Architectures for Invariant Image Recognition

Yann LeCun (Courant Institute, NYU)
Fu Jie Huang (Courant Institute, NYU)
Leon Bottou (NEC Labs)

http://yann.lecun.com

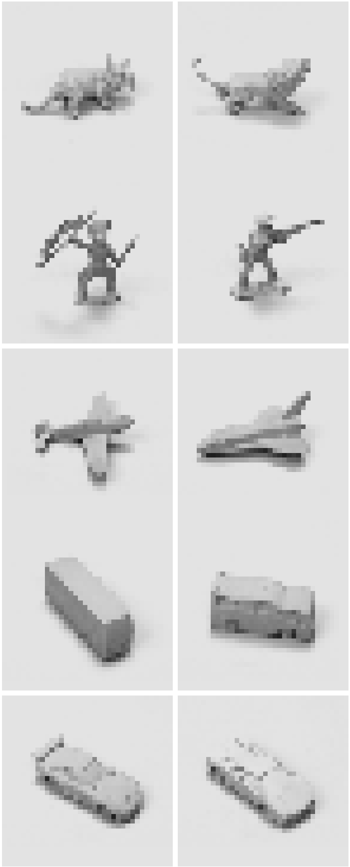
http://www.cs.nyu.edu/~yann

Invariance

- **The appearance of an object (in terms of pixels) changes considerably under changes of pose, illumination, clutter, and occlusions.**
- **Two instance of the same category may have widely differing shapes and appearances**
  - An airliner and a fighter plane, a person standing and another one kneeling,...
- **Template-based methods are doomed because the number of templates necessary to cover the space of variations grows exponentially with the number of dimensions of the variations.**

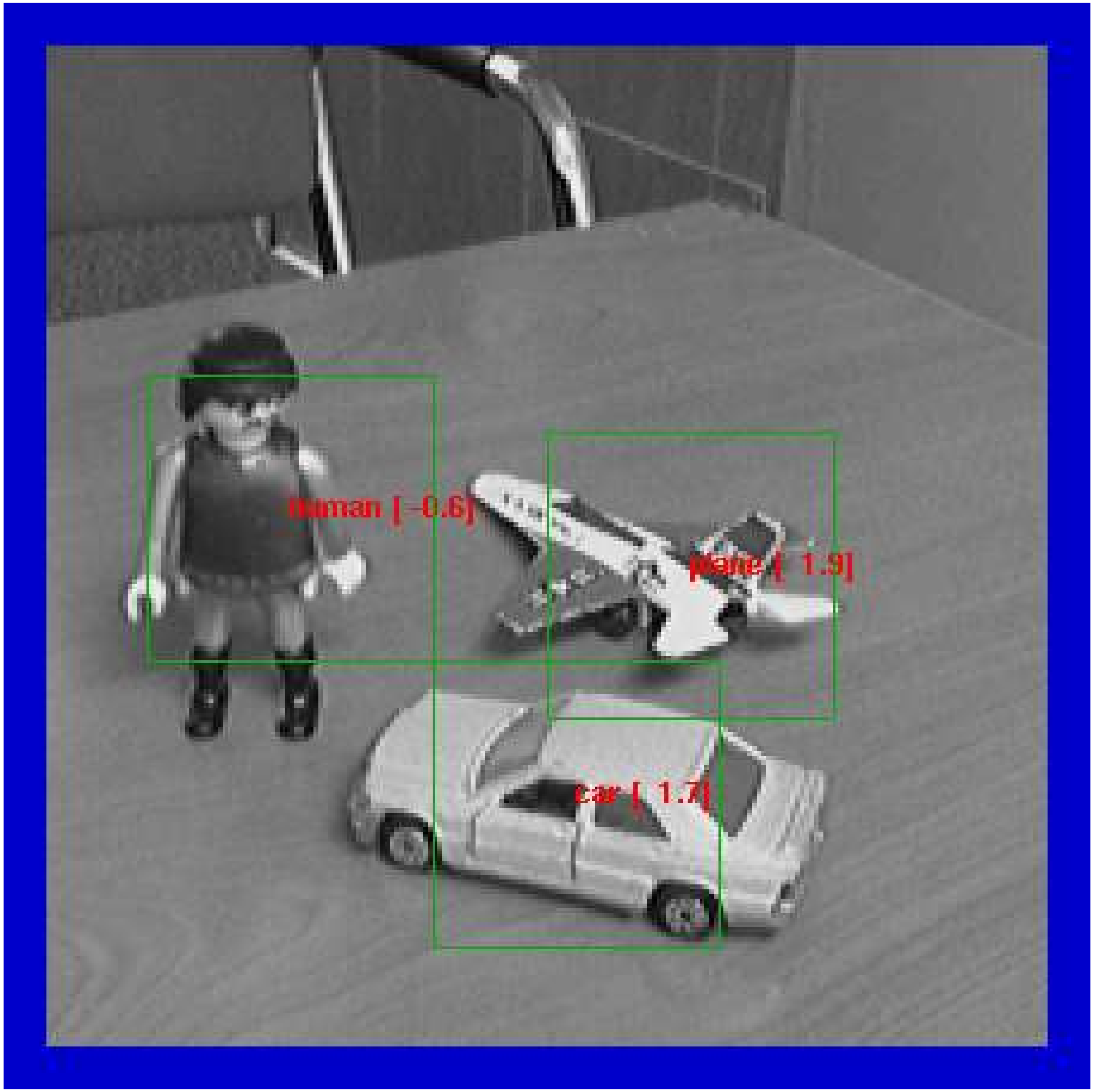
Generic Object Recognition

- **Generic Object Recognition** is the problem of detecting and classifying objects into generic categories such as "cars", "trucks", "airplanes", "animals", or "human figures"
- **Appearances are highly variable within a category** because of shape variation, position in the visual field, scale, viewpoint, illumination, albedo, texture, background clutter, and occlusions.
- **Learning invariant representations is key.**
- **Understanding the neural mechanism behind invariant recognition is one of the main goals of Visual Neuroscience.**

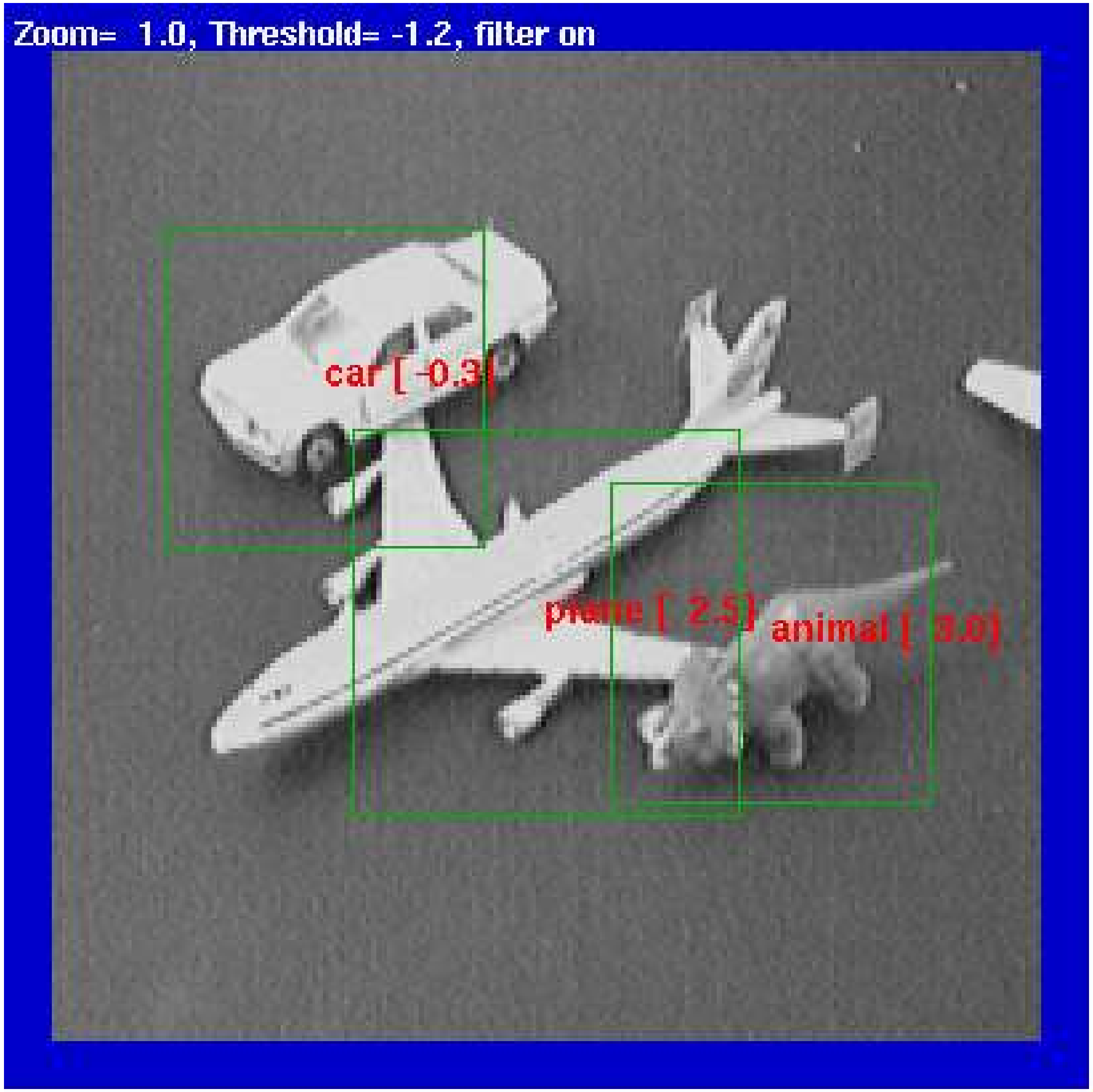


What we want to achieve

- **recognition from shape:** color, texture, and distinctive local features may be useful, but they merely allow us to sweep the real problems under the rug.
- **Full invariance to viewpoint, illumination, clutter, occlusions.**



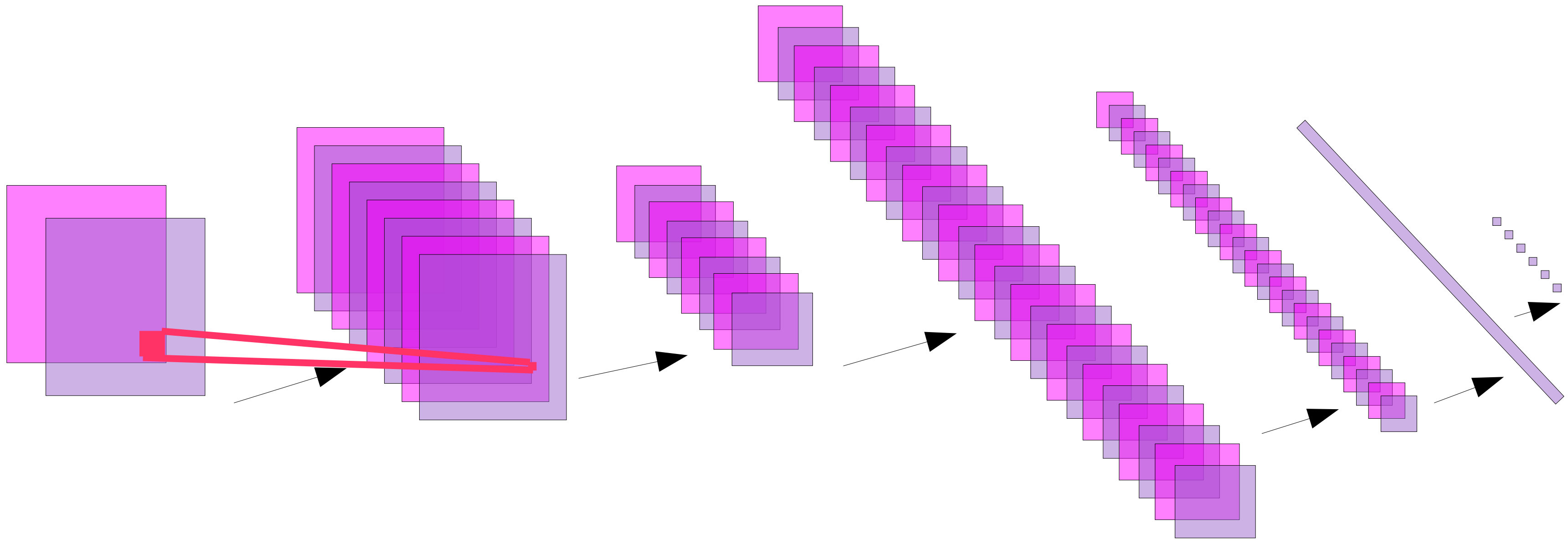
Occlusions



Clutter

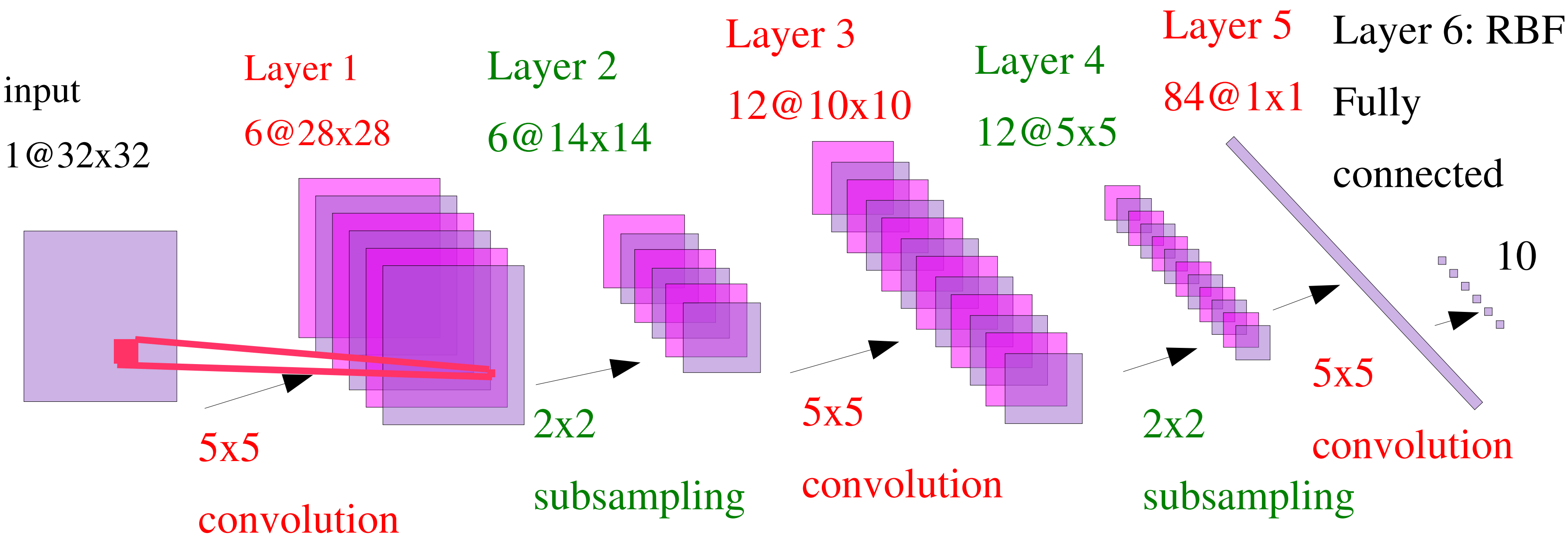
Thrs= 0.5, f on , os=40, nwin=23616
car [1.7]
Thrs= 0.5, f on , os=40, nwin=23616
human [3.1]
Thrs= 0.5, f on , os=40, nwin=23616
animal [1.9]
Thrs= 0.5, f on , os=40, nwin=23616
truck [2.3]
Thrs= 0.5, f on , os=40, nwin=23616
human [3.6]
Thrs= 0.5, f on , os=40, nwin=23616
animal [1.2]

Convolutional Network



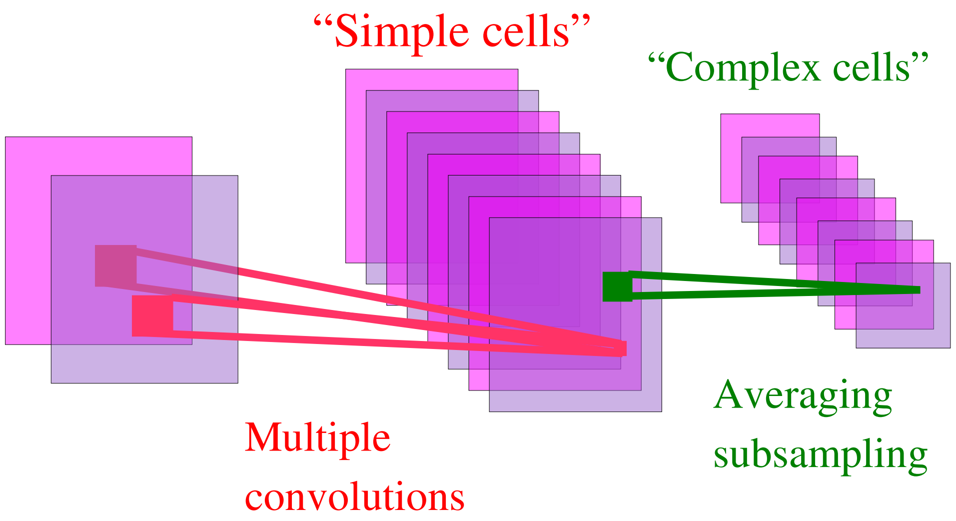
- **Hierarchical/multilayer:** features get progressively more global, invariant, and numerous
- **dense features:** features detectors applied everywhere (no interest point)
- **broadly tuned (possibly invariant) features:** sigmoid units are on half the time.
- **Global discriminative training:** The whole system is trained “end-to-end” with a gradient-based method to minimize a global loss function
- **Integrates segmentation, feature extraction, and invariant classification in one fell swoop.**

