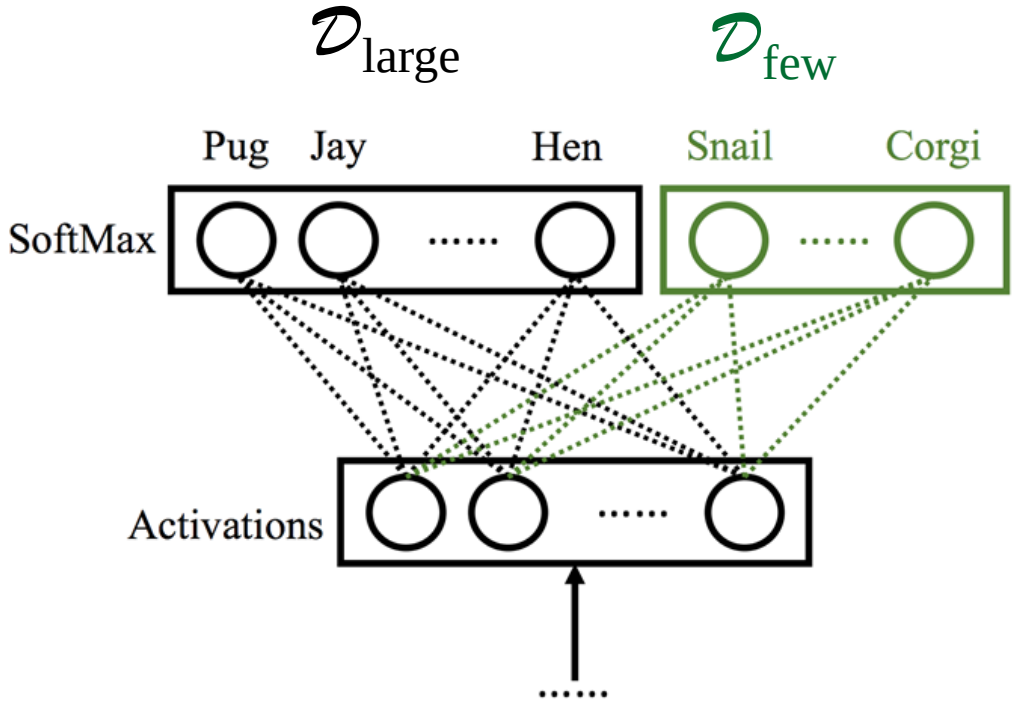


Figure 1: Illustration of pre-training on $\mathcal{D}_{\text{large}}$ (black) and few-shot novel category adaptation to $\mathcal{D}_{\text{few}}$ (green). The green circles are the novel categories, and the green lines represent the unknown parameters for categories in C_{few}.

Few-Shot Learning

- What we know:
  i. Activations of the penultimate Deep Net layers for many-shot categories in $\mathcal{D}_{\text{large}}$
  ii. The final weights (parameters) for the multi-shot categories in $\mathcal{D}_{\text{large}}$
  iii. The activations of the final Deep Net for the few-shot categories in $\mathcal{D}_{\text{few}}$

- What we need -- the parameters for the few-shot categories.

We need the green arrows.

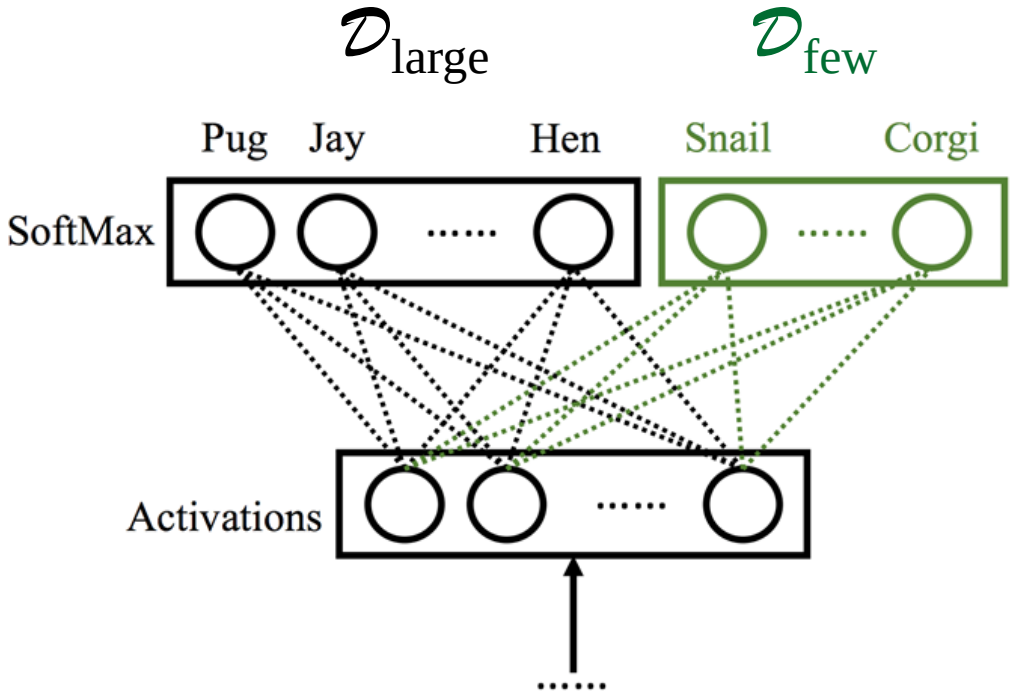
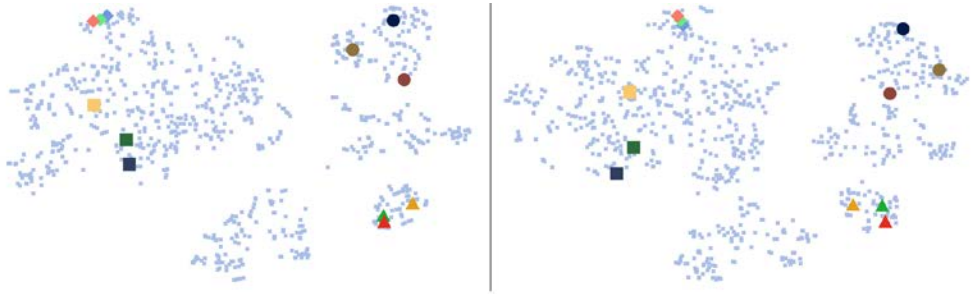


Figure 1: Illustration of pre-training on $\mathcal{D}_{\text{large}}$ (black) and few-shot novel category adaptation to $\mathcal{D}_{\text{few}}$ (green). The green circles are the novel categories, and the green lines represent the unknown parameters for categories in C_{few}.

