Texture and materials

Subhransu Maji

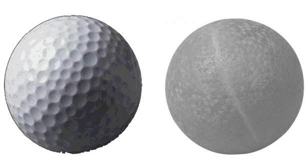
CMPSCI 670: Computer Vision

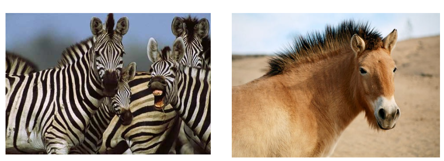
December 1, 2016

What does texture tell us?

- Indicator of materials properties, e.g. brick vs wooden

- Complementary to shape





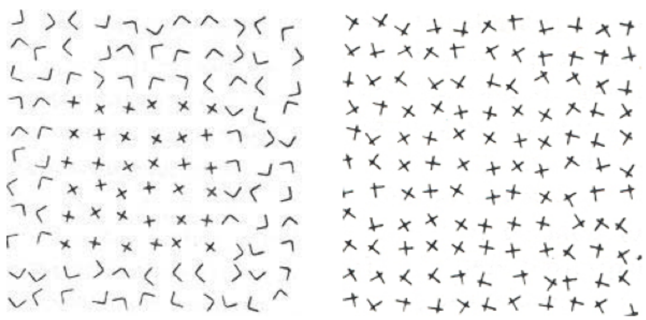
correlated with identity but not the same

Lecture outline

- Texture perception
  - Texture attributes
  - Describing textures from images
- Texture representation
  - Filter-banks and bag-of-words
  - CNN filter-banks for texture

Pre-attentive texture segmentation

- Phenomena in which two regions of texture *quickly* (i.e., in less than 250 ms) and *effortlessly* segregate

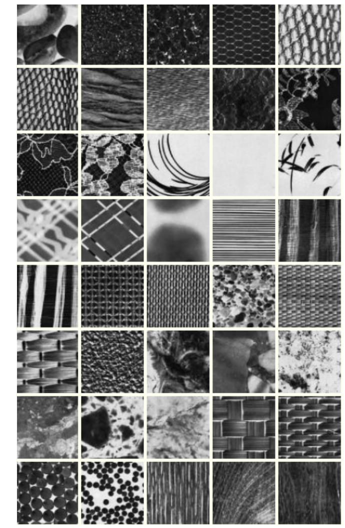


Led to early models of texture representation "textons"

Béla Julesz, Nature, 1981

High-level attributes of texture

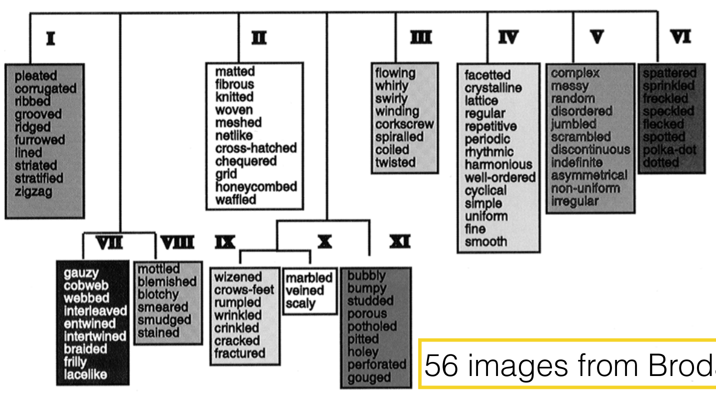
- Early works include:
  - Orientation, contrast, size, spacing, location
  [Bajscy 1973]
  - Coarseness, contrast, directionality, line-like, regularity, roughness
  [Tamura et al., 1978]
  - Coarseness, contrast, busyness, complexity and texture strength
  [Amadusen and King, 1989]
- These attributes can be measured reasonably well from images using low-level statistics of pixel intensities



Brodatz dataset

Towards a texture lexicon

- The texture lexicon: understanding the categorization of visual texture terms and their relationship to texture images, Bhusan, Rao, Lohse, Cognitive Science, 1997



56 images from Brodatz

http://csjarchive.cogsci.rpi.edu/1997v21/i02/p0219p0246/MAIN.PDF

Describable texture dataset

- From human perception to computer vision
- 47 attributes (after accounting for synonyms, etc)
- 120+ images per attribute (crowdsourced)
  https://people.cs.umass.edu/~smaji/papers/textures-cvpr14.pdf