Handwritten Digit Recognition with a Convolutional Network

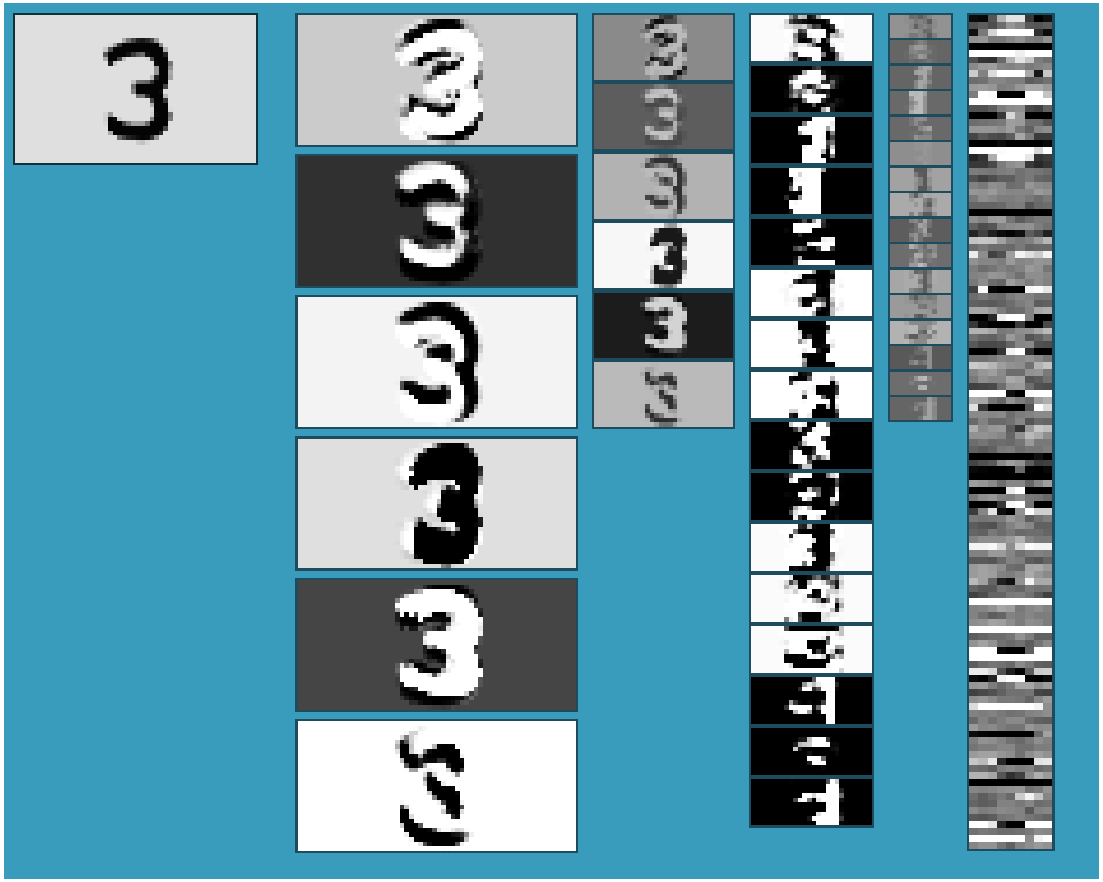


- **60,000 free parameters, 400,000 connections.**
- The architecture alternates convolutional layers (feature detectors) and subsampling layers (local feature pooling for invariance to small distortions).
- Handwritten Digit Dataset MNIST: 60,000 training samples, 10,000 test samples
- **The entire network is trained end-to-end** (all the layers are trained simultaneously).
- Test Error Rate: 0.8%

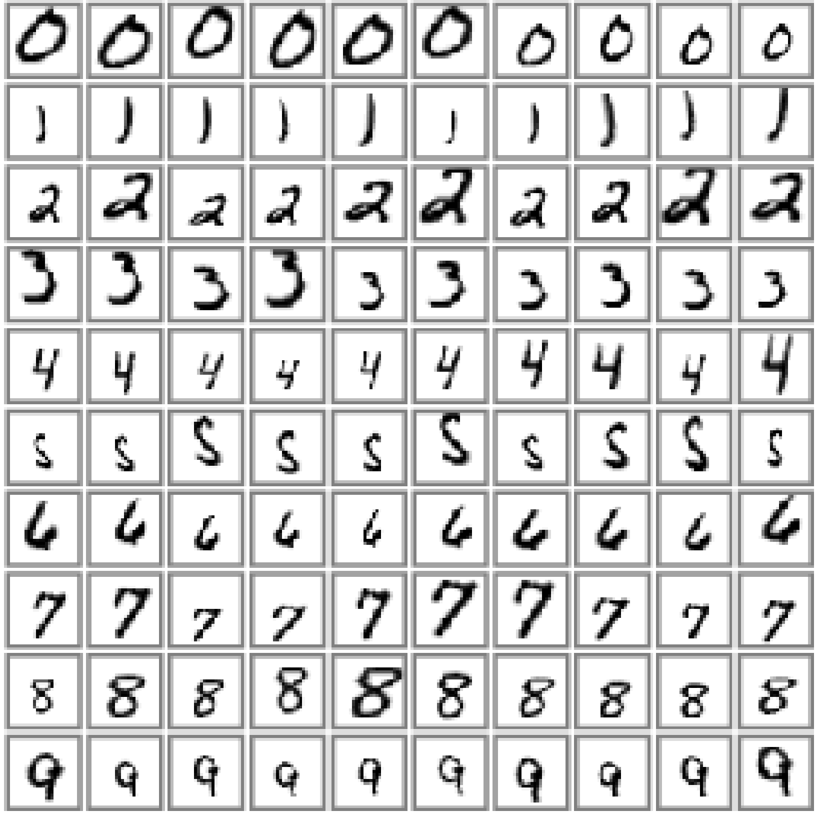
Alternated Convolutions and Subsampling



- Local features are extracted everywhere.
- averaging/subsampling layer builds robustness to variations in feature locations.
- Hubel/Wiesel'62, Fukushima'71, LeCun'89, Riesenhuber & Poggio'02, Ullman'02,....



MNIST Dataset



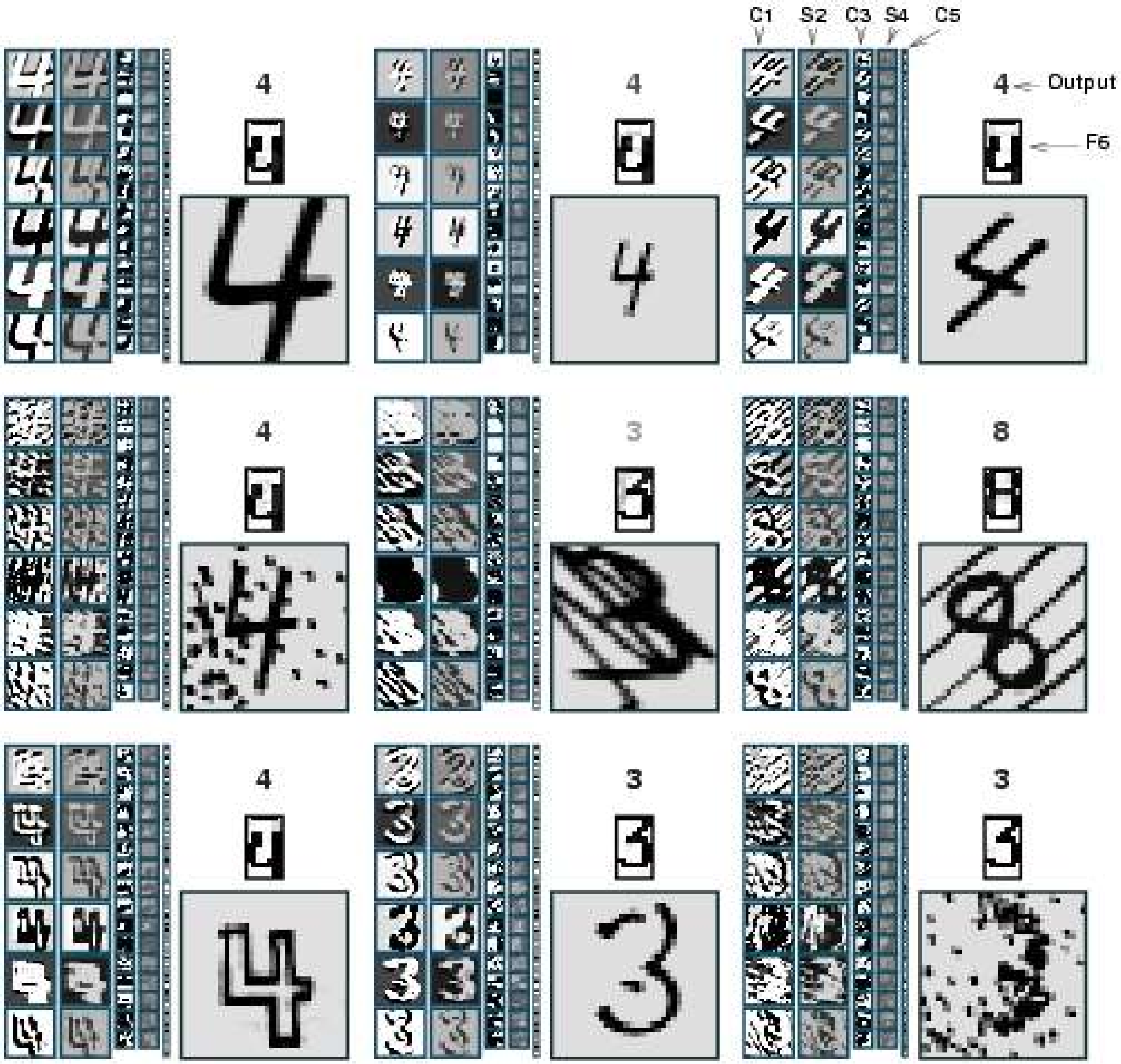
- Handwritten Digit Dataset MNIST: 60,000 training samples, 10,000 test samples

Results on MNIST Handwritten Digits

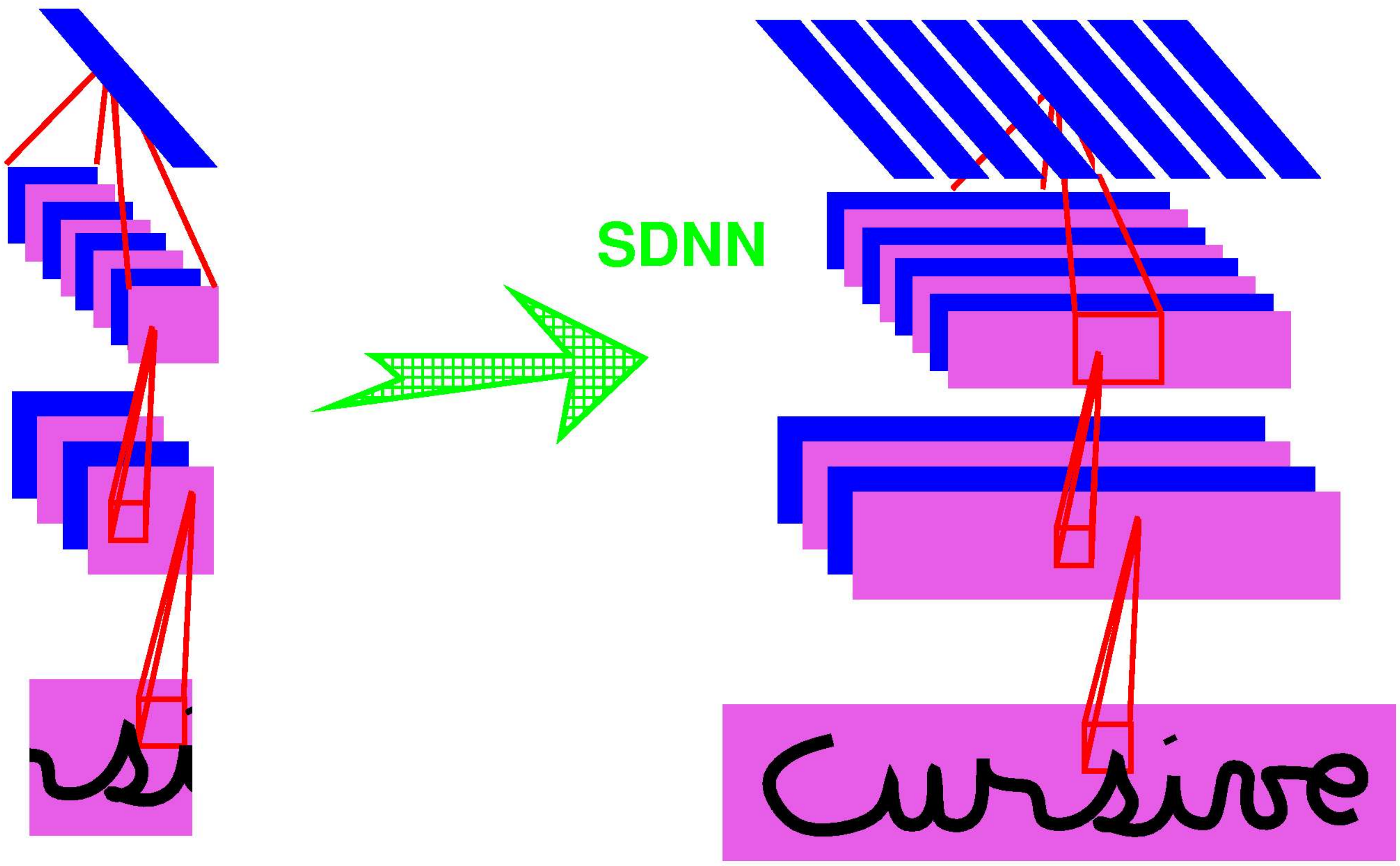
CLASSIFIER	DEFORMATION	PREPROCESSING	ERROR (%)	Reference
linear classifier (1-layer NN)		none	12.00	LeCun et al. 1998
linear classifier (1-layer NN)		deskewing	8.40	LeCun et al. 1998
pairwise linear classifier		deskewing	7.60	LeCun et al. 1998
K-nearest-neighbors, (L2)		none	3.09	Kenneth Wilder, U. Chicago
K-nearest-neighbors, (L2)		deskewing	2.40	LeCun et al. 1998
K-nearest-neighbors, (L2)		deskew, clean, blur	1.80	Kenneth Wilder, U. Chicago
K-NN L3, 2 pixel jitter		deskew, clean, blur	1.22	Kenneth Wilder, U. Chicago
K-NN, shape context matching		shape context feature	0.63	Belongie et al. IEEE PAMI 2002
40 PCA + quadratic classifier		none	3.30	LeCun et al. 1998
1000 RBF + linear classifier		none	3.60	LeCun et al. 1998
K-NN, Tangent Distance		subsamp 16x16 pixels	1.10	LeCun et al. 1998
SVM, Gaussian Kernel		none	1.40	
SVM deg 4 polynomial		deskewing	1.10	LeCun et al. 1998
Reduced Set SVM deg 5 poly		deskewing	1.00	LeCun et al. 1998
Virtual SVM deg-9 poly	Affine	none	0.80	LeCun et al. 1998
V-SVM, 2-pixel jittered		none	0.68	DeCoste and Scholkopf, MLJ 2002
V-SVM, 2-pixel jittered		deskewing	0.56	DeCoste and Scholkopf, MLJ 2002
2-layer NN, 300 HU, MSE		none	4.70	LeCun et al. 1998
2-layer NN, 300 HU, MSE,	Affine	none	3.60	LeCun et al. 1998
2-layer NN, 300 HU		deskewing	1.60	LeCun et al. 1998
3-layer NN, 500+150 HU		none	2.95	LeCun et al. 1998
3-layer NN, 500+150 HU	Affine	none	2.45	LeCun et al. 1998
3-layer NN, 500+300 HU, CE, reg		none	1.53	Hinton, unpublished, 2005
2-layer NN, 800 HU, CE		none	1.60	Simard et al., ICDAR 2003
2-layer NN, 800 HU, CE	Affine	none	1.10	Simard et al., ICDAR 2003
2-layer NN, 800 HU, MSE	Elastic	none	0.90	Simard et al., ICDAR 2003
2-layer NN, 800 HU, CE	Elastic	none	0.70	Simard et al., ICDAR 2003
Convolutional net LeNet-1		subsamp 16x16 pixels	1.70	LeCun et al. 1998
Convolutional net LeNet-4		none	1.10	LeCun et al. 1998
Convolutional net LeNet-5,		none	0.95	LeCun et al. 1998
Conv. net LeNet-5,	Affine	none	0.80	LeCun et al. 1998
Boosted LeNet-4	Affine	none	0.70	LeCun et al. 1998
Conv. net, CE	Affine	none	0.60	Simard et al., ICDAR 2003
Comv net, CE	Elastic	none	0.40	Simard et al., ICDAR 2003