Key Idea: Relate the activations to the weights

- We plot the t-SNE of activations (left) and weights (right).
- These plots look very similar. This is surprising, may give insight into Deep Networks.
- This suggests that there is a mapping function from activations to weights. We learn this mapping by modelling it as a neural network.
- In other words, we train a neural network to predict the parameters of another neural network that classifies the few-shot categories.



Few-Shot Learning Summary

- There is an empirical relationship between the activations of the deep network features at the penultimate level and the weights of the decision layer of the network.

- *This relationship can be exploited for few-shot learning.*
  - *We learn the relationship from the big dataset, where we have many examples of each category, and apply this relationship to estimate the final weights for the categories in the small dataset.*

- This significantly improves the state-of-the-art on several few-shot datasets. E.g., ImageNet with 900 multi-shot objects and 100 few-shot objects.

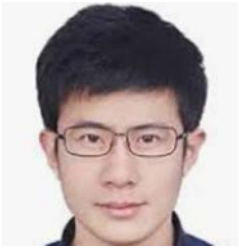
Several Parts

- Part 1. Examples
- Part 2. Why are Deep Networks Deep? GUNN
- Part 3. Combining Deep Networks with Random Forests
- Part 4. Few-Shot Learning
- **Part 5. Unsupervised Deep Networks**
- Part 6. Attacking Deep Networks
- Part 7. When is Big Data not enough?

Unsupervised Deep Networks

- Annotation is very hard for some visual tasks – e.g., optical flow and depth estimation.
- For people interested in biology – it seems impossible that humans/primates learn from fully supervised data.
- But recent work trains Deep Networks – e.g., for optical flow – by only requiring coarse statistics of the optical flow – e.g., local smoothness. Loss function depends on statistics of flow within local neighborhoods.
- Similarly can train Deep Networks without supervision to estimate rigid depth from ego-motion.
- ***In short – uses 1980's computer vision to train Deep Networks.***

Unsupervised Deep Networks



- Recent work Chenxu Luo et al (Baidu-JHU collaboration). Peng Wang.
- Use unsupervised Deep Networks to learn, and learn to combine, different cues:
- (1) Optical Flow
- (2) Structure from Rigid Motion
- (3) Shape-from-X
- Use a Holistic Motion Parser (HMP) to combine these cues – intuition 3D depth makes prediction for optical flow, inconsistency gives “dynamic objects”, structure from motion helps train shape-from-X.

Unsupervised Deep Networks

- Input: Image Sequences without Annotation.
- Output: Estimated Depth. Estimated Optical Flow. Estimated Object Model Mask.
- Evaluation: KITTI dataset – cars driving in streets.

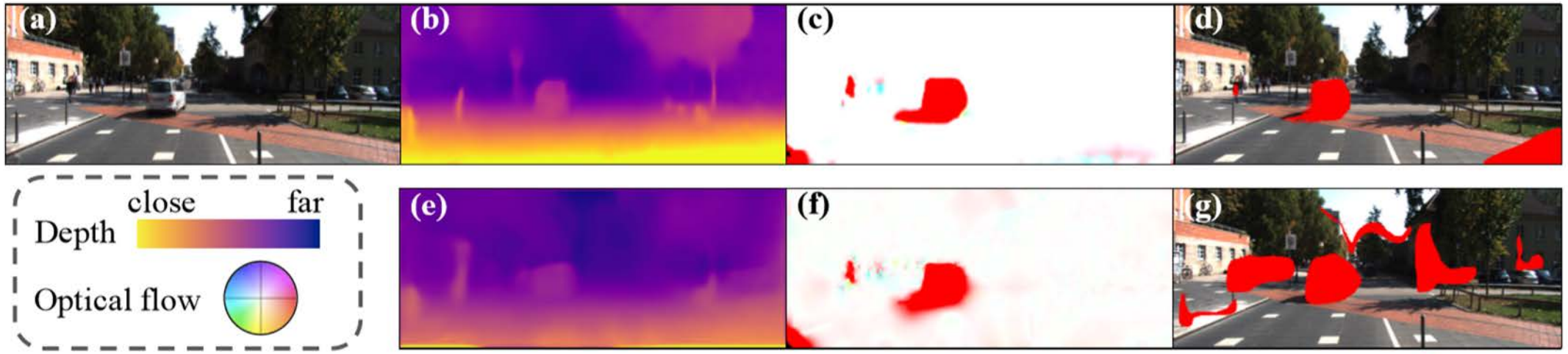


Fig. 1: (a) image, (b) our estimated depth, (c) our estimated optical flow, (d) our moving object mask, (e) depth from Yang *et al.* [5], (f) optical flow from Wang *et al.* [6], (g) segmentation mask from Yang *et al.* [7]. We show significant improvement of all tasks over other SOTA methods.