Human centric applications

Properties complementary to materials

Retrieving fabrics and wallpapers

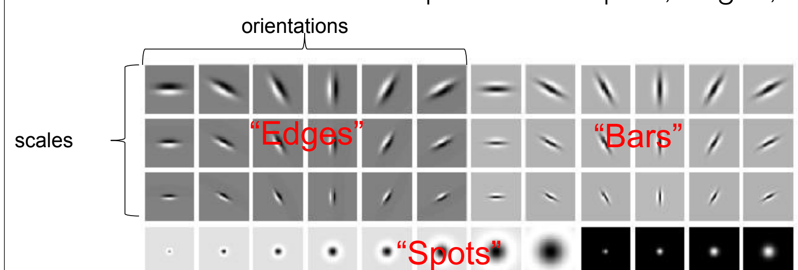
Automatic predictions using computer vision (more later…)

Talk outline

- Texture perception
  - Texture attributes
  - Describing textures in the wild [CVPR 14]
- Texture representation
  - Filter-banks and bag-of-words
  - CNN filter-banks for texture [CVPR 15, IJCV 16]

Texture representation

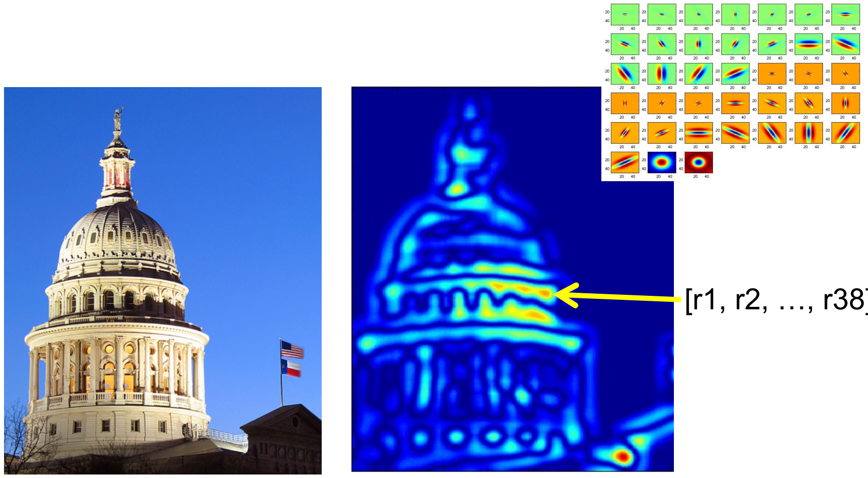
- Textures are made up of repeated local patterns
  - Use filters that look like patterns — spots, edges, bars



Leung & Malik filter bank, IJCV 2001

- Describe their statistics within each image/region

Filter bank response



"Bag of words" for texture

- Absolute positions of local patterns don't matter as much
- Bag of words approach:
  - Inspired by text representation, i.e., document ~ word counts
  - In vision we don't have a pre-defined dictionary
    - Learn words by clustering local responses (Vector quantization)
  - Computational basis of "textons" [Julesz, 1981]

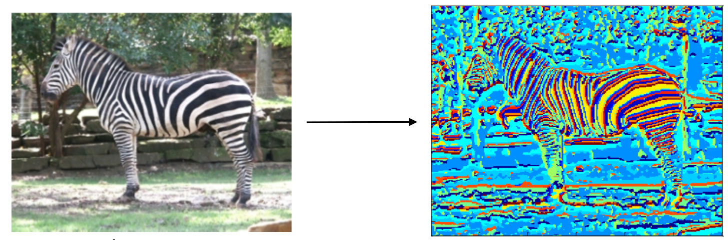


image textons

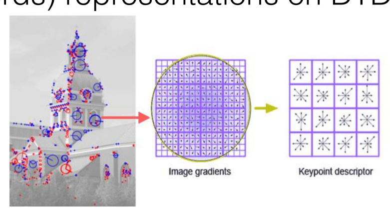
Learning attributes on DTD

Local descr.	Kernel Linear	Hellinger	add-χ^2	exp-χ^2
MR8	15.9±0.8	19.7 ± 0.8	24.1 ± 0.7	30.7 ± 0.7
LM	18.8 ± 0.5	25.8 ± 0.8	31.6 ± 1.1	39.7 ± 1.1
Patch$_{3\times3}$	14.6 ± 0.6	22.3 ± 0.7	26.0 ± 0.8	30.7 ± 0.9
Patch$_{7\times7}$	18.0 ± 0.4	26.8 ± 0.7	31.6 ± 0.8	37.1 ± 1.0
LBP^u	8.2 ± 0.4	9.4 ± 0.4	14.2 ± 0.6	24.8 ± 1.0
LBP-VQ	21.1 ± 0.8	23.1 ± 1.0	28.5 ± 1.0	34.7 ± 1.3
SIFT	**34.7 ± 0.8**	**45.5 ± 0.9**	**49.7 ± 0.8**	**53.8 ± 0.8**