LeNet5 errors on the MNIST test set

Invariance and Robustness to Noise

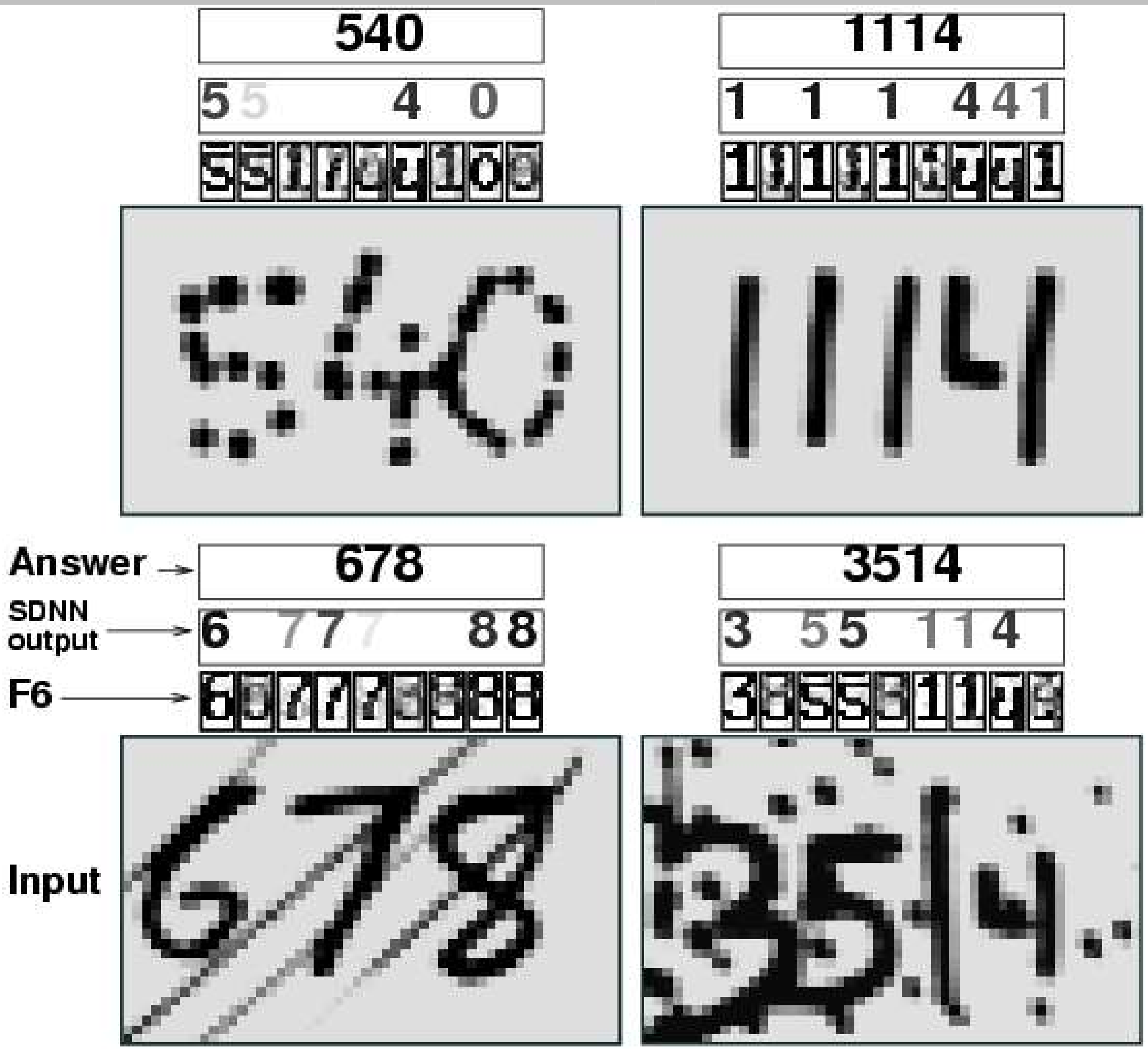


Recognizing Multiple Characters with Replicated Nets



Recognizing Multiple Characters with Replicated Nets

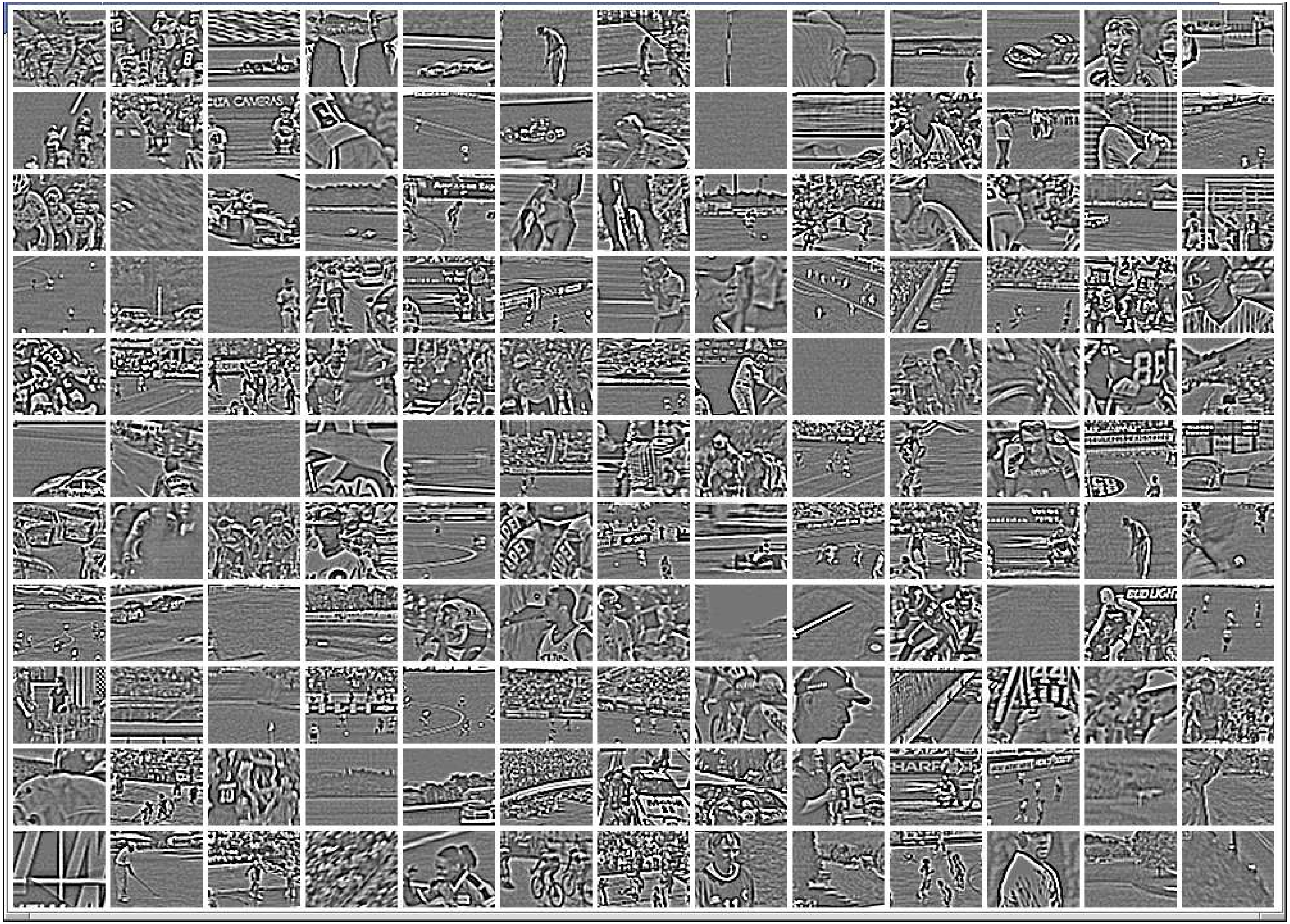
Handwriting Recognition



TV sport categorization (with Alex Niculescu, Cornell)

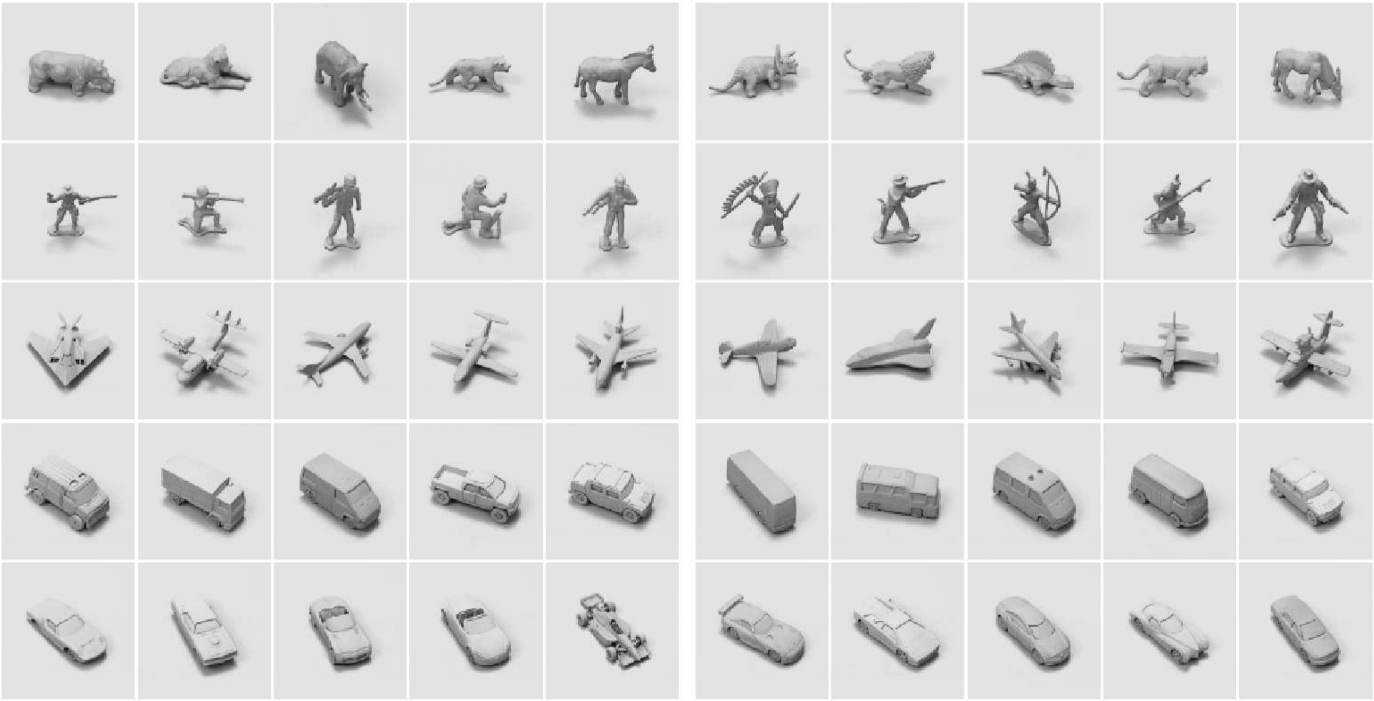
- **Classifying TV sports snapshots into 7 categories: auto racing, baseball, basketball, bicycle, golf, soccer, football.**
- **123,900 training images (300 sequence with 59 frames for each sport)**
- **82,600 test images (200 sequences with 59 frames for each sport)**
- **Preprocessing: convert to YUV, high-pass filter the Y component, crop, subsample to 72x60 pixels**
- **Results:**
  - frame-level accuracy: 61% correct
  - Sequence-level accuracy 68% correct (simple voting scheme).

TV sport categorization (with Alex Niculescu, Cornell)



The NYU Object Recognition Benchmark (NORB Dataset)

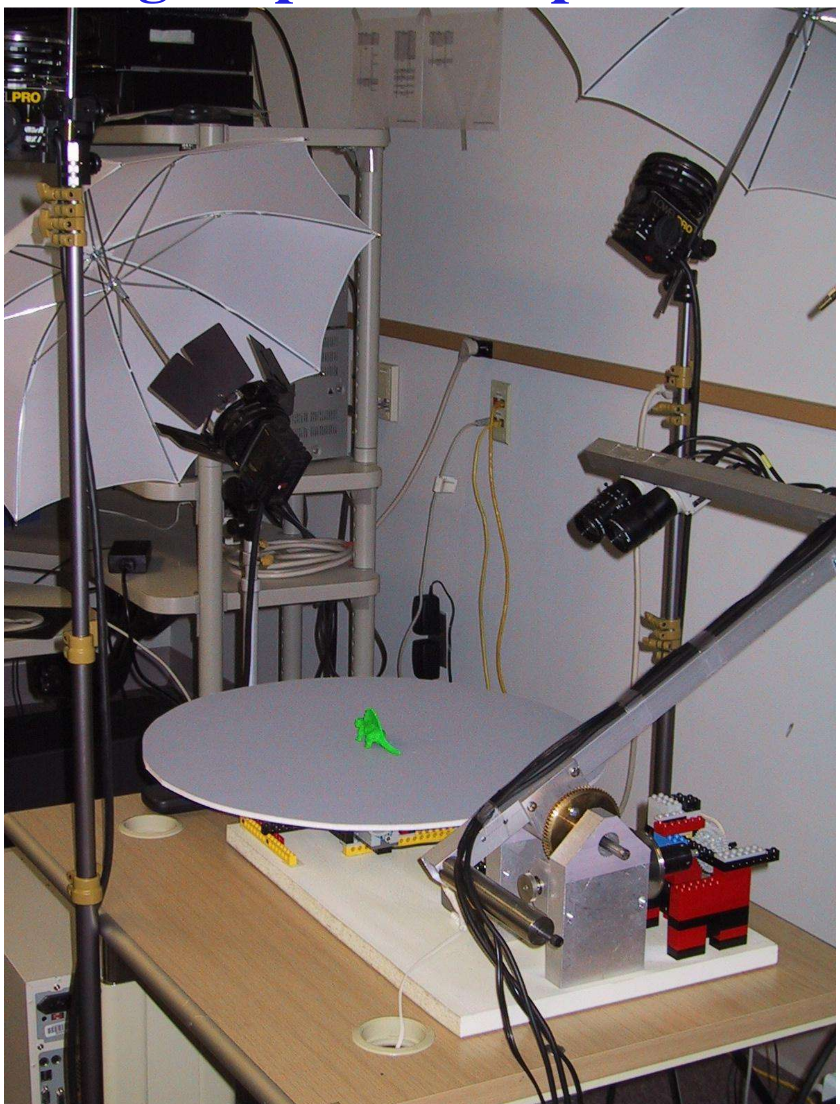
- **50** toys belonging to 5 categories: **animal, human figure, airplane, truck, car**
- **10** instance per category: 5 instances used for training, 5 instances for testing
- **Raw dataset: 972** stereo pair of each object instance. **48,600** image pairs total.
- **For each instance:**
- **18 azimuths**
  - 0 to 350 degrees every 20 degrees
- **9 elevations**
  - 30 to 70 degrees from horizontal every 5 degrees
- **6 illuminations**
  - on/off combinations of 4 lights
- **2 cameras (stereo)**
  - 7.5 cm apart
  - 40 cm from the object



Training instances **Test instances**

Data Collection, Sample Generation

Image capture setup



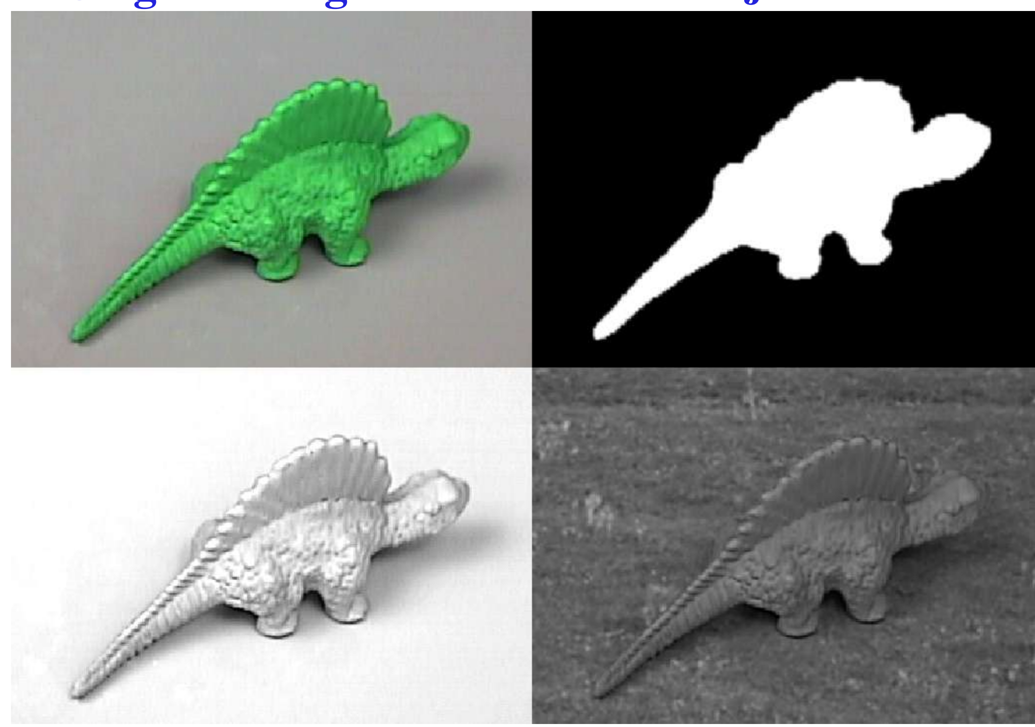
Objects are painted green so that:

- all features other than shape are removed

- objects can be segmented, transformed,

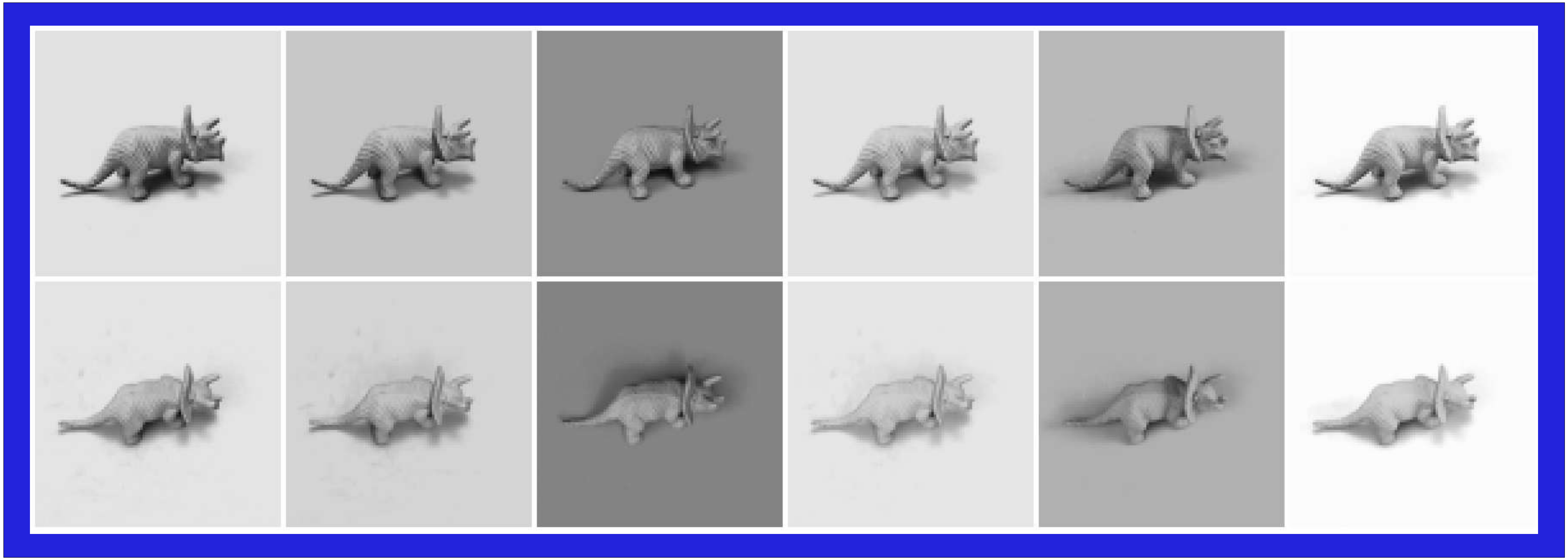
  and composited onto various backgrounds