Unsupervised Deep Networks

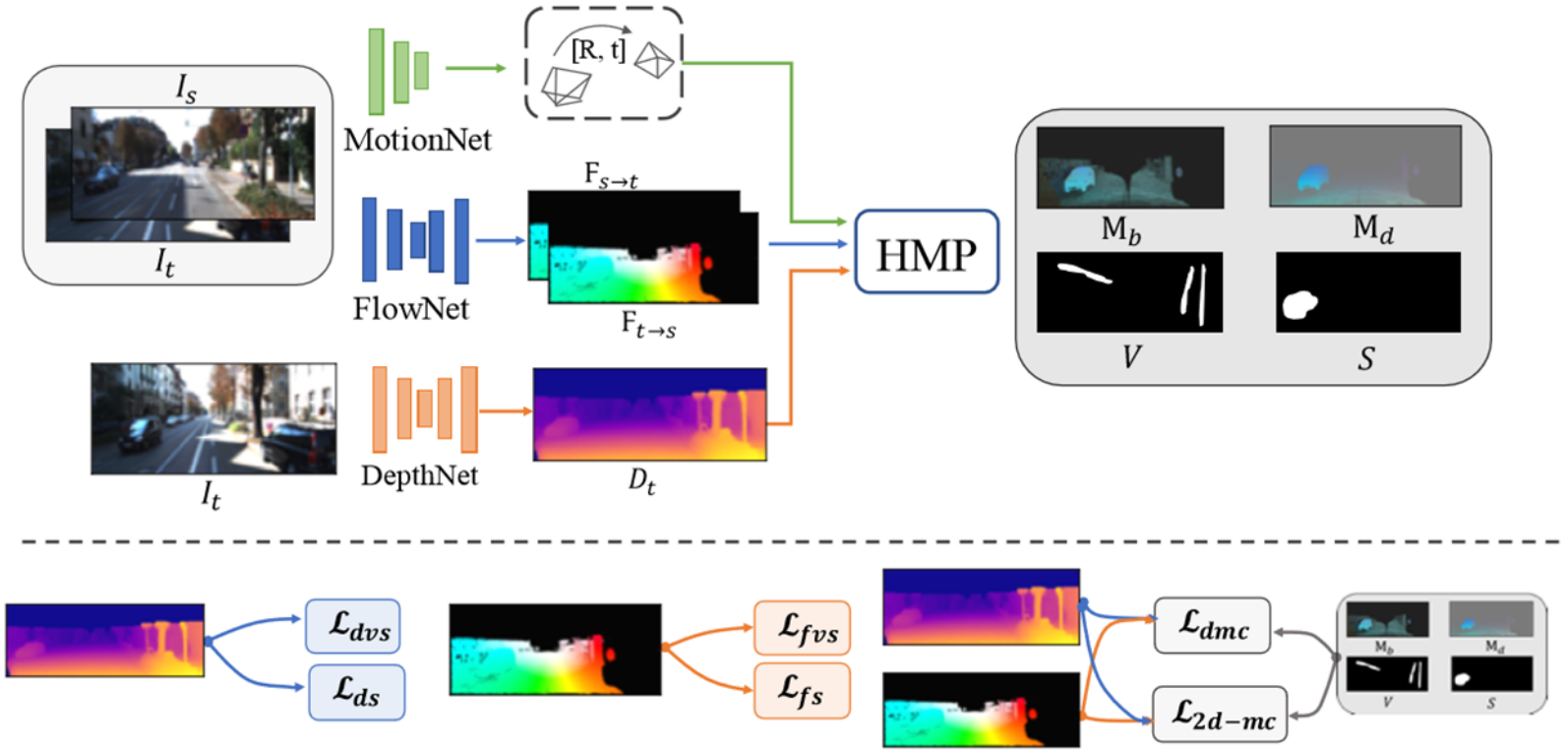


Fig. 2: Pipeline of our framework. Given a pair of consecutive frames, *i.e.*target image I_t and source image I_s, a FlowNet is used to predict optical flow $\mathbf{F}$ from I_t to I_s. Notice here FlowNet is not the one in [17]. A MotionNet predicts their relative camera pose $\mathbf{T}_{t\to s}$. A single view DepthNet estimates their depths $\mathbf{D}_t$ and $\mathbf{D}_s$ independently. All the informations are put into our Holistic 3D Motion Parser (HMP), which produce an segmentation mask for moving object $\mathbf{S}$, occlusion mask, 3D motion maps for rigid background $\mathbf{M}_s$ and dynamic objects $\mathbf{M}_d$. Finally, we apply corresponding loss over each of them. Corresponding loss are added afterwards for training different networks. (Details in Sec. 3.2.2)

Unsupervised Deep Networks: Summary

- Can use simple 1980's computer vision models to train Deep Networks. (Smirnakis and Yuille 1994 – similar, but no follow-up).
- Combining different visual cues consistently ("holistically") leads to strong performance.
- Suggests strategies that a visual system interacting with the 3D world might be able to bootstrap itself by unsupervised learning exploiting simple assumptions about the world.
- (David Mumford speculation – in Knill and Richards 1996)

Several Parts

- Part 1. Examples
- Part 2. Why are Deep Networks Deep? GUNN
- Part 3. Combining Deep Networks with Random Forests
- Part 4. Few-Shot Learning
- Part 5. Unsupervised Deep Networks
- **Part 6. Attacking Deep Networks**
- Part 7. When is Big Data not enough?

Part 5: Attacking Deep Networks

- Attacking Deep Networks is a way to understand them and to strengthen them.

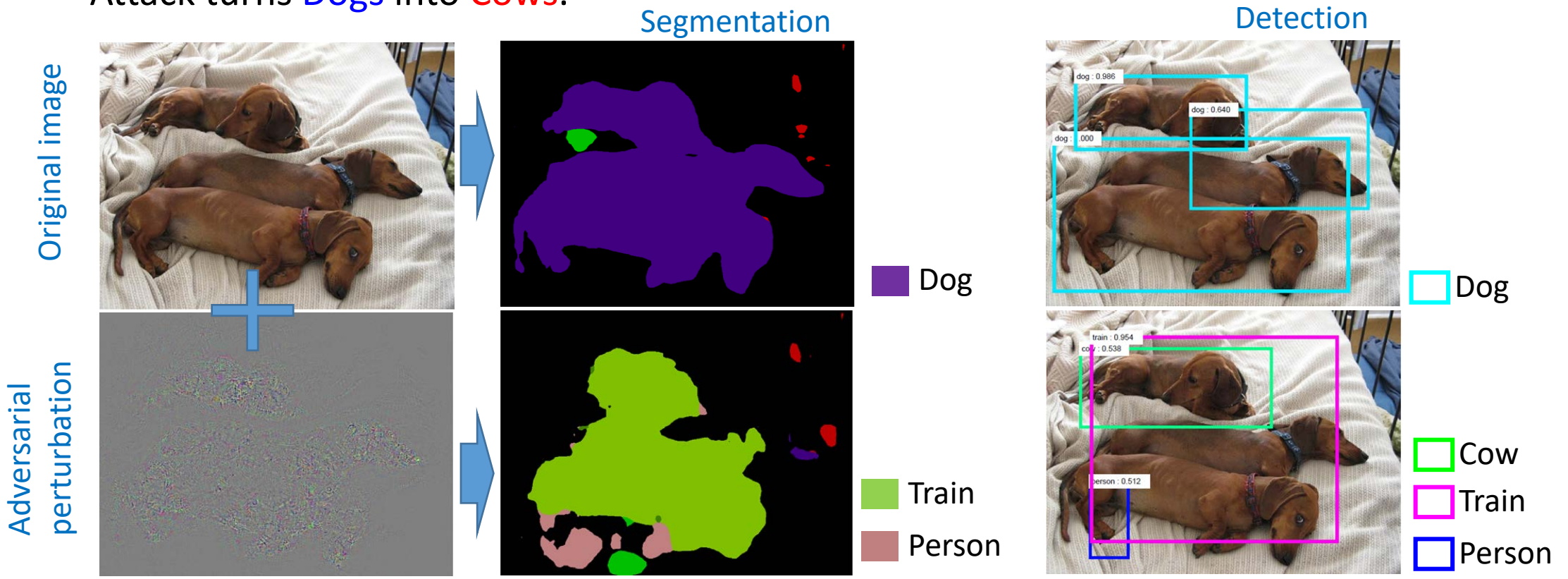
- Distinction: Local Attacks and Structured Attacks.