Bag of words (~1k words) representations on DTD dataset

SIFT works quite well

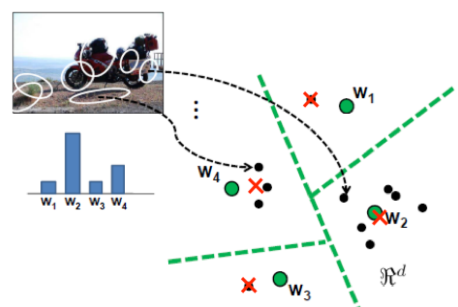
David Lowe, ICCV 99



http://www.codeproject.com/Articles/619039/Bag-of-Features-Descriptor-on-SIFT-Features-with-O

Dealing with quantization error

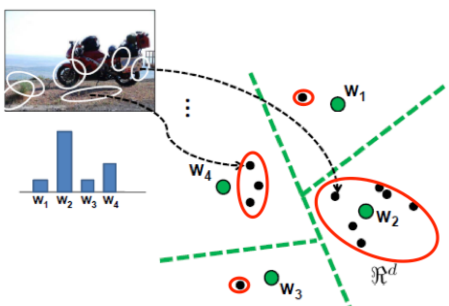
- Bag of words is only **counting** the number of local descriptors assigned to each word (Voronoi cell)
- Why not include other statistics? For instance:
  - Mean of local descriptors **x**



http://www.cs.utexas.edu/~grauman/courses/fall2009/papers/bag_of_visual_words.pdf

Dealing with quantization error

- Bag of words is only **counting** the number of local descriptors assigned to each word (Voronoi cell)
- Why not include other statistics? For instance:
  - Mean of local descriptors x
  - Covariance of local descriptors



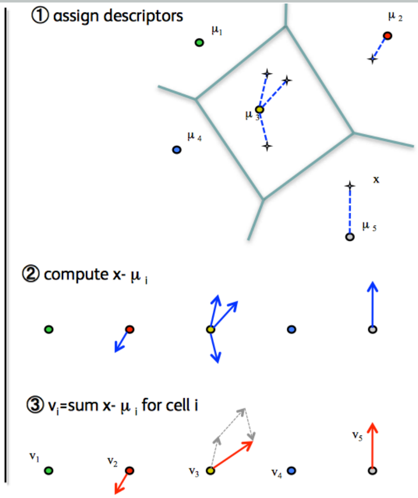
http://www.cs.utexas.edu/~grauman/courses/fall2009/papers/bag_of_visual_words.pdf

The VLAD descriptor

Given a codebook $\{\mu_i, i = 1 \dots N\}$, e.g. learned with K-means, and a set of local descriptors $X = \{x_t, t = 1 \dots T\}$:

- ① assign: $\mathrm{NN}(x_t) = \arg\min_{\mu_i} ||x_t - \mu_i||$
- ②③ compute: $v_i = \sum_{x_t : \mathrm{NN}(x_t) = \mu_i} x_t - \mu_i$
- concatenate v_i's + ℓ_2 normalize

Very high dimensional: NxD

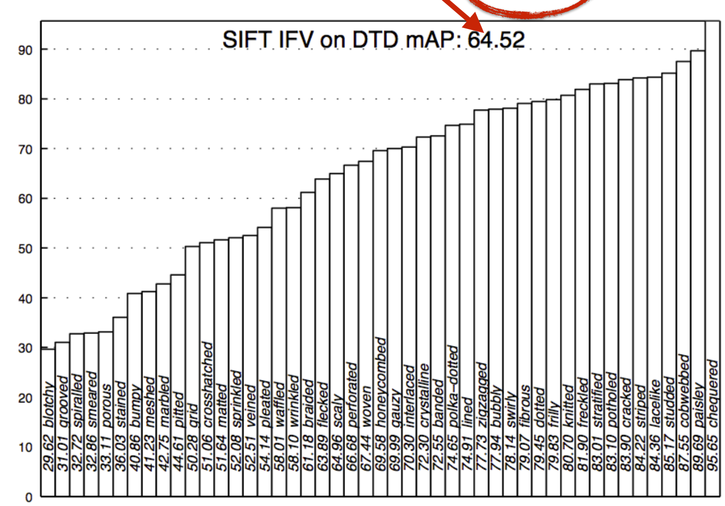


Jégou, Douze, Schmid and Pérez, "Aggregating local descriptors into a compact image representation", CVPR'10.