Original image **Object mask**



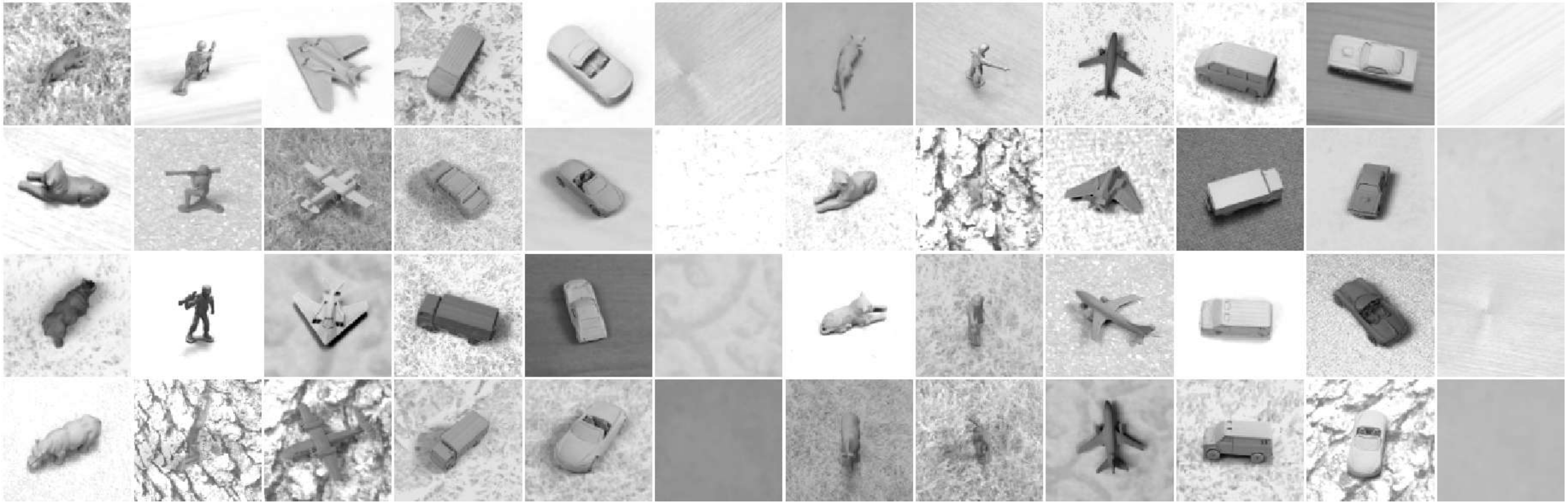
Shadow factor **Composite image**

Data Collection, Sample Generation



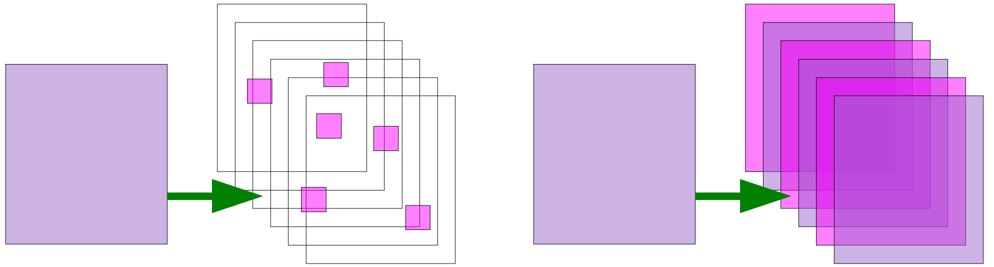
Samples showing the 6 different illuminations for 2 different elevations

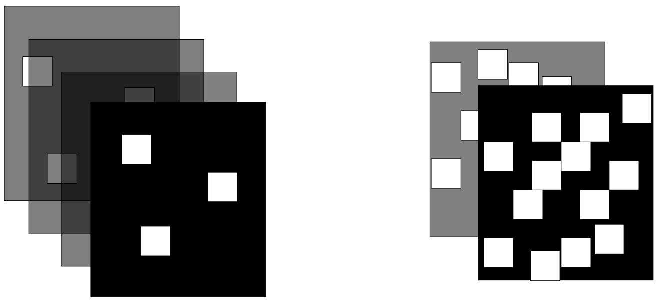
Textured and Cluttered Datasets

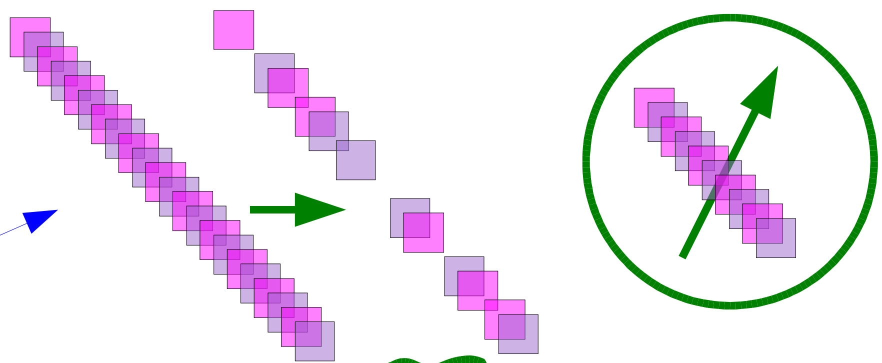


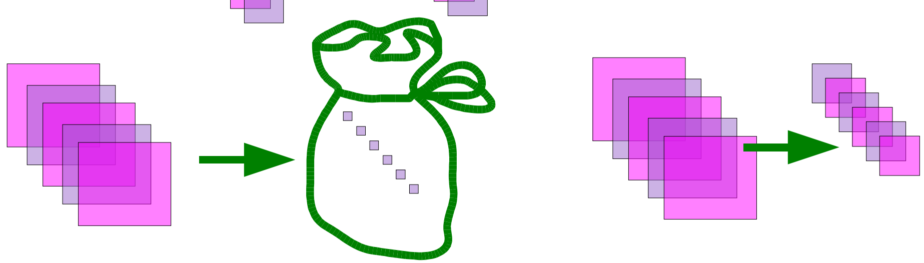
Computational Models of Object Recognition

- Detecting features at interest points (Schmid, Perona, Ponce, Lowe) versus detecting them everywhere (LeCun, Ullman).
- Fixed features (Gabor, SIFT, Shape Context...), versus learned features
- Many sparse/selective features (Ullman's fragments) versus few dense/broad features (features that are “on” half the time).
- Selection from lots of simple features (Viola/Jones), vs tuning/optimization of a small number of features.
- Bag of features vs spatial relationships



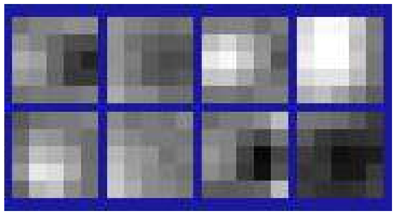


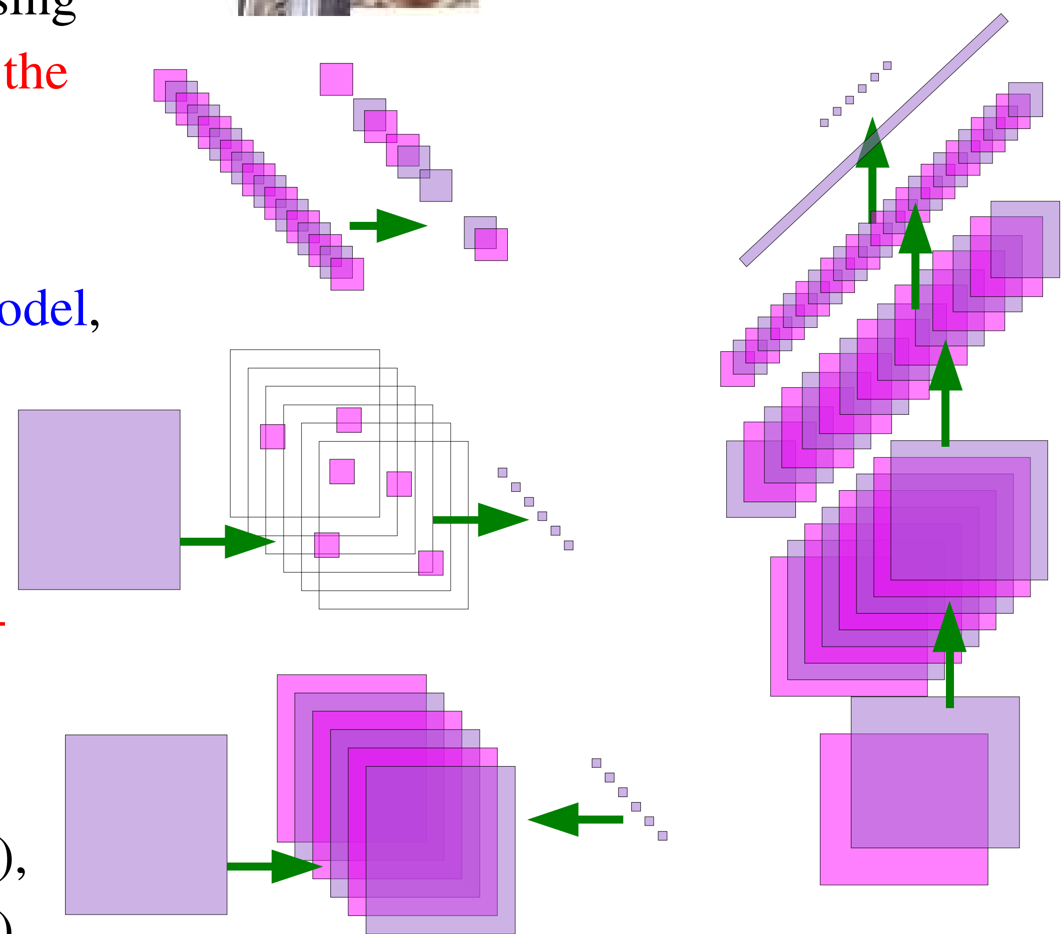




What Architecture, what training?

- Selection of “patch” features (Schmid, Ullman, Ponce, Perona,.....), versus optimization of non-template features.
- “heuristic” feature selection (e.g. Using mutual information) versus learning the features by optimizing a global performance measure.
- Piecemeal training of feature and model, versus global training of the whole system
- 2-layer feature+model (almost everyone), versus hierarchical/multi-level (LeCun, Riesenhuber, Geman, Ullman)
- Generative (Perona, Amit, Freeman), versus discriminative (LeCun, Viola)

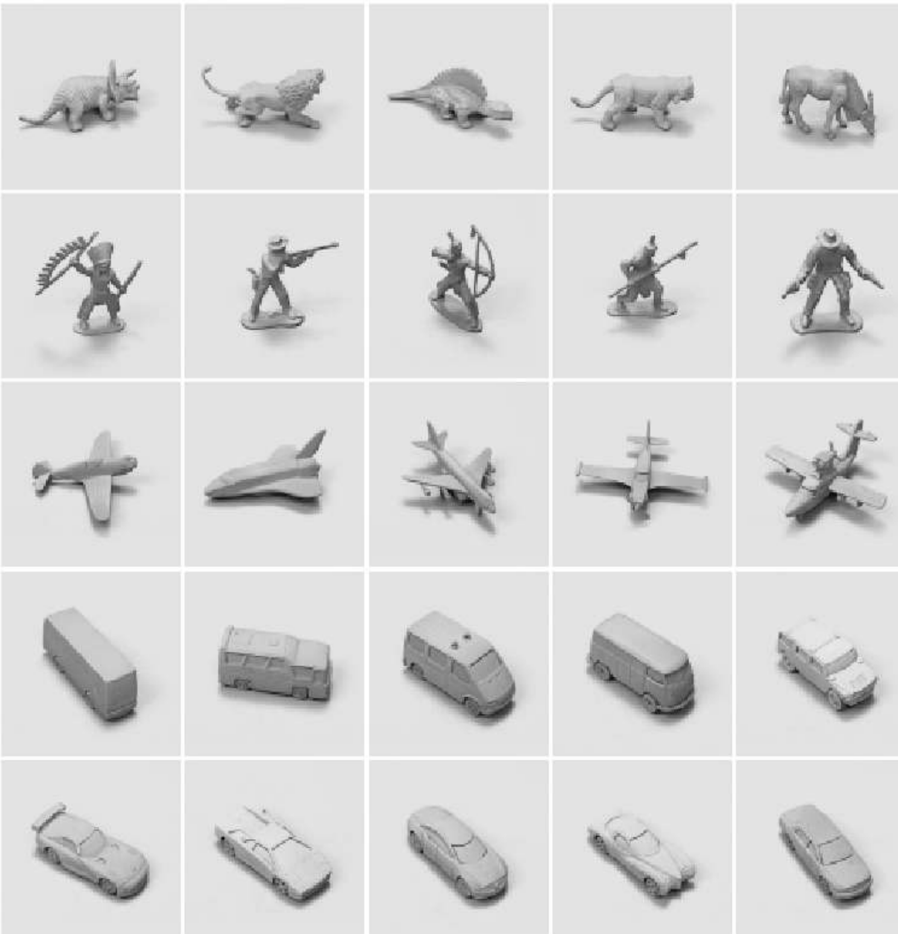




Experiment 1: Normalized-Uniform Dataset

- **Normalized-Uniform Dataset:** **972 stereo pair of each object instance** (18 azimuths X 9 elevations X 6 illuminations).
- **5 categories. 5 instances/category for training, 5 instances/category for testing**
- **24,300** stereo pairs for training, **24,300** for testing
- Objects are centered and size-normalized so all the views of each object instance fits in an 80x80 pixel window.
- Objects are placed on uniform backgrounds (one for each of the 6 illuminations) of size 96x96 pixels
- Each sample is composed of two 96x96 images

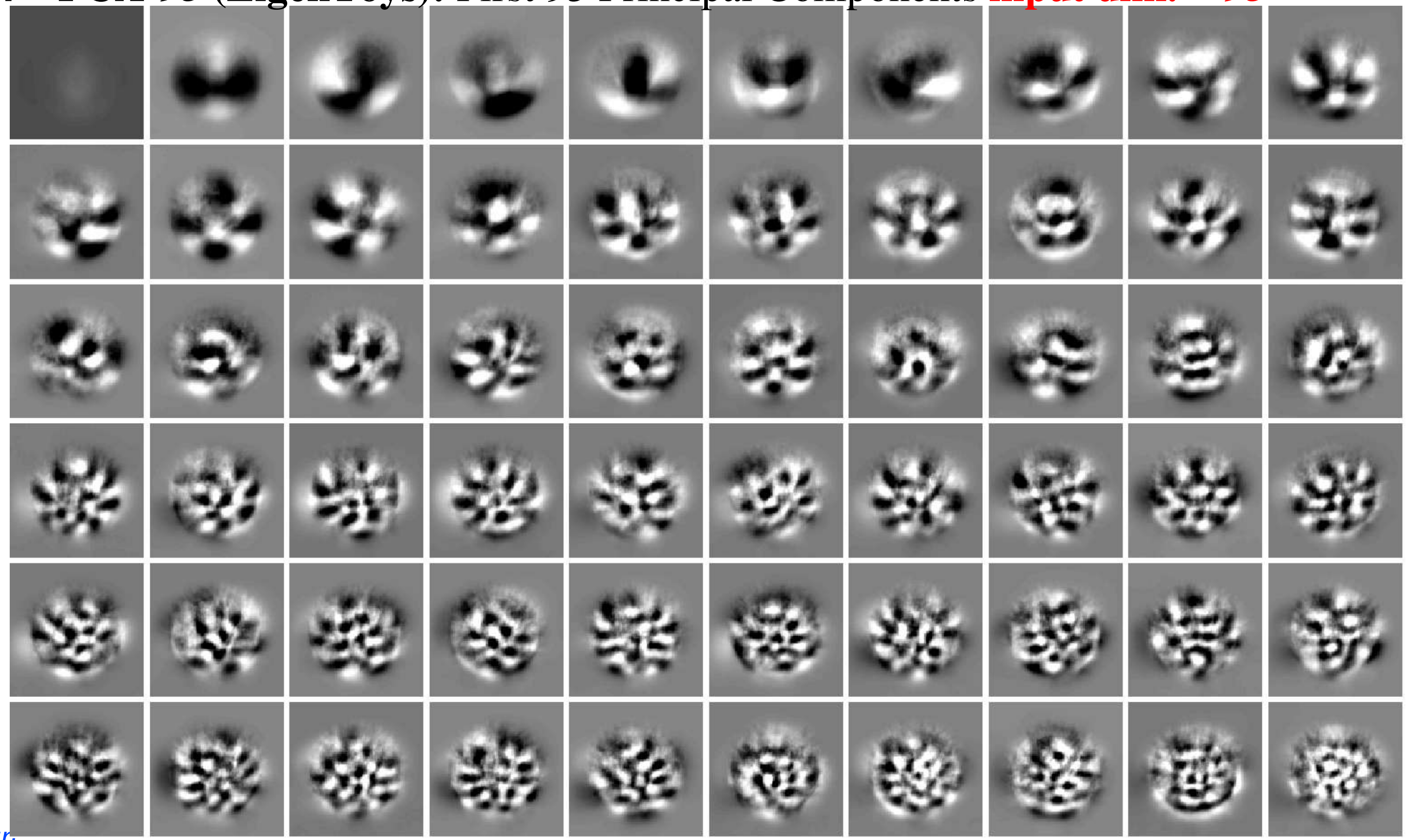
Experiment 1: Normalized-Uniform Dataset



Training instances **Test instances**

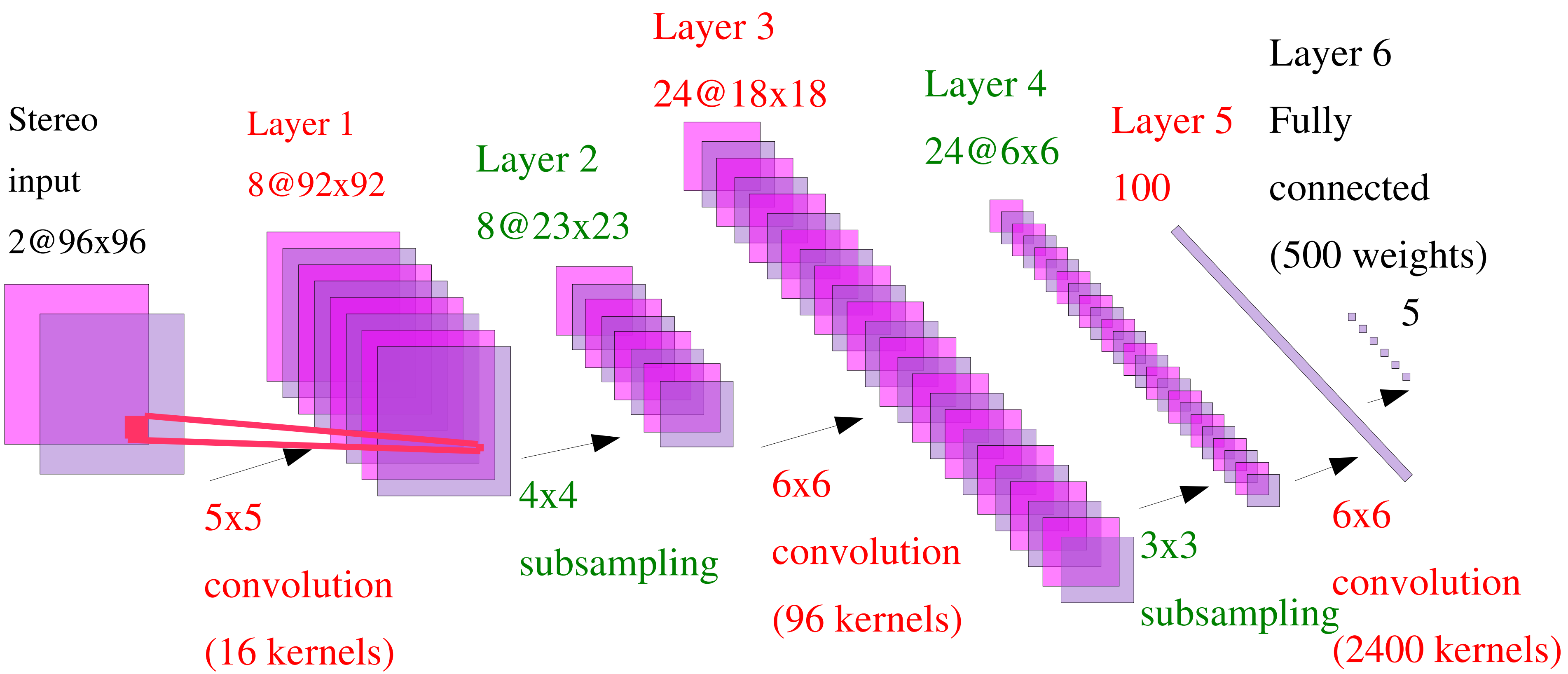
Experiment 1: Normalized-Uniform Set: Representations