- Local Attacks. Very small changes to images can cause a deep network to make very big mistakes.

- More Structured Attacks. These are much larger changes to the images. Like partially occluding objects. Or changing the viewpoint.

Attacking Deep Nets for Semantic Segmentation and Object Detection

- Deep Nets attacks are not only for classification.
  - Attack turns Dogs into Cows.

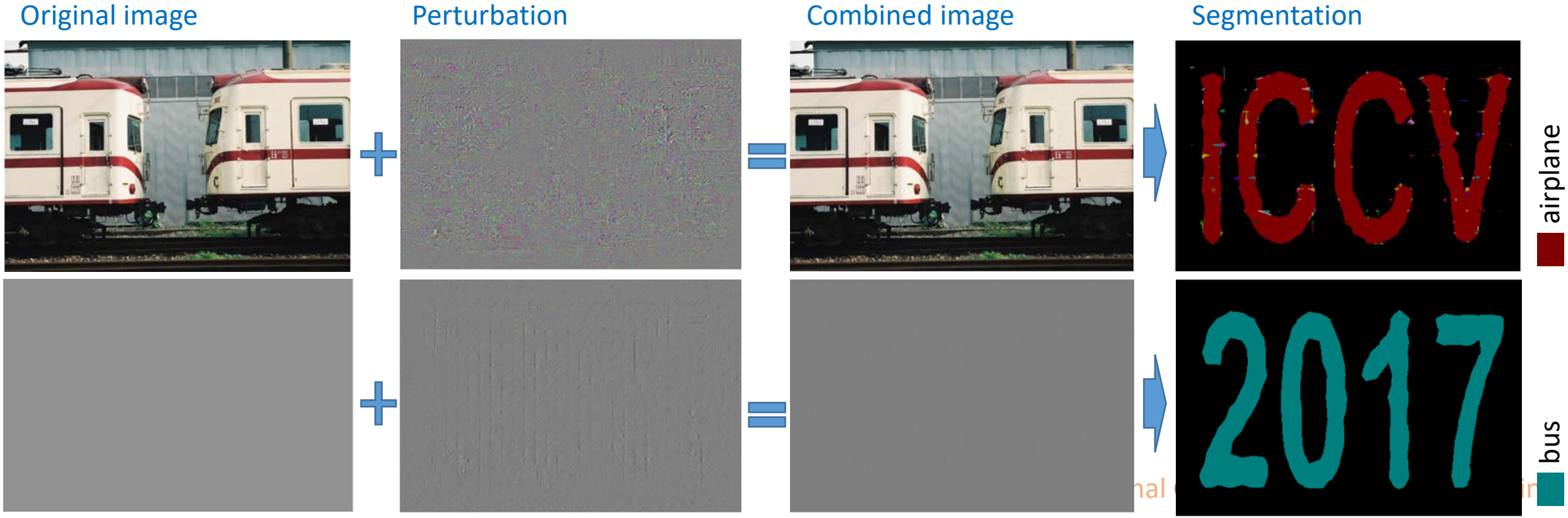


*Cihang Xie, Jianyu Wang, Zhishuai Zhang et al. ICCV 2017.

JOHNS HOPKINS UNIVERSITY

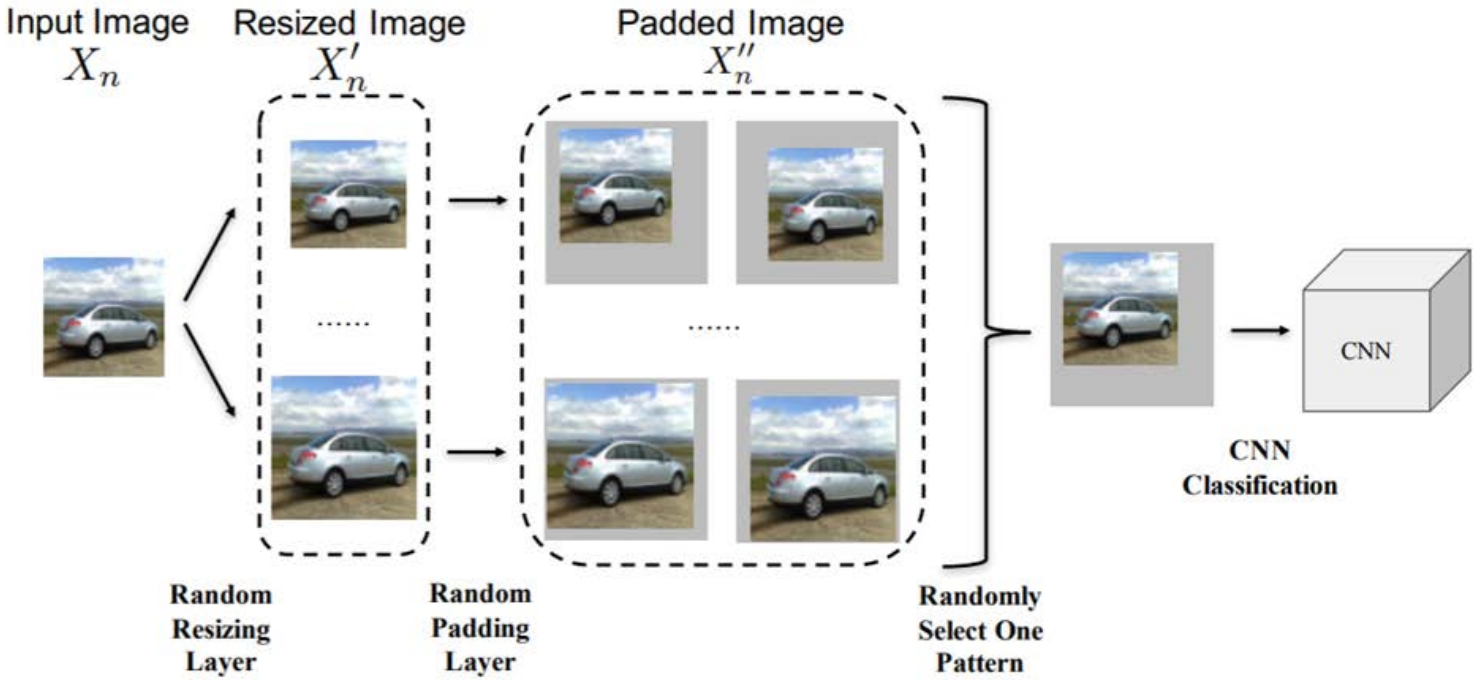
Attacking Deep Nets for Semantic Segmentation and Object Detection