Fisher-vectors use both mean and covariance [Perronnin et al, ECCV 10]

Fisher-vectors with SIFT

SIFT BoVW + linear SVM: mAP = **37.4** **+27%**



Describable attributes as features

- Train classifiers to predict 47 attributes
  - SIFT + AlexNet features to make predictions
  - On a new dataset, learn classifiers on 47 features

Features	KTH-2b	FMD	
DTD	73.8%	61.1%	47 dim
Prev best	57.1%	66.3%	
DTD + SIFT + DeCAF	**77.1%**	**67.1%**	66K dim

- DTD attributes correlate well with material properties

The quest for better features …

- Early filter banks were based on simple linear filters - is there something better? Can we learn them from data?
- Slow progress for a while and performance plateaued on a number of benchmarks, e.g. PASCAL VOC

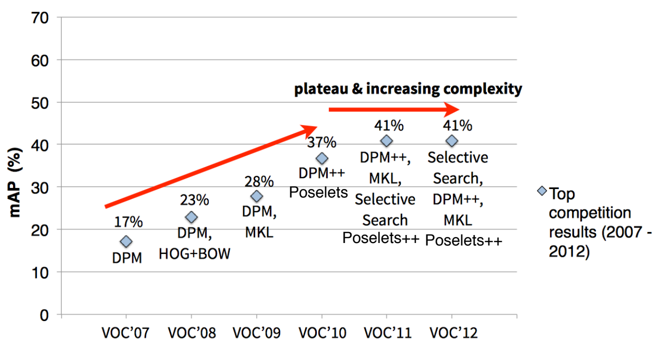
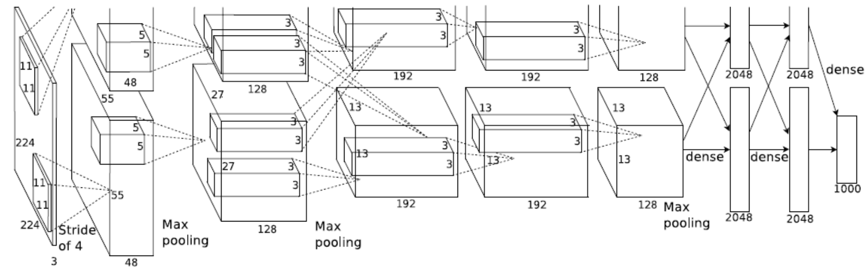


Figure by Ross Girshick

[Source: http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc20{07,08,09,10,11,12}/results/index.html]

ImageNet classification breakthrough



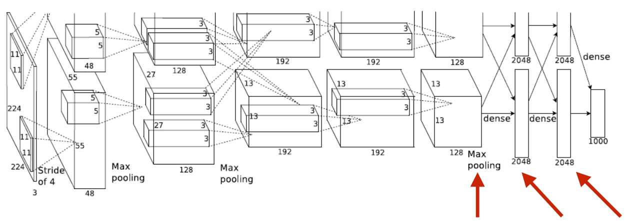
"AlexNet" CNN
60 million parameters trained on 1.2 million images

Krizhevsky, Strutsvekar, Hinton, NIPS 2012

ILSVRC 2012 test	Top-5 error
Fisher Vectors (ISI)	26.2%
5 SuperVision CNNs	16.4%
7 SuperVision CNNs	**15.3%**

+1 for crowdsourcing

CNNs as feature extractors



- Take the outputs of various layers *conv5, fc6, fc7*
- State of the art on many datasets (Donahue et al, ICML 14)
- Regions with CNN features (Girshick et al., CVPR 14) achieves **41%⇨53.7%** on PASCAL VOC 2007 detection challenge. Current best results **66%**!
- A flurry of activity in computer vision; benchmarks are being shattered every few months! Great time for vision applications