- **1 - Raw Stereo Input:** 2 images 96x96 pixels **input dim. = 18432**
- **2 - Raw Monocular Input:**1 image, 96x96 pixels **input dim. = 9216**
- **3 – Subsampled Mono Input:** 1 image, 32x32 pixels **input dim = 1024**
- **4 – PCA-95 (EigenToys):** First 95 Principal Components **input dim. = 95**



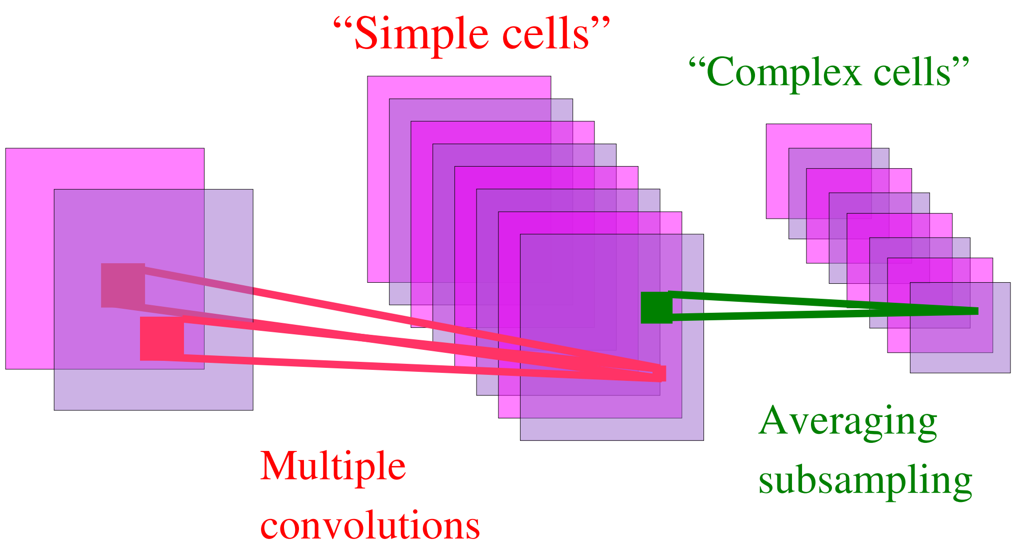
First 60 eigenvectors (EigenToys)

Convolutional Network

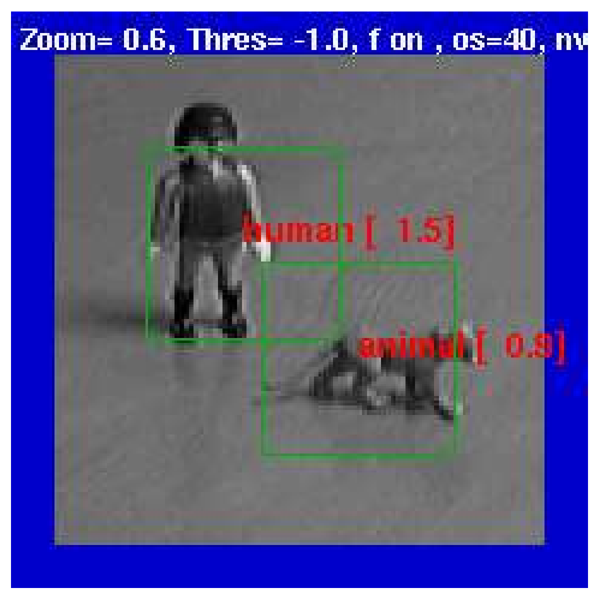


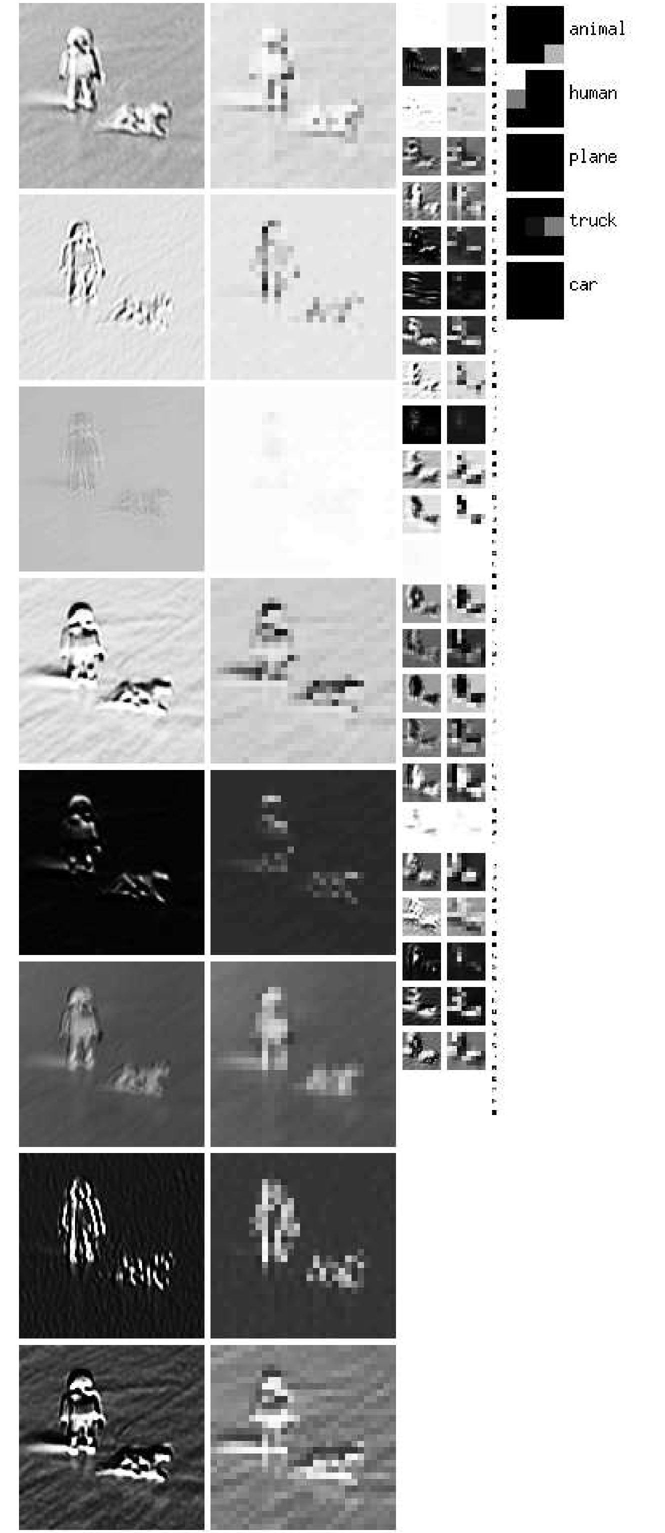
- **90,857 free parameters, 3,901,162 connections.**
- The architecture alternates convolutional layers (feature detectors) and subsampling layers (local feature pooling for invariance to small distortions).
- **The entire network is trained end-to-end** (all the layers are trained simultaneously).
- A gradient-based algorithm is used to minimize a supervised loss function.

Alternated Convolutions and Subsampling



- Local features are extracted everywhere.
- averaging/subsampling layer builds robustness to variations in feature locations.
- Hubel/Wiesel'62, Fukushima'71, LeCun'89, Riesenhuber & Poggio'02, Ullman'02,....



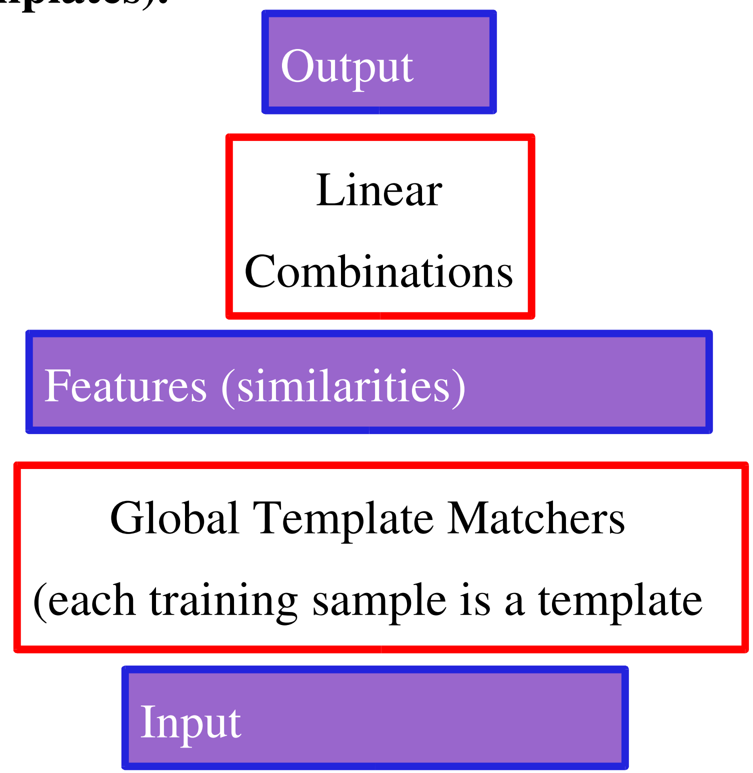


Experiment 1: Normalized-Uniform Set: Error Rates

- **Linear Classifier on raw stereo images:** **30.2% error.**
- **K-Nearest-Neighbors on raw stereo images:** **18.4% error.**
- **K-Nearest-Neighbors on PCA-95:** **16.6% error.**
- **Pairwise SVM on 96x96 stereo images:** **14.1% error**
- **Pairwise SVM on 48x48 stereo images:** **12.5% error**
- **Pairwise SVM on 32x32 stereo images:** **11.8% error.**
- **Pairwise SVM on 48x48 monocular images:** **13.9% error.**
- **Pairwise SVM on 32x32 monocular images:** **12.6% error.**
- **Pairwise SVM on 95 Principal Components** **13.3% error.**
- **Convolutional Net on 32x32 stereo images:** **11.3% error.**
- **Convolutional Net on 48x48 stereo images:** **8.7% error.**
- **Convolutional Net on 96x96 stereo images:** **6.6% error.**

What's wrong with K-NN and SVMs?

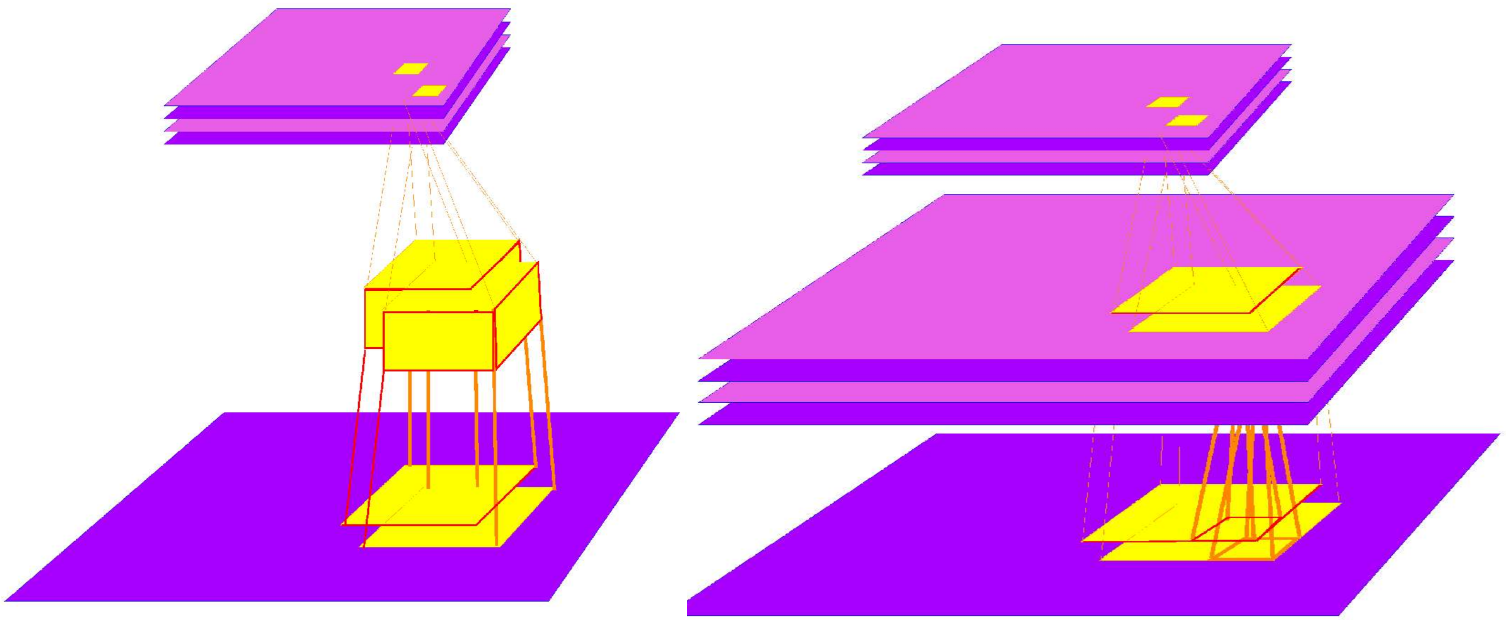
- **K-NN and SVM with Gaussian kernels are based on matching global templates**
- **Both are "shallow" architectures**
- **There is now way to learn invariant recognition tasks with such naïve architectures (unless we use an impractically large number of templates).**
  - **The number of necessary templates grows exponentially with the number of dimensions of variations.**
  - **Global templates are in trouble when the variations include: category, instance shape, configuration (for articulated object), position, azimuth, elevation, scale, illumination, texture, albedo, in-plane rotation, background luminance, background texture, background clutter, ...**

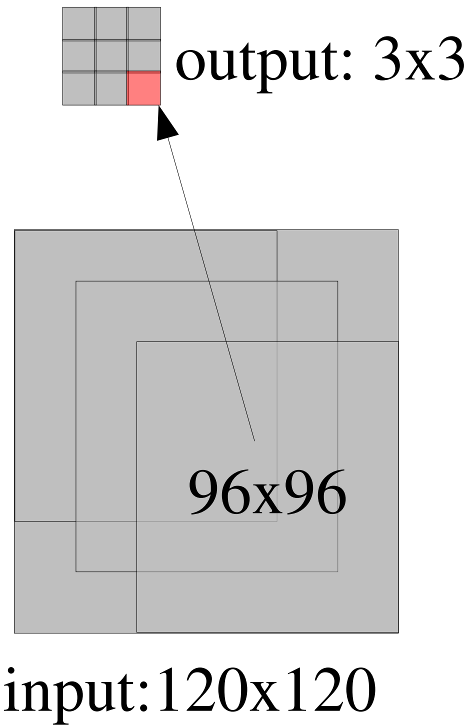


Experiment 2: Jittered-Cluttered Dataset

- **291,600 training samples, 58,320 test samples**
- **Convolutional Net with binocular input: 7.8% error**
- **Convolutional Net + SVM on top: 5.8% error**
- **Convolutional Net with monocular input: 20.8% error**
- **Smaller mono net (DEMO): 26.0% error**
- **Dataset available from http://www.cs.nyu.edu/~yann**

Building a Detector/Recognizer: Replicated Conv. Nets

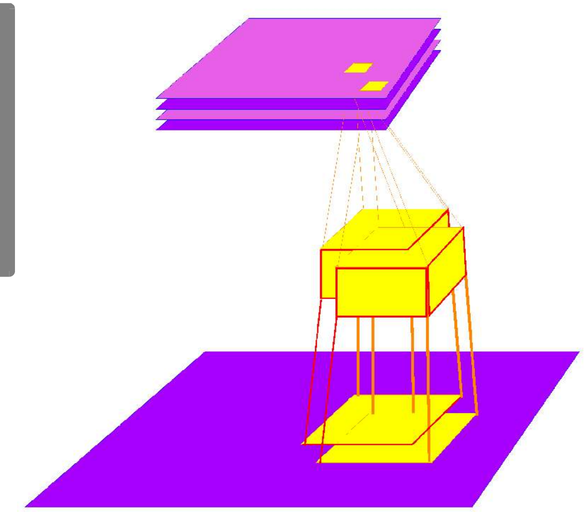


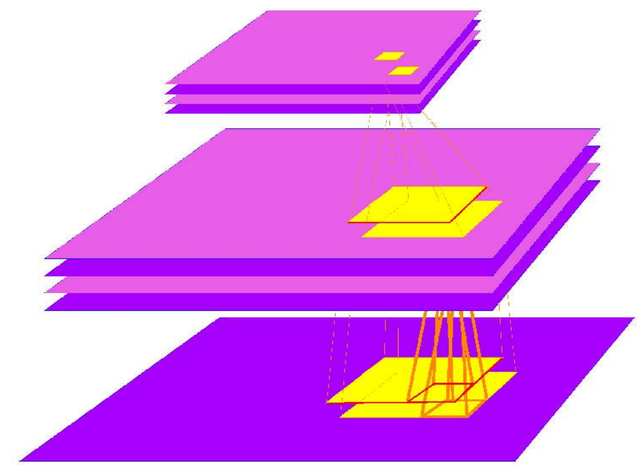


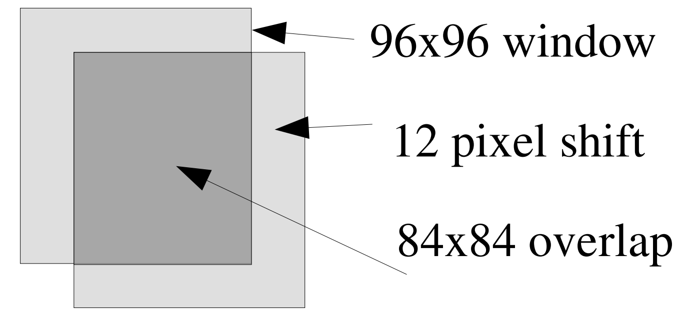
- Traditional Detectors/Classifiers must be applied to every location on a large input image, at multiple scales.
- Convolutional nets can replicated over large images very cheaply.
- The network is applied to multiple scales spaced by 1.5.