- AN turns Train into Airplane with shape ICCV.
- AN turns blank Image into Bus with shape 2017.



Defending against adversarial noise

- Defense Strategy: defeat adversarial attacks by randomization.
  - Random resizing and random padding.
  - No additional training or fine-tuning, little computation, can be used in conjunction with other defense methods.
- But more recent attacks can beat this defense. A new Arms Race.



*Cihang Xie et al. ICLR 2018.
2nd prize among 110 defense teams in NIPS 2017.

Attacking Deep Networks: Local and Structured

- It is possible that local attacks can be dealt with using defenses which strengthen the Deep Network.
- (E.g., FAIR-JHU team. Cihang Xie et al. 1st place in the Adversarial Defense track of the Competition on Adversarial Attacks and Defenses 2018 (CAAD2018).
- But local adversarial attacks are only the tip of the iceberg.

JOHNS HOPKINS UNIVERSITY

More Structured Attacks: Occlusion.

- You can fool the Deep Net by adding occlusion or changing context.

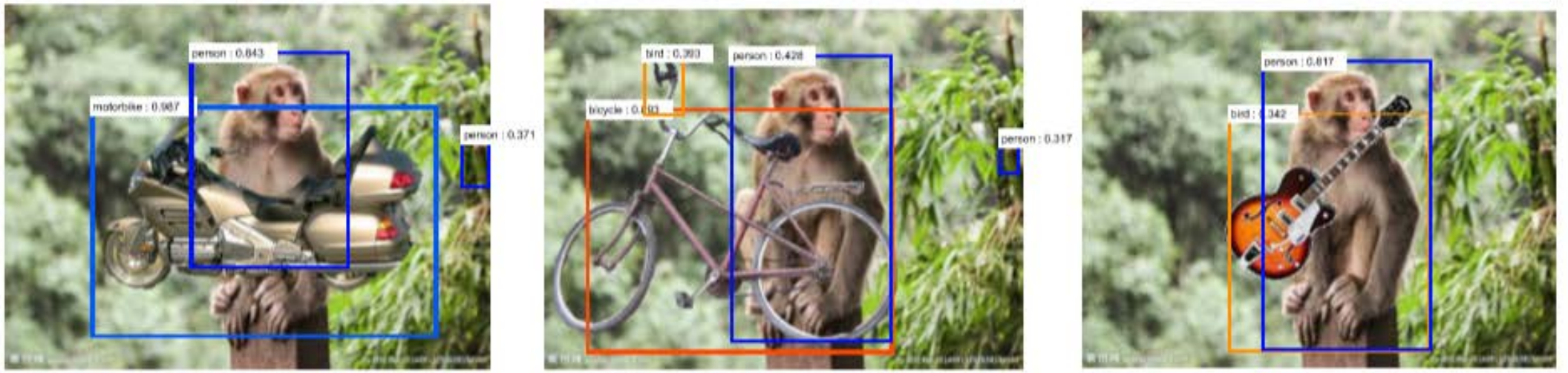


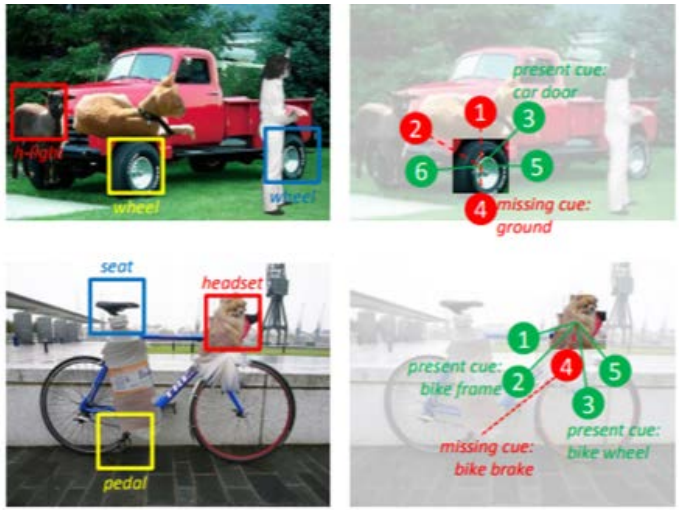
Figure 1: Caption: Adding occluders causes deep network to fail. Left Panel: The occluding motorbike turns a monkey into a human. Center Panel: The occluding bicycle turns a monkey into a human and the jungle turns the bicycle handle into a bird. Right Panel: The occluding guitar turns the monkey into a human and the jungle turns the guitar into a bird.

*Jianyu Wang et al. Annals of Mathematical Sciences. 2018.

JOHNS HOPKINS UNIVERSITY

Vehicle Parts with Occlusion: Defend against Occlusion

- Label Parts (e.g., wheel, headlight) on Vehicles.
- Add occlusion at three levels of complexity.
- *Can we detect the parts even if the models have been trained without any occlusion?*
- Yes (partially): two defenses using visual concepts (BMVC 2017, CVPR



DeepVote

- DeepVote is an end-to-end Deep Network for detecting semantic parts, which is robust to occluders and context.
- DeepVote can exploit context if it is helpful, but switch off context if is not.