CNNs for texture

Dataset	FV (SIFT)	AlexNet
CUReT	**99.5**	97.9
UMD	**99.2**	96.4
UIUC	**97.0**	94.2
KT	**99.7**	96.9
KT-2a	**82.2**	78.9
KT-2b	69.3	**70.7**
FMD	58.2	**60.7**
DTD	**61.2**	54.8
mean	***83.3***	*81.3*

Texture recognition accuracy

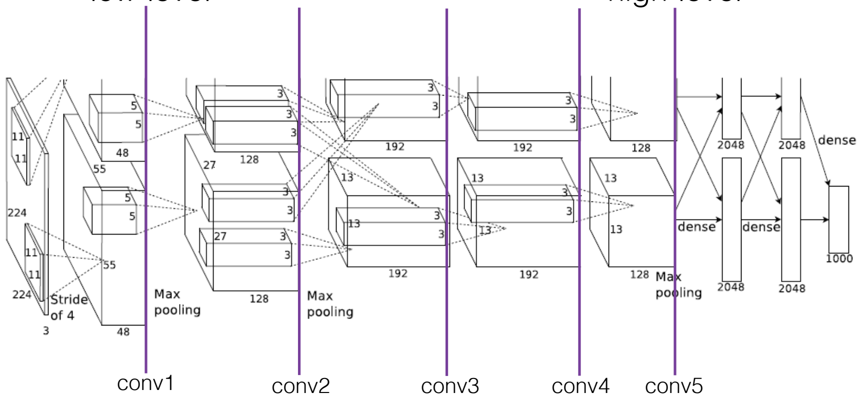
Flickr material dataset (10 categories)

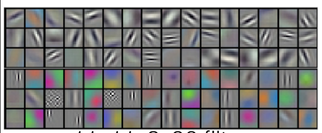
http://people.csail.mit.edu/celiu/CVPR2010/FMD/

- CNN features from the last layer don't seem to outperform SIFT on texture datasets
- Speculations on why?
  - Textures are different from categories on ImageNet which are mostly objects
  - Dense layers preserve spatial structure are not ideal for measuring orderless statistics

CNN layers are non-linear filter banks

low-level → high-level





11x11x3x96 filters

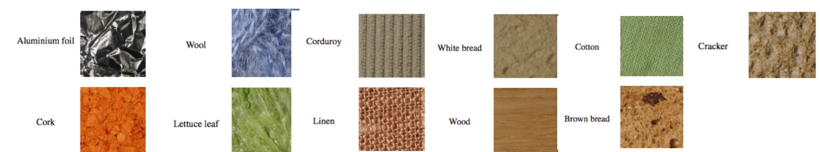
Obtain filter banks by truncating the CNN

CNNs for texture

Texture recognition accuracy

Dataset	FV (SIFT)	AlexNet
KT-2b	69.3	70.7
FMD	58.2	60.7
DTD	61.2	54.8

KT-2b dataset (11 material categories)



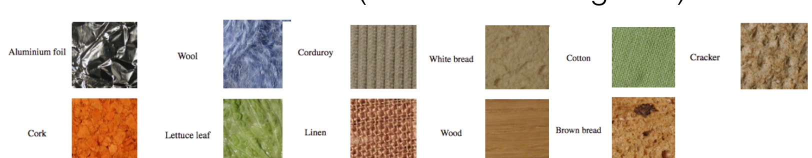
CNNs for texture

Texture recognition accuracy

Dataset	FV (SIFT)	AlexNet (FC)	FV (conv5)
KT-2b	69.3	70.7	**71.0**
FMD	58.2	60.7	**72.6**
DTD	61.2	54.8	**66.7**

Significant improvements over simply using CNN features

KT-2b dataset (11 material categories)



CNNs for texture

Texture recognition accuracy

Dataset	FV (SIFT)	AlexNet (FC)	FV (conv5)	FV (conv13)
KT-2b	69.3	70.7	**71.0**	**72.2**
FMD	58.2	60.7	**72.6**	**80.8**
DTD	61.2	54.8	**66.7**	**80.5**

Using the model from Oxford VGG group that performed the best on LSVRC 2014 (ImageNet classification challenge)
http://www.robots.ox.ac.uk/~vgg/research/very_deep/