Building a Detector/Recognizer: Replicated Convolutional Nets

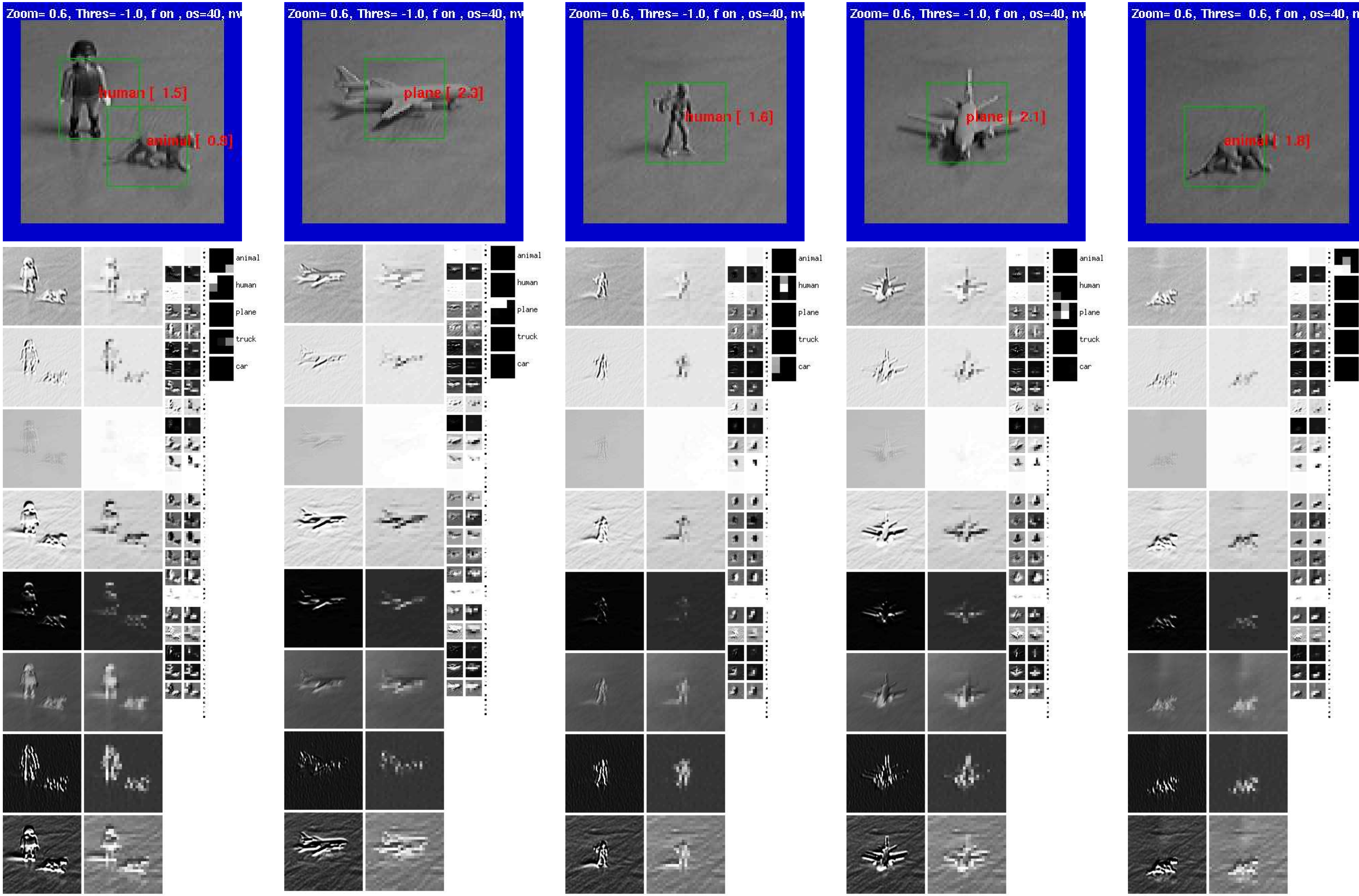
- Computational cost for replicated convolutional net:
  - 96x96 -> 4.6 million multiply-accumulate operations
  - 120x120 -> 8.3 million multiply-accumulate operations
  - 240x240 -> 47.5 million multiply-accumulate operations
  - 480x480 -> 232 million multiply-accumulate operations
- Computational cost for a non-convolutional detector of the same size, applied every 12 pixels:
  - 96x96 -> 4.6 million multiply-accumulate operations
  - 120x120 -> 42.0 million multiply-accumulate operations
  - 240x240 -> 788.0 million multiply-accumulate operations
  - 480x480 -> 5,083 million multiply-accumulate operations



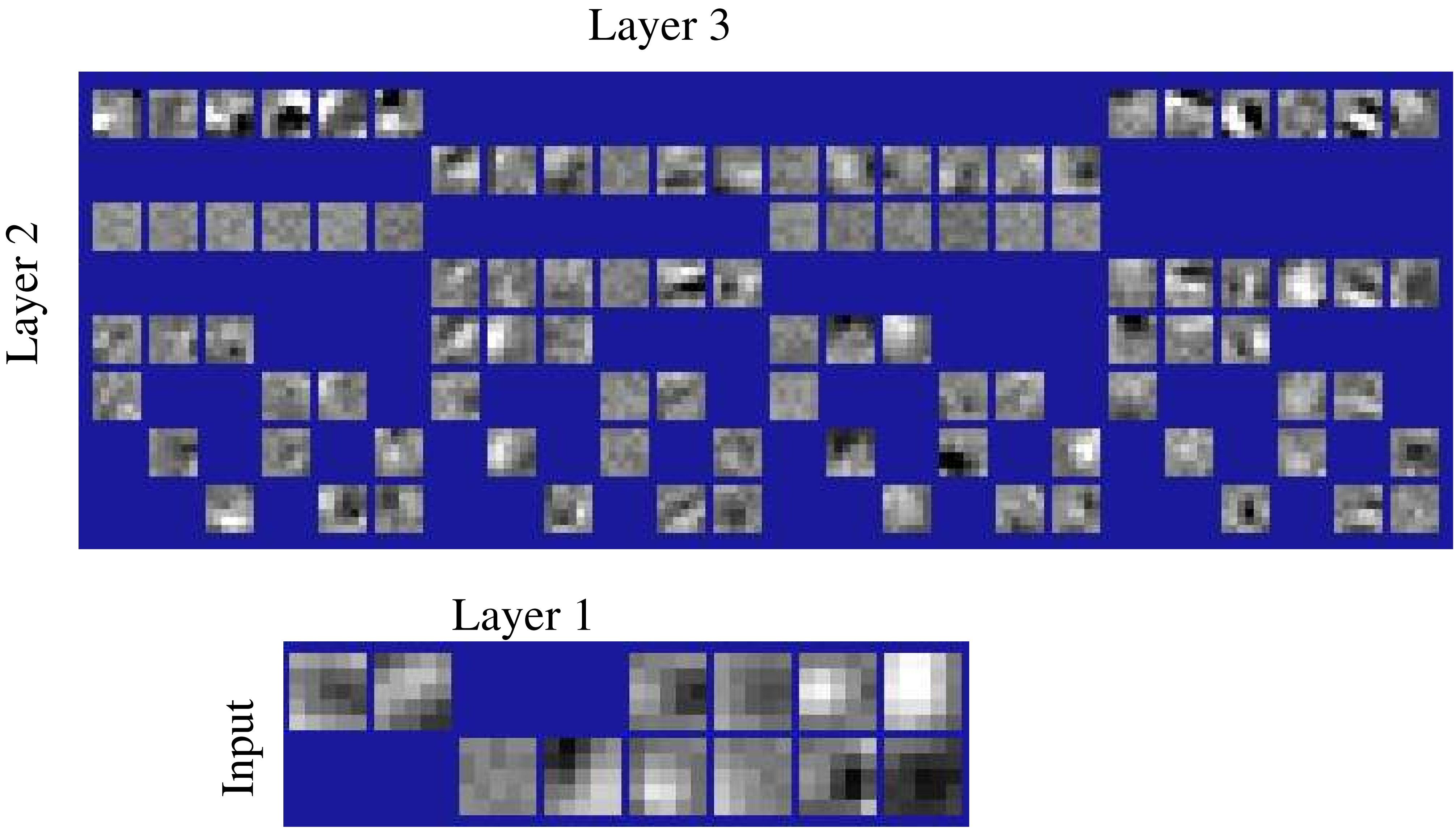




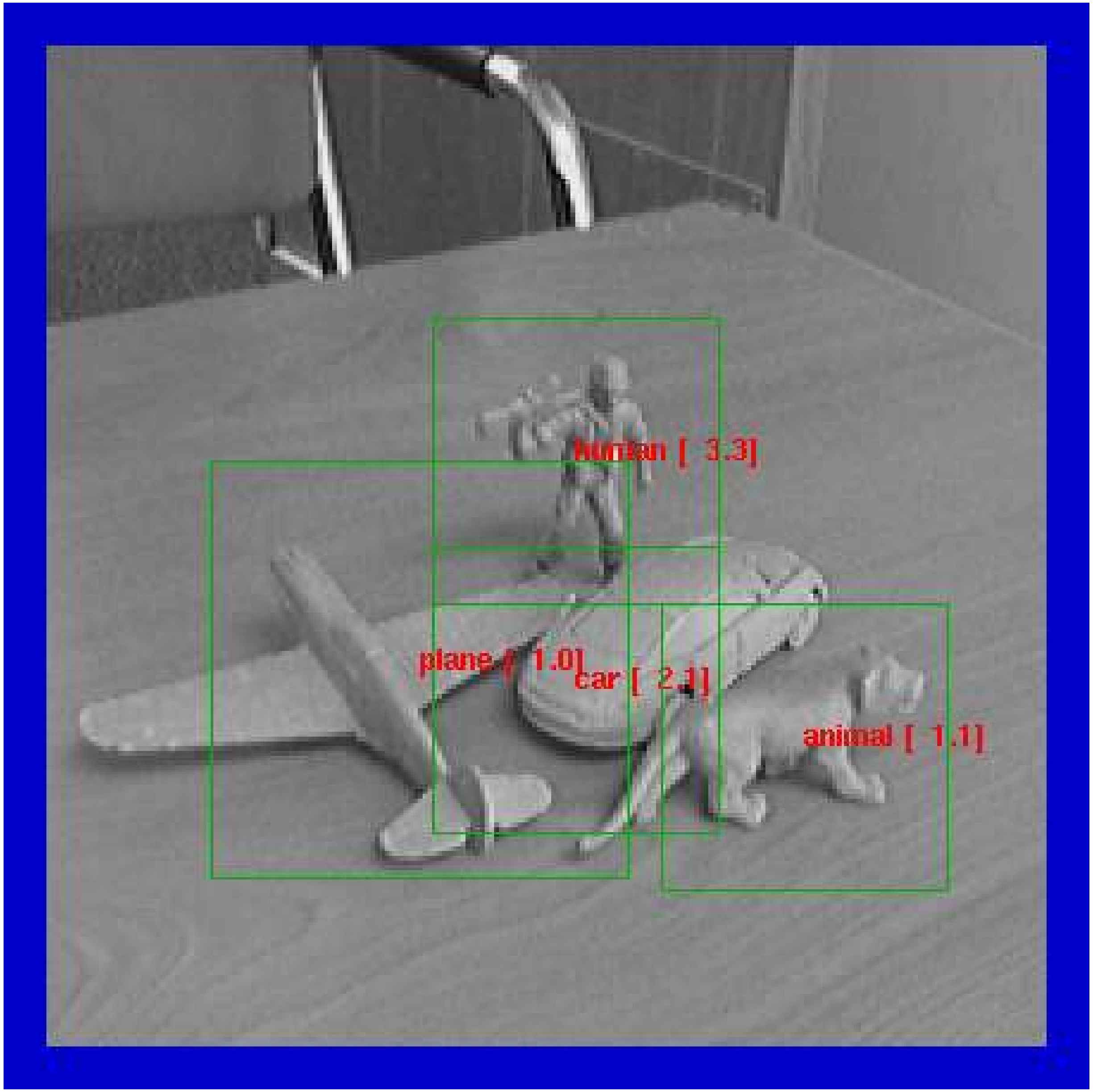
Examples (Monocular Mode)



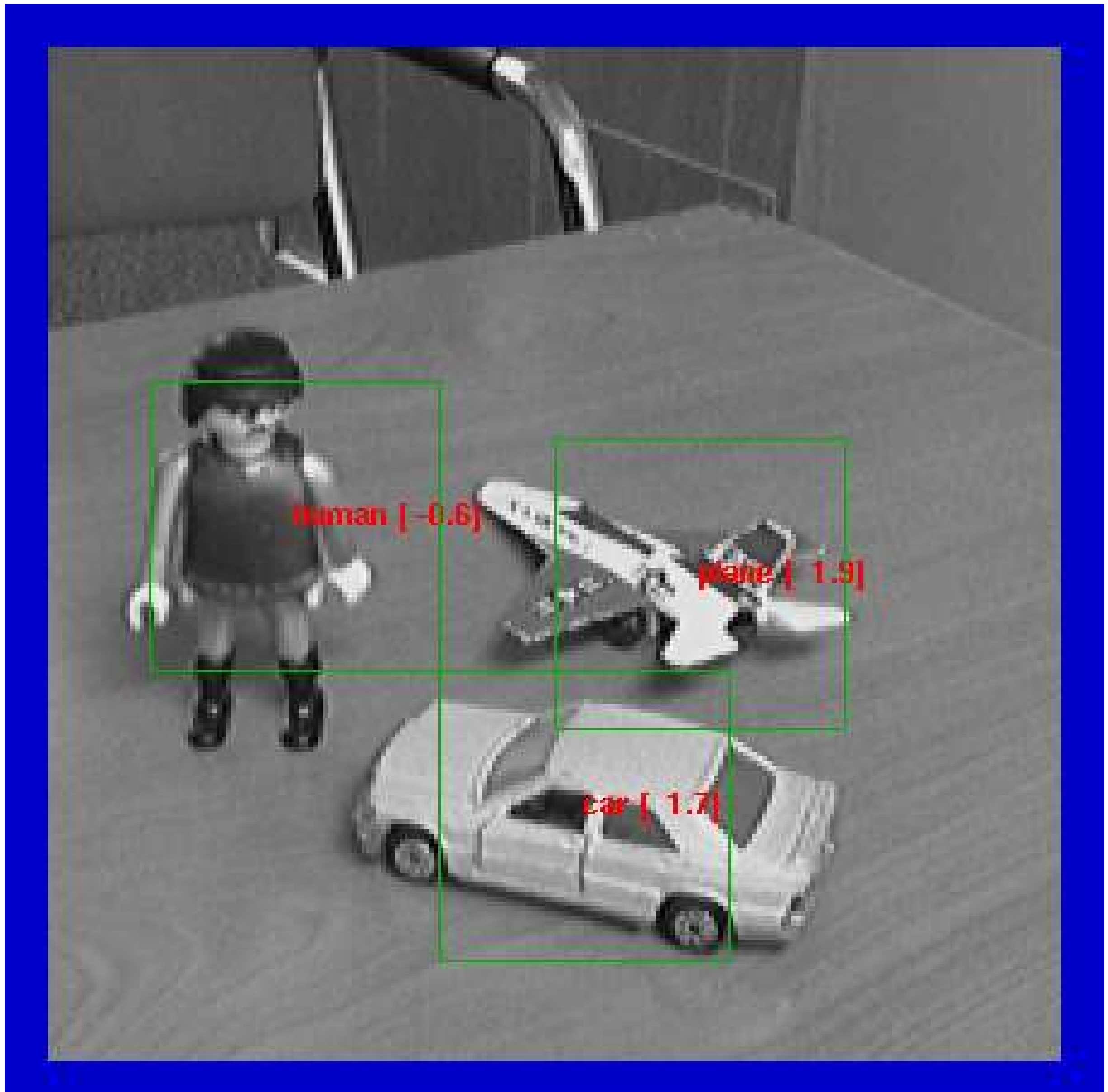
Learned Features



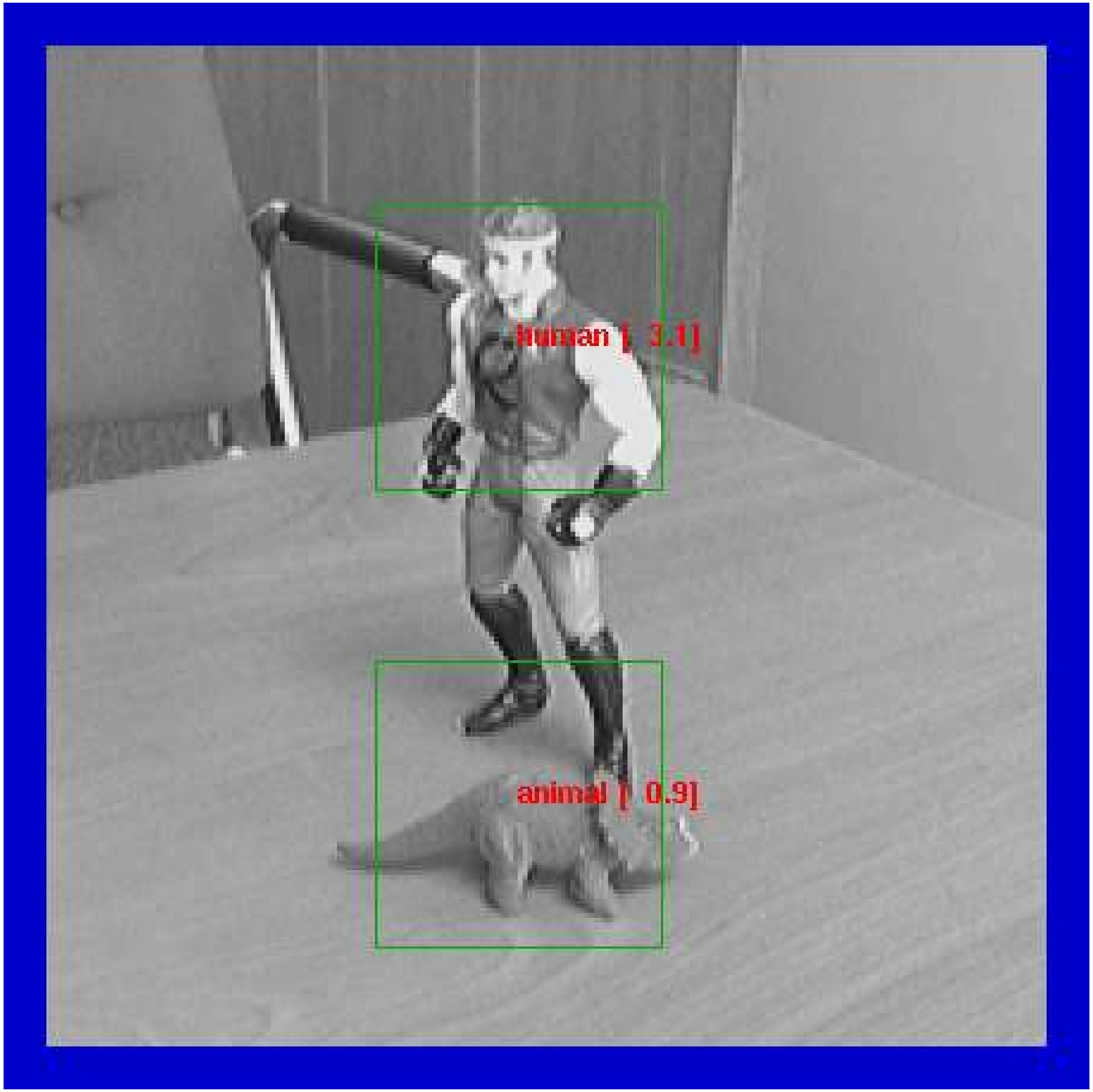
Examples (Monocular Mode)