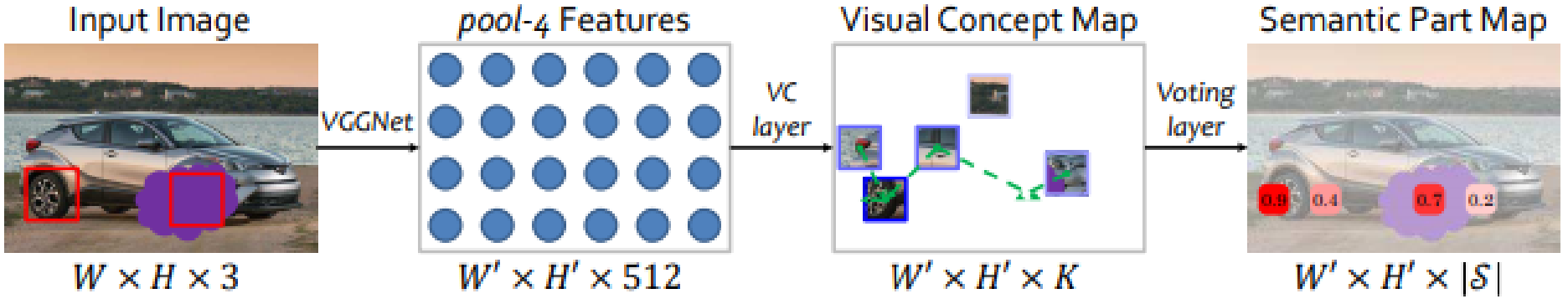


Figure 2. The overall framework of DeepVoting (best viewed in color). A *car* image with two *wheels* (marked by red frames, one of them is occluded) is fed into VGGNet [21], and the intermediate outputs are passed through a visual concept extraction layer and a voting layer. We aggregate local cues from the visual concept map (darker blue indicates more significant cues), consider their spatial relationship to the target semantic part via voting, and obtain a low-resolution map of semantic parts (darker red or a larger number indicates higher confidence). Based on this map, we perform bounding box regression followed by non-maximum suppression to obtain the final results.

*Zhishuai Zhang, Cihang Xie, Jianyu Wang et al. DeepVote... CVPR. 2018.

DeepVote: Results

- DeepVote is equivalent to Faster-RCNN (FR) if there is no occlusion.
- DeepVote significantly outperforms FR when there is occlusion.
- All methods are trained without occluded examples.

	No Occlusions						L1				L2				L3			
Category	**KVC**	**DVC**	**VT**	**FR**	**DV**	**DV+**	**VT**	**FR**	**DV**	**DV+**	**VT**	**FR**	**DV**	**DV+**	**VT**	**FR**	**DV**	**DV+**
airplane	15.8	26.6	30.6	56.9	59.0	**60.2**	23.2	35.4	**40.6**	**40.6**	19.3	27.0	31.4	**32.3**	15.1	20.1	**25.9**	25.4
bicycle	58.0	52.3	77.8	90.6	89.8	**90.8**	71.7	77.0	83.5	**85.2**	66.3	62.0	78.7	**79.6**	54.3	41.1	**63.0**	62.5
bus	23.8	25.1	58.1	**86.3**	78.4	81.3	31.3	55.5	56.9	**65.8**	19.3	40.1	44.1	**54.6**	9.5	25.8	30.8	**40.5**
car	25.2	36.5	63.4	83.9	80.4	**80.6**	35.9	48.8	56.1	**57.3**	23.6	30.9	40.0	**41.7**	13.8	19.8	27.3	**29.4**
motorbike	32.7	29.2	53.4	63.7	65.2	**69.7**	44.1	42.2	51.7	**55.5**	34.7	32.4	41.4	**43.4**	24.1	20.1	29.4	**31.2**
train	12.3	12.8	35.5	59.9	59.4	**61.2**	21.7	30.6	33.6	**43.7**	8.4	17.7	19.8	**29.8**	3.7	10.9	13.3	**22.2**
mean	28.0	30.4	53.1	73.6	72.0	**74.0**	38.0	48.3	53.7	**58.0**	28.6	35.0	42.6	**46.9**	20.1	23.0	31.6	**35.2**

Table 1. Left 6 columns: Comparison of detection accuracy (mean AP, %) of **KVC**, **DVC**, **VT**, **FR**, **DV** and **DV+** without occlusion. Right 12 columns: Comparison of detection accuracy (mean AP, %) of **VT**, **FR**, **DV** and **DV+** when the object is occluded at three different levels. Note that **DV+** is DeepVoting trained with context outside object bounding boxes. See the texts for details.

Explaining DeepVote Results

- DeepVote can partially explain its detection results. I.e. the cues that it has used to detect the semantic parts.
- Preliminary follow-up work (in progress) gives improved interpretability