Scenes and objects as textures

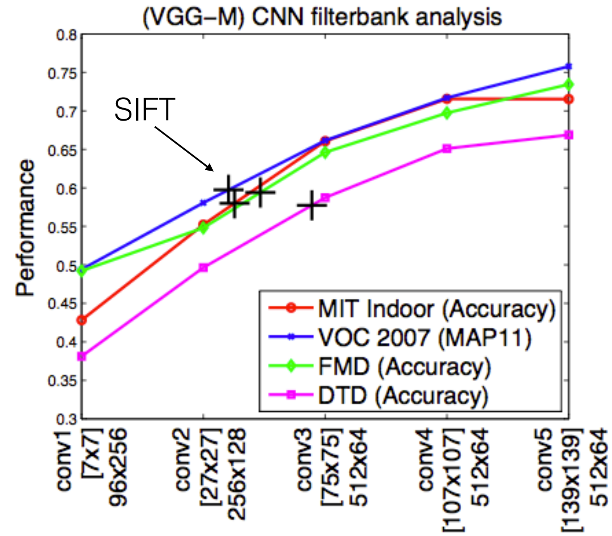
- MIT Indoor dataset (67 classes)

Prev. best: **70.8**%   D-CNN **81.7**%
Zhou et al., NIPS 14

- CUB 200 dataset (bird sub-category recognition)

Prev. best: **76.4**%(w/ parts)   FV-CNN **72.1**% (w/o parts)
Zhang et al., ECCV 14

SIFT vs. CNN filter banks



OpenSurfaces material segmentation

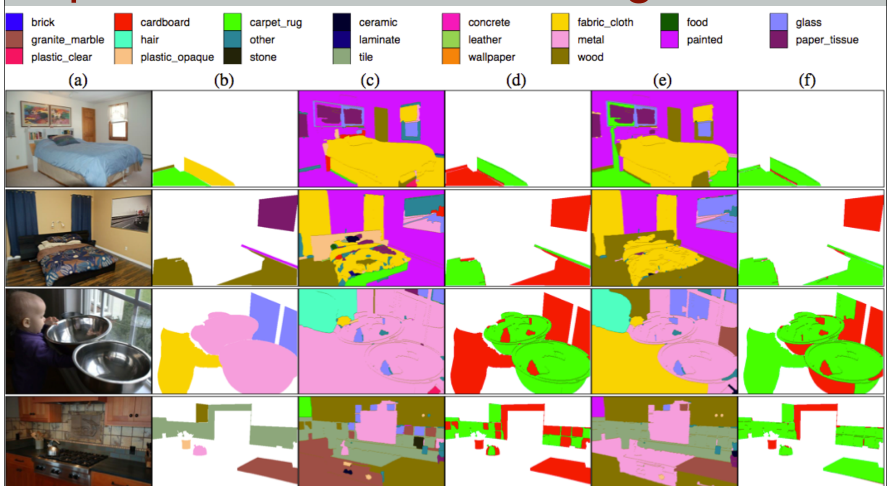


image   gt   R-CNN   errors   FV-CNN   errors

MSRC segmentation dataset

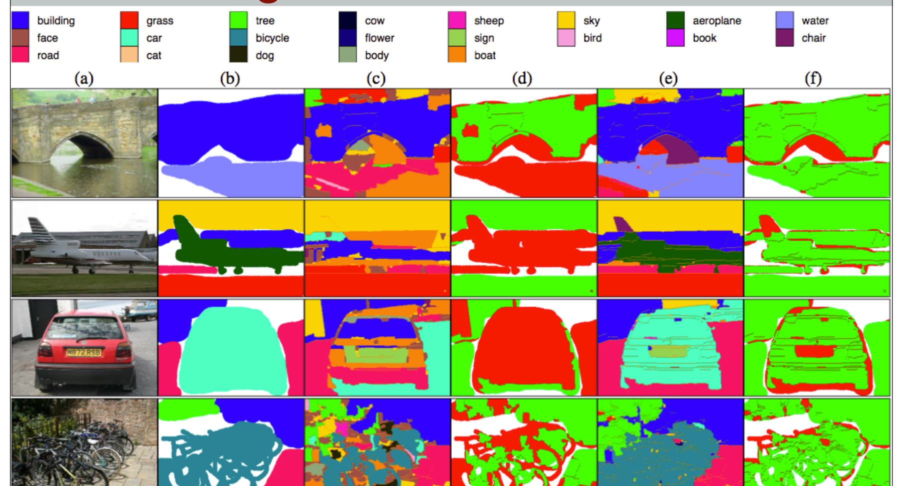


image   gt   R-CNN   errors   D-CNN   errors

FV-CNN **87.0**% vs **86.5**% [Ladicy et al., ECCV 2010]