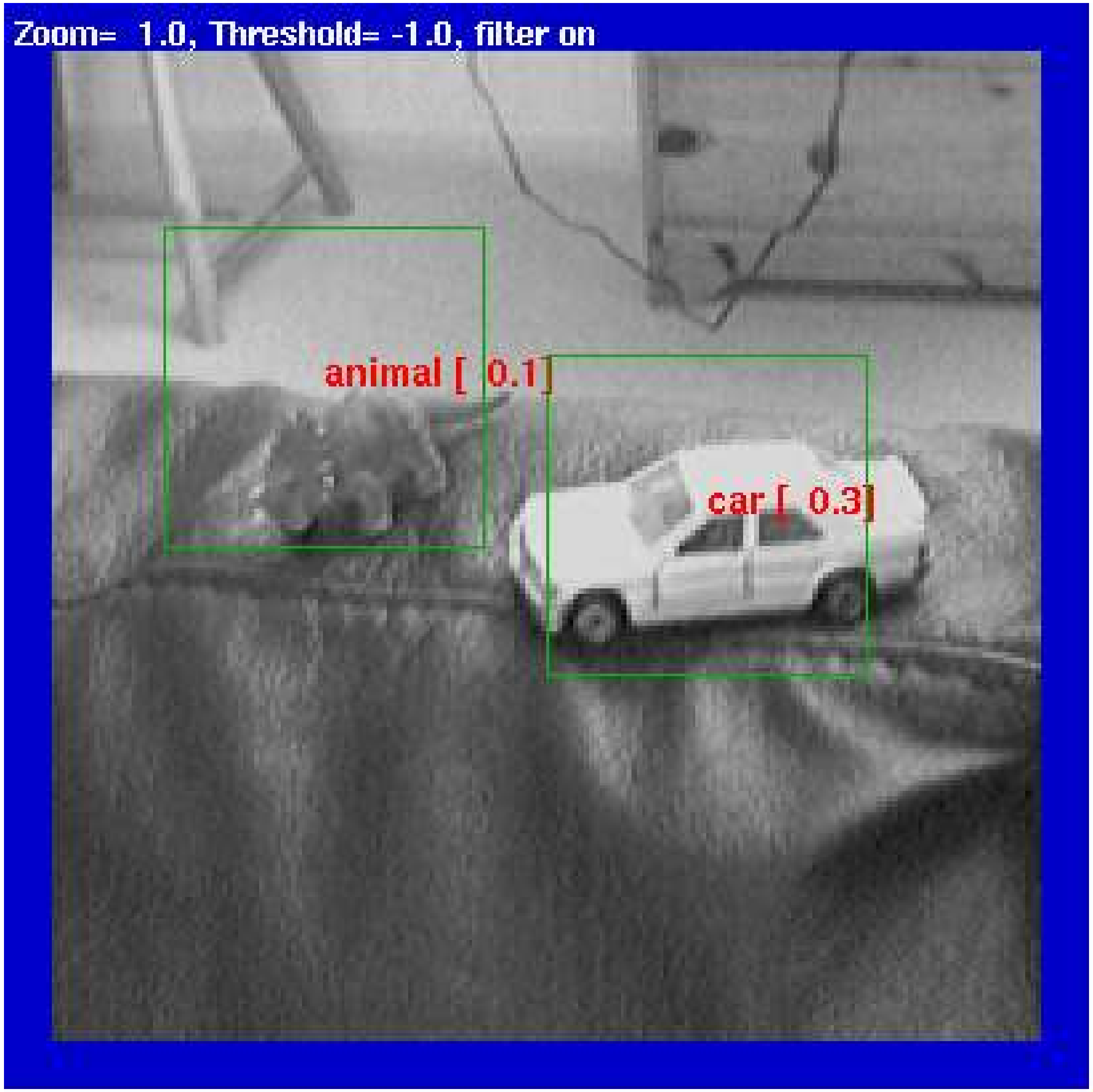
Examples (Monocular Mode)



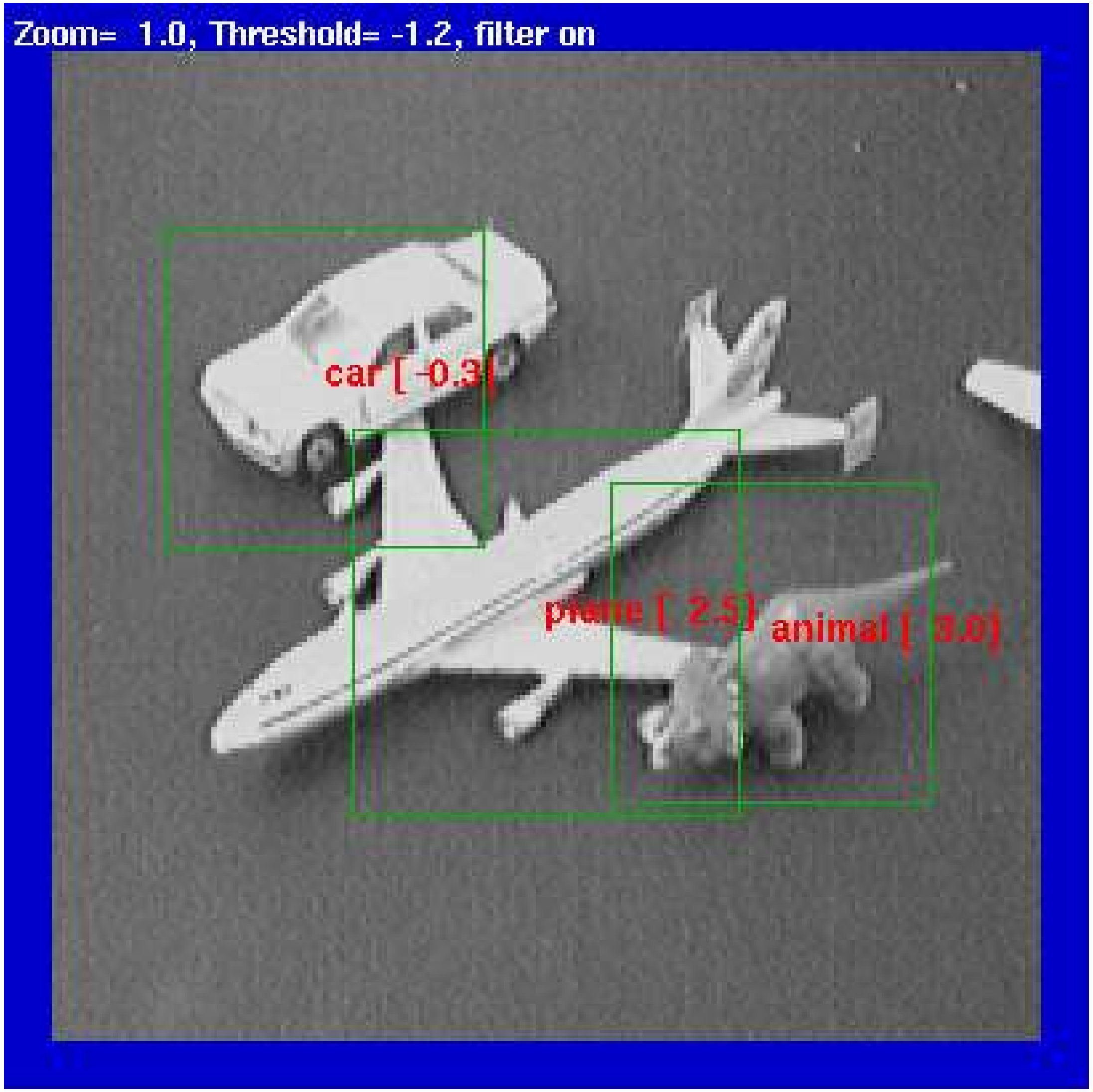
Examples (Monocular Mode)



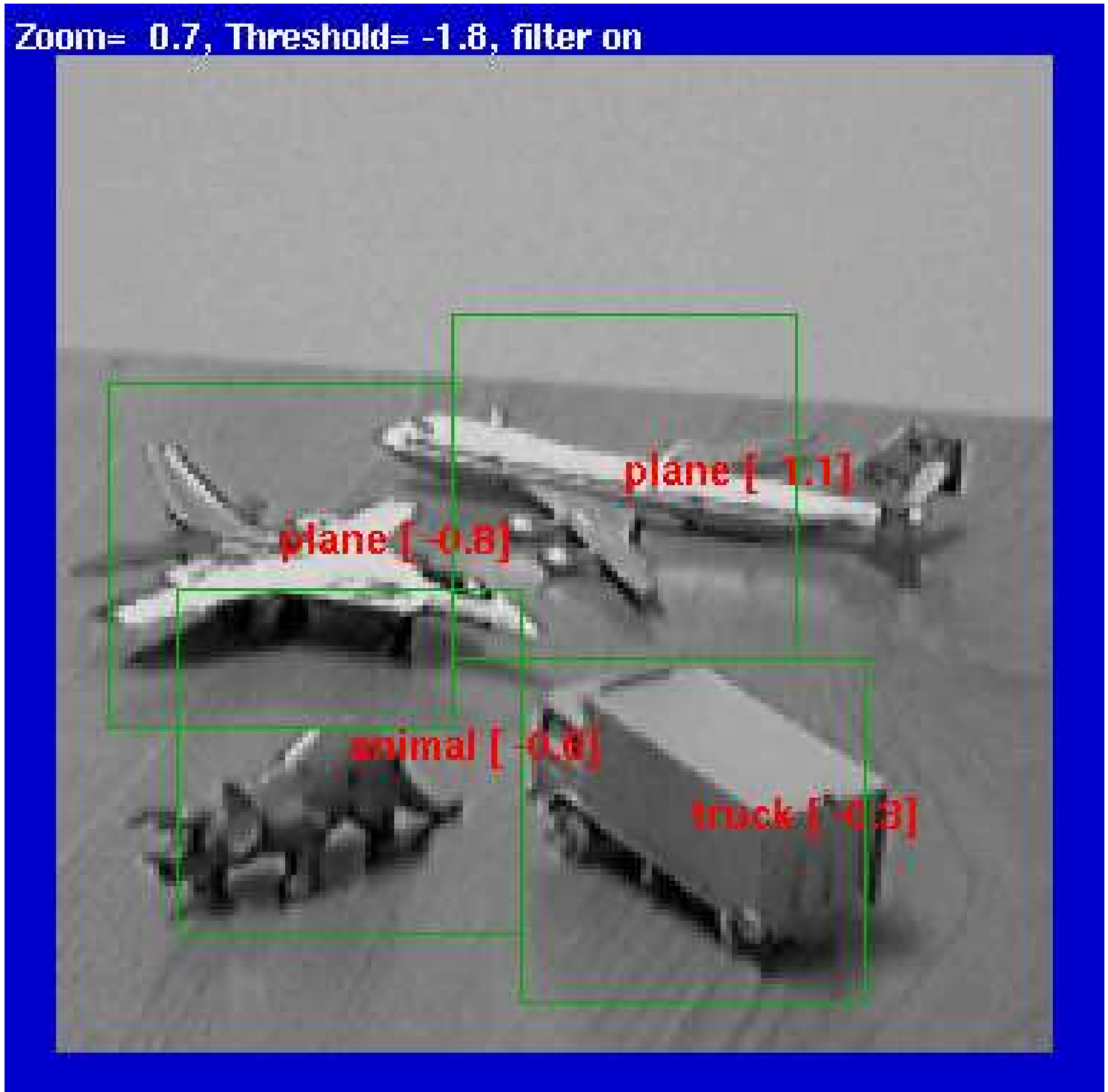
Examples (Monocular Mode)



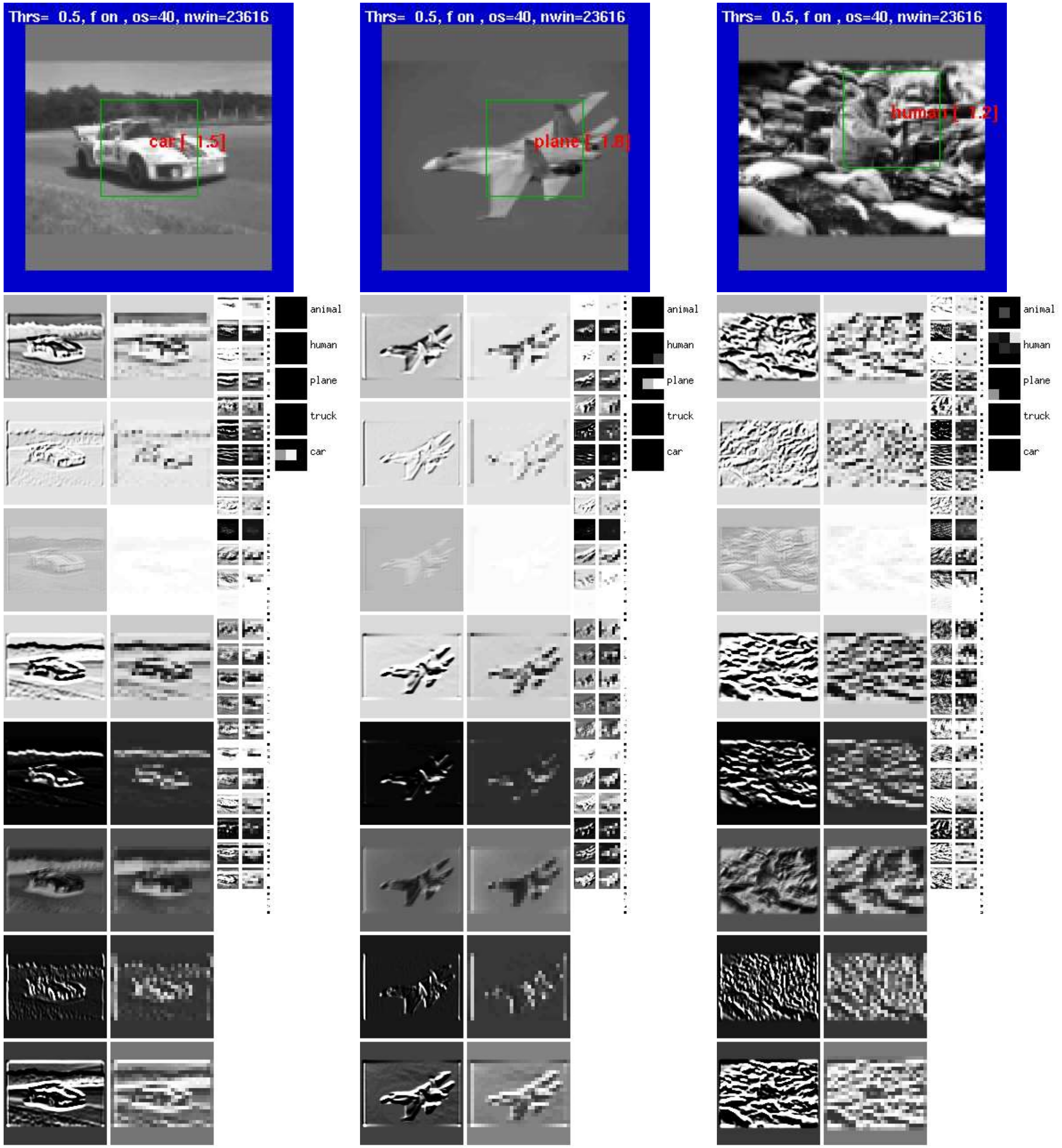
Examples (Monocular Mode)



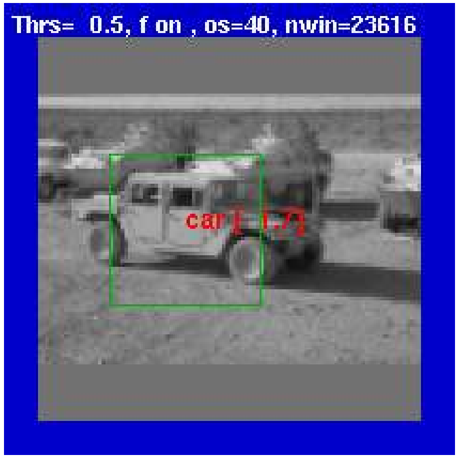
Examples (Monocular Mode)

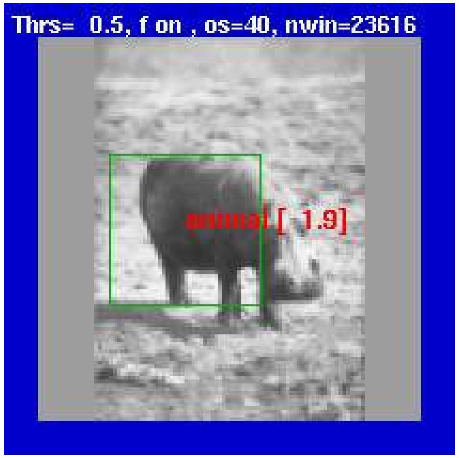


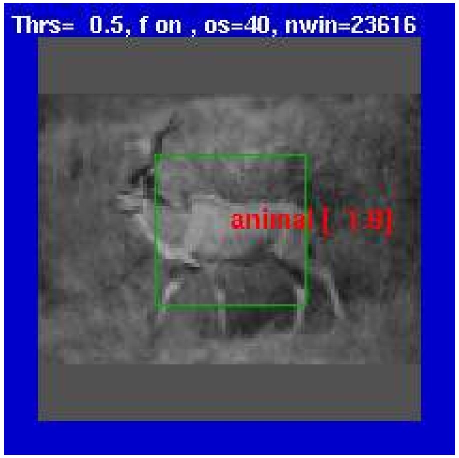
Natural Images (Monocular Mode)