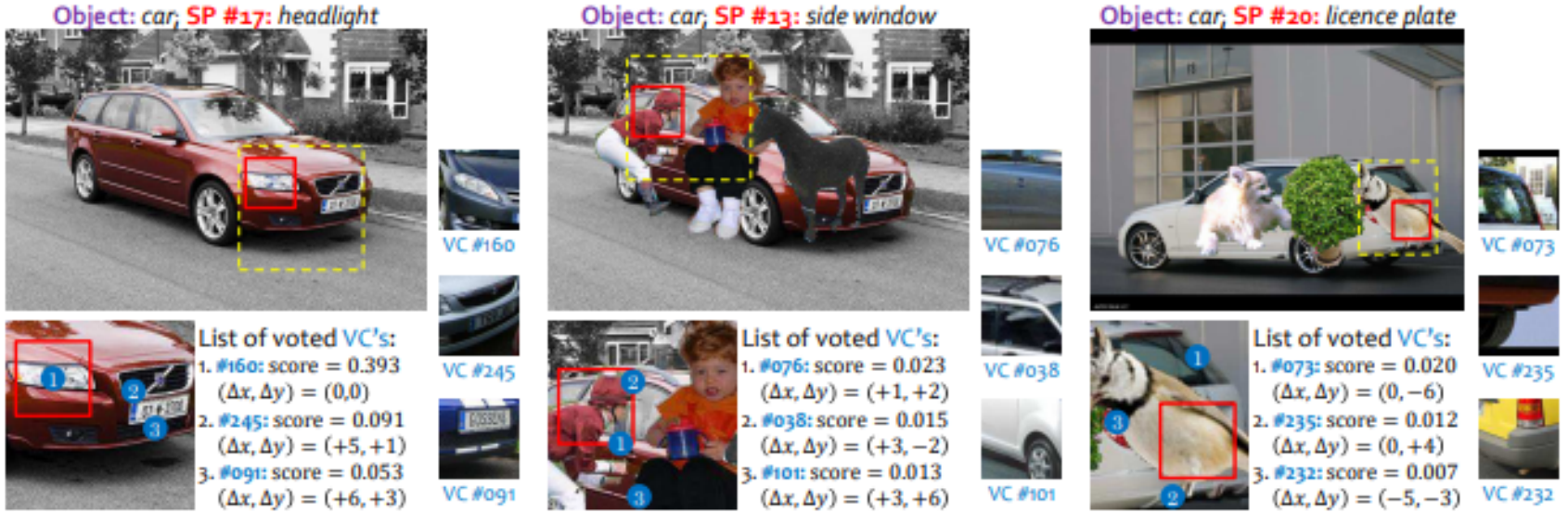


Figure 6. DeepVoting allows us to explain the detection results. In the example of heavy occlusion (the third column), the target semantic part, *i.e.*, the *licence plate* on a *car*, is fully occluded by a *bird*. With the help of some visual concepts (blue dots), especially the 73-rd VC (also displayed in Figure 5), we can infer the position of the occluded semantic part (marked in red). Note that we only plot the 3 VC's with the highest scores, regardless the number of voting VC's can be much larger.

Several Parts

- Part 1. Examples
- Part 2. Why are Deep Networks Deep? GUNN
- Part 3. Combining Deep Networks with Random Forests
- Part 4. Few-Shot Learning
- Part 5. Unsupervised Deep Networks
- Part 6. Attacking Deep Networks
- **Part 7. When is Big Data not enough?**

When is Big Data Not Enough? Fundamental limits of current methods?

- Limits 1: there are limits to the functions that Deep Nets can learn and represent.
- Limits 2: there are weakness of Deep Nets to adversarial examples and other attacks.
- Limits 3: there are limits to annotated datasets and big data.

Limits to what Deep Nets can represent.

- Ultimately Deep Nets remain "memorization" methods. They can store and interpolate between examples. But they cannot, as yet, abstract and extrapolate to unseen data. Humans have no difficulty recognizing a blue tree, even if they have never seen one.
- *Deep Nets are very sophisticated regression methods. But, like all regression methods, they can only learn some regression functions (continuous and discrete).*
- Hard to classify exactly what regression functions can be learnt. Intuitively, they are suitable for visual tasks that can solved by storing enough templates. Don't ask the Deep Net to do too much.

Limits to what Deep Nets can represent.

- Deep Nets can learn to remember patterns but cannot understand the causal rules that generate these patterns.

- Astronomy – Deep Nets could learn to provide a description of the planets in solar system (similar to Ptolomy's Epicycles) but they cannot discover the underlying causes of these patterns (Newton's Laws). Hence they cannot generalize knowledge of our solar system to other solar systems.

- For vision: computer graphic models generate images from geometry, material properties (reflectance), and viewpoints. Underlying causes.

Limits to Annotated Datasets

- (1). They bias the research community toward vision problems for which there are high-profile annotated datasets.
- Annotation is easy for some vision tasks – object detection/classification ("is there a cat in this box?") – but hard for others (e.g., depth estimation).
- (2). The image datasets are only partly representative of the complexity of natural images.
- Rare, but important, events may not occur in the datasets – "is there a baby in the road"? Or they will occur very infrequently..
- (3). It is impossible to follow the principles of experimental design and vary the factors in an experiments systematically.
- E.g., detecting a chair as we vary factors like: (i) viewpoint, (ii) lighting, (iii) material properties.

For some vision tasks datasets may never be big enough for current Deep Nets!

- For complex visual problems, the amount of data needed to train and test vision algorithms may become exponentially large as the complexity of the visual task increases.
- An image can be constructed in a combinatorial number of ways: objects, locations, lighting, etc.
- The basic assumptions of Machine Learning will break down. Training and test datasets will not be big enough to represent the space of images.