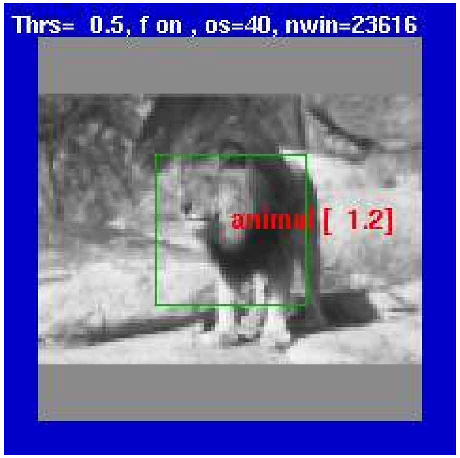


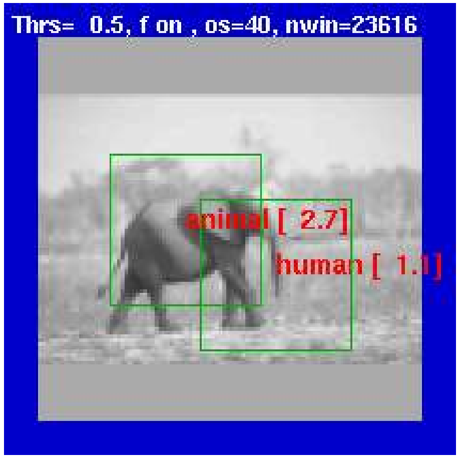
Natural Images (Monocular Mode)

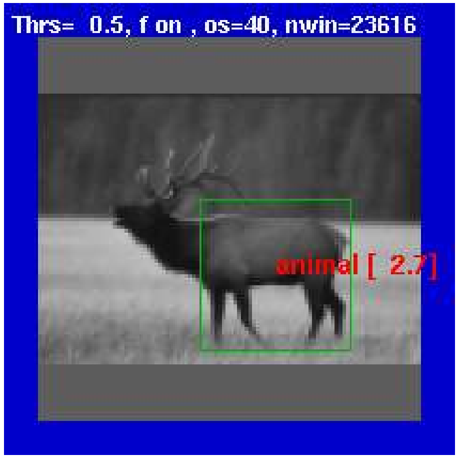


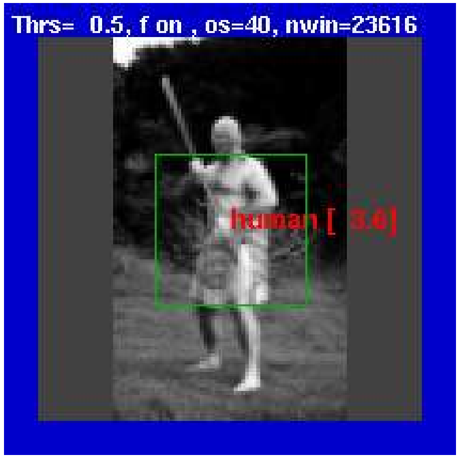




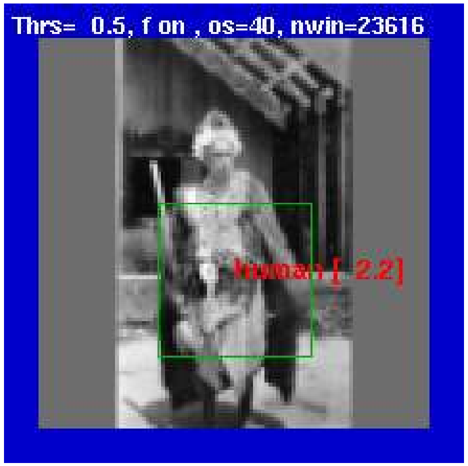


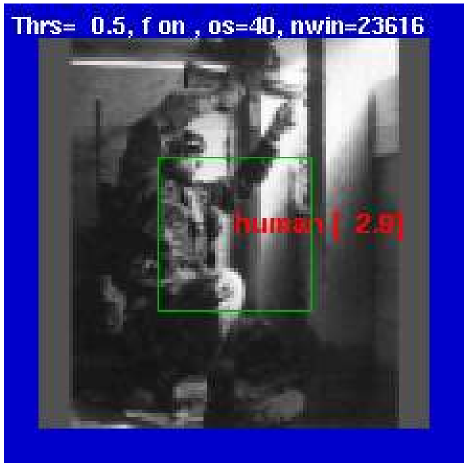


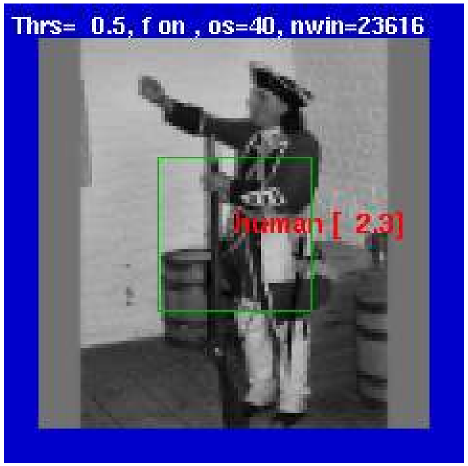


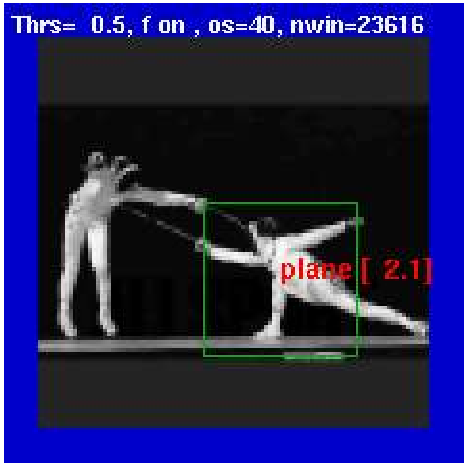


Natural Images (Monocular Mode)



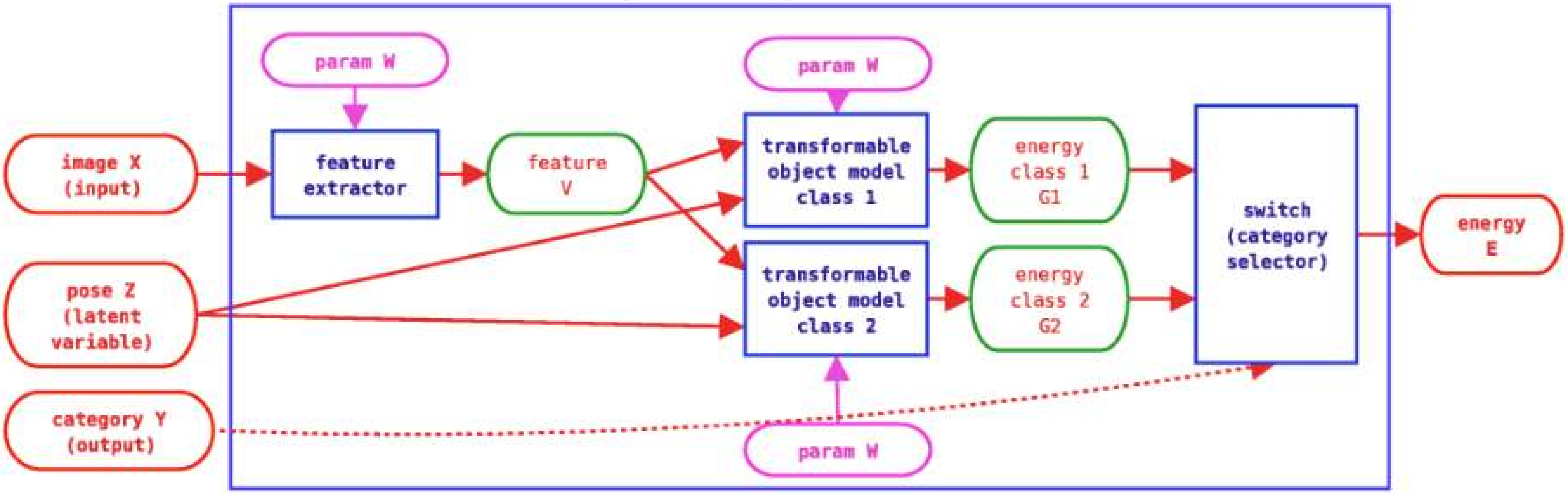






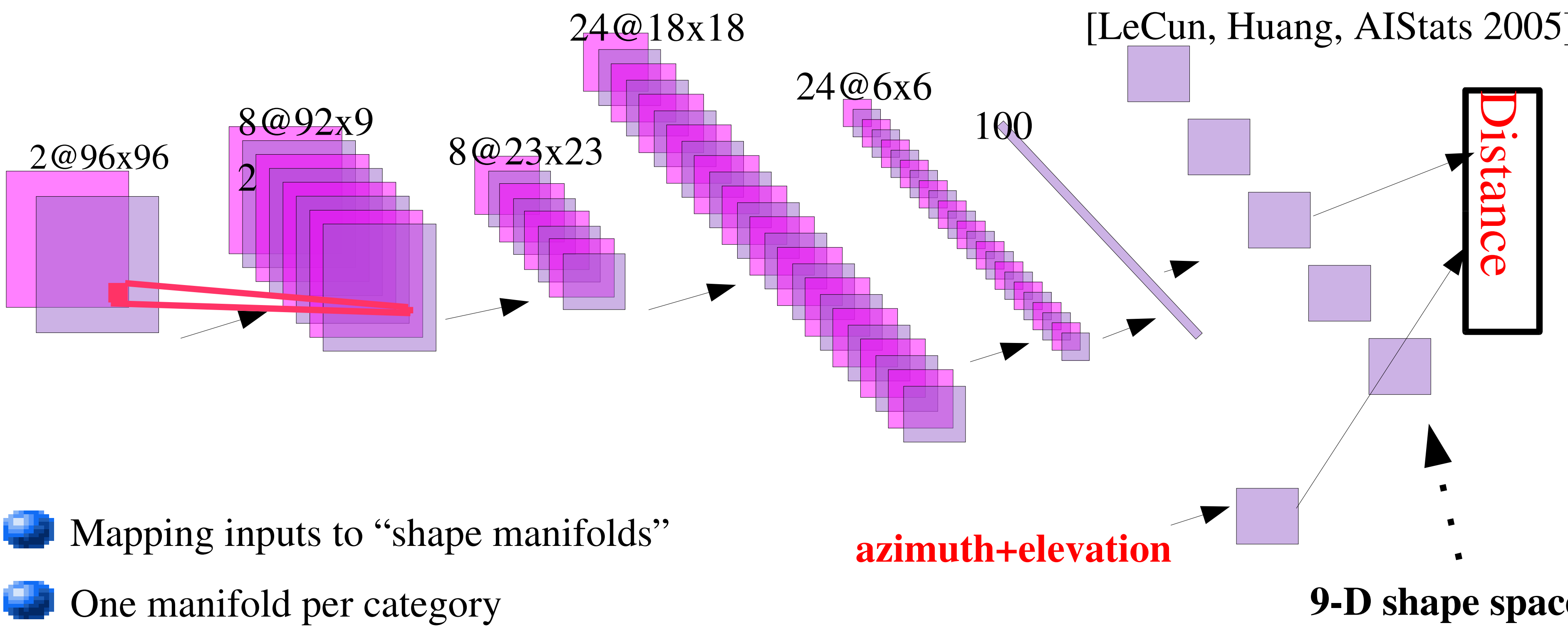
EBM with Latent Variable for Pose Invariance

EBM Architecture for invariant object recognition

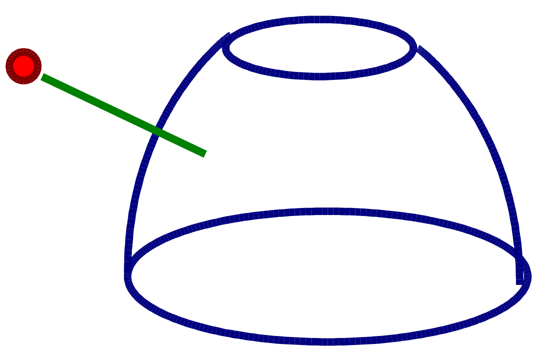


Each object model matches the output of the feature extractor to a reference representation that is transformed by the pose parameters.
Inference finds the category and the pose that minimize the energy.

EBM with a latent pose variable

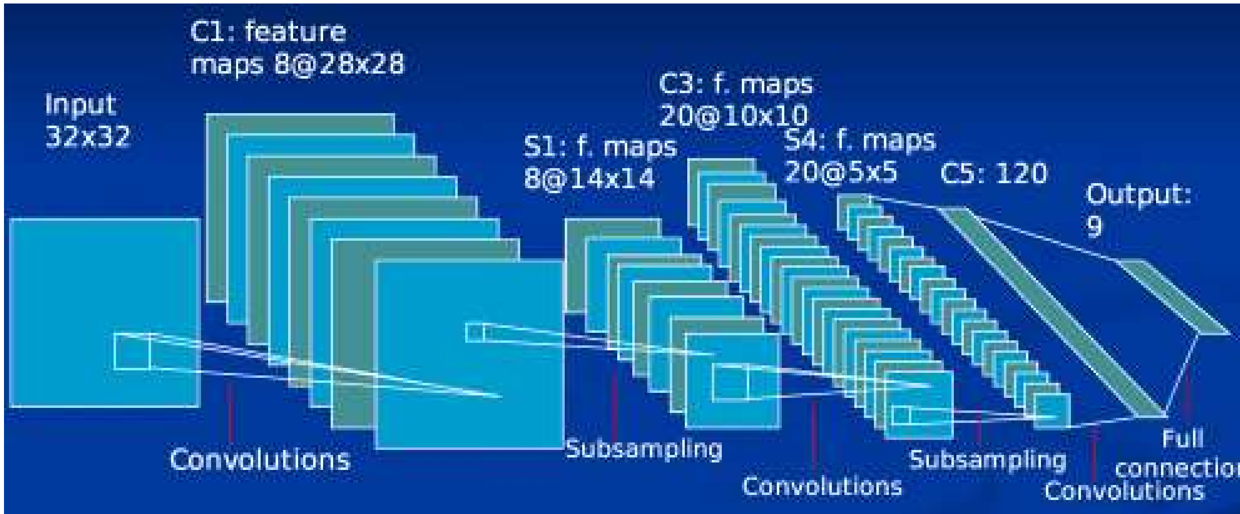


- Mapping inputs to "shape manifolds"
- One manifold per category
- Each manifold is a 2-D half sphere embedded in a 9-D space
- 2 latent variables parameterize position on the manifold (azimuth and elevation).
- **Loss function:** pulls the network output toward manifold of deesird class, and repel from manifolds of non-desired classes.

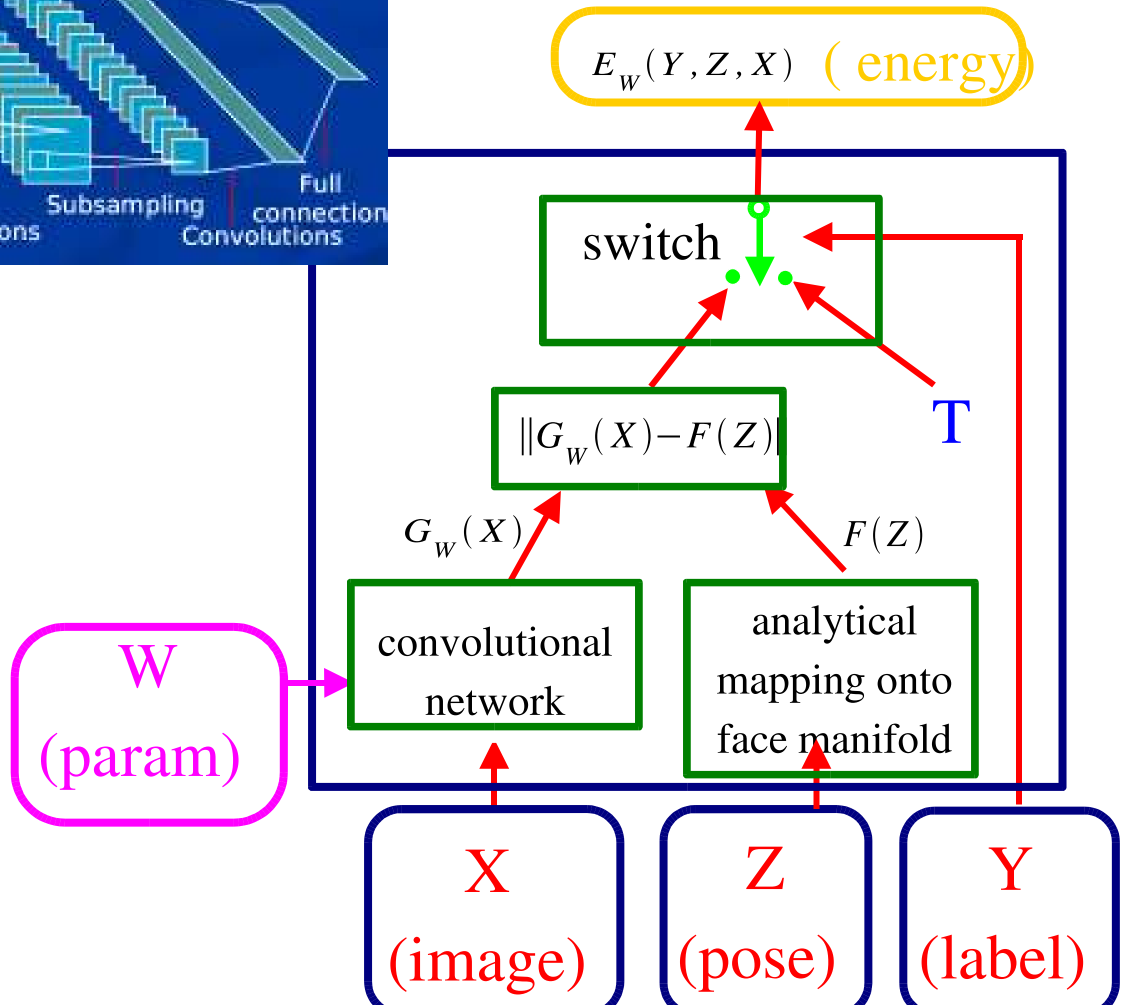


- **Result on uniform set: 5.1% error (vs 6.6%)**

Face Detection and Pose Estimation with a Convolutional EBM