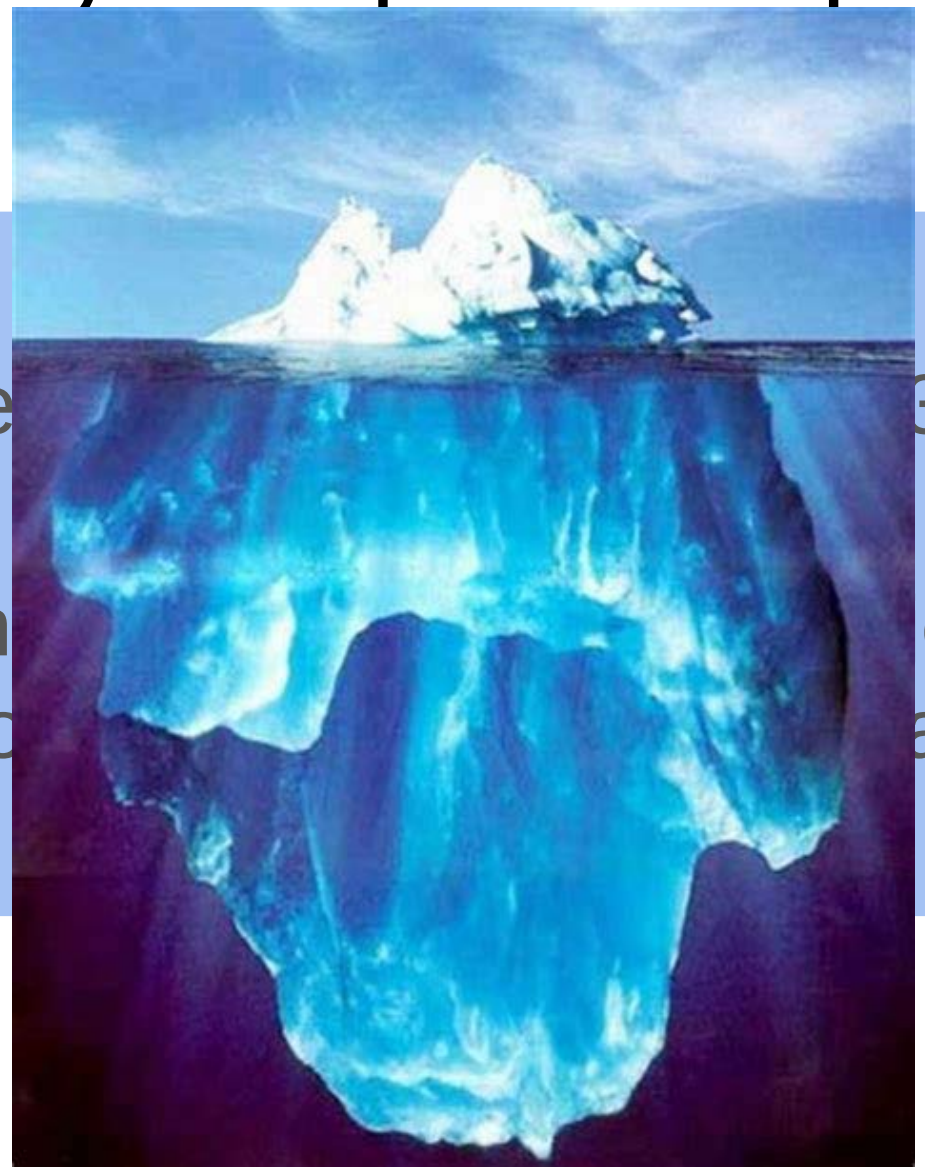
Limits 3: Limits to Annotated Data

Example: synthetize images by Computer Graphics

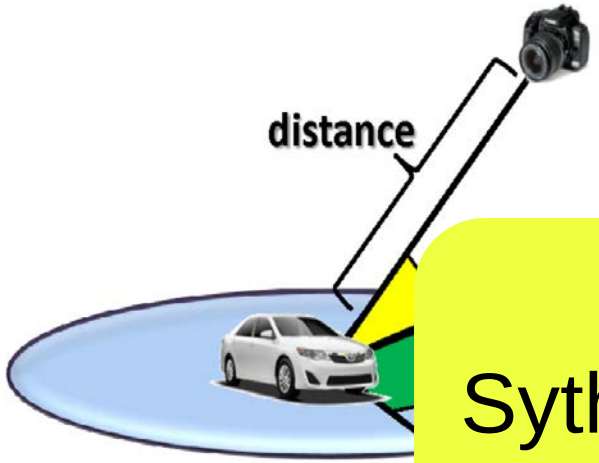
Real Data
(Pascal3D+[2])
~10K images

(Re
(con
mo

3])
ect
at



Images from synthesized computer graphics model.



Sythesized data: INFINITE image space

Camera Pose(4):
azimuth
elevation
tilt(in-plane rotation)
distance

#light source
type(point, dire
omni)
position
color
...

cene Layout(3):
Background
Foreground
Position(Occlusion)

Suppose we simply sample 10^3 possibilities of each parameter listed...

Worse if we allow occlusion and construct scence using multiple objects.

- An object can be occluded in an exponential number of ways.

- An image can be constructed in a combinatorial number of ways: objects, locations, lighting, etc.

Text Captioning: Exponentially Complex?

- Text captioning means giving a text caption to an image (see below).
- Big progress by several groups (including mine), written up in the New York Times.
- But this task is arguably too difficult for complex images..
- There are infinitely many possible images. 20,000 objects, placed in an exponential number of configurations, with occlusion, in 1,400 different types of scenes.

a close up of a bowl of food on a table

a train is traveling down the tracks in a city

a pizza sitting on top of a table next to a box of pizza

- Junhua Mao et al. Deep Captioning… ICLR. 2015.

Experimental Design: UnrealCV //http:unrealcv.org/

- Using Computer Graphics you can systematically vary the stimulus parameters.
- Example 1: Can a Deep Net trained on ImageNet really detect sofas?
  Vary lighting and material properties: (W. Qiu & A. Yuille. ECCV. 2016).

Fig. 4. Images with different camera height and different sofa color.

Elevation \ Azimuth	90	135	180	225	270
0	-	0.713	0.769	0.930	0.319
30	0.900	1.000	0.588	1.000	0.710
60	0.255	0.100	0.148	0.296	0.649

Table 1. The Average Precision (AP) when viewing the sofa from different viewpoints. Observe the AP varies from 0.1 to 1.0 showing the sensitivity to viewpoint. This is perhaps because the biases in the training cause Faster-RCNN to favor specific viewpoints.

- Example 2: Stress-Testing Algorithms: Sensitivity of Stereo algorithms to hazard factors such as specularity, textureless regions, etc. Y. Zhang et al. 3Dvision. 2018.

Virtual Scenes with varying hazardous factors.

Non-Lambertian surfaces

Textureless regions

Transparency

Disparity Jumps

- Stress-test stereo algorithms by varying hazardous factors. 8 levels for each factor.

(a) Specularity

(b) Texturelessness

(c) Transparency

When big data and Deep Nets are not enough

- We need new algorithms that can generalize/extrapolate away from the training examples. Compositional part models, factorize shape and appearance, generative models.
  - E.g., recognize an object from novel viewpoint, with novel material properties, with unknown occluders.

- We must be able to learn from limited size datasets. But be able to test over infinite datasets.
  - Exploit computer graphics data. UnrealCV
- Testing over infinite data is impractical. But we can exploit attacks to explore the data by finding the worst examples.

➔ *Let your worst enemy test your algorithm.*

Guidance from Cognitive Science & Neuroscience

- The human visual system is much more flexible and general purpose than any computer vision algorithm.
  - Humans can generalize to situations they have not seen.
  - They can extrapolate much better than deep networks.
  - Humans have causal understanding of images.
  - And humans use less than 1 watt for computation.

- Cognitive Science aims at understanding human abilities.
  - It can provide challenges to Computer Vision. Computer Vision should mimic human's cognitive abilities.
  - This will probably require computer vision theories which understand the causal structure of the world and the data.

- .

Conclusion

- Big data and Deep Nets are extremely powerful tools for many vision applications. They are, and will be, hugely successful for important real world tasks. They are rapidly developing and evolving.

- Risks that AI will soon take over from humans are science fiction. Current Deep Nets and Big Data will break down in the face of exponential complexity.

- Developing AI systems with human-like capabilities will take time. It will require better understanding of human intelligence. AI, Cognitive Science, and Neuroscience should develop together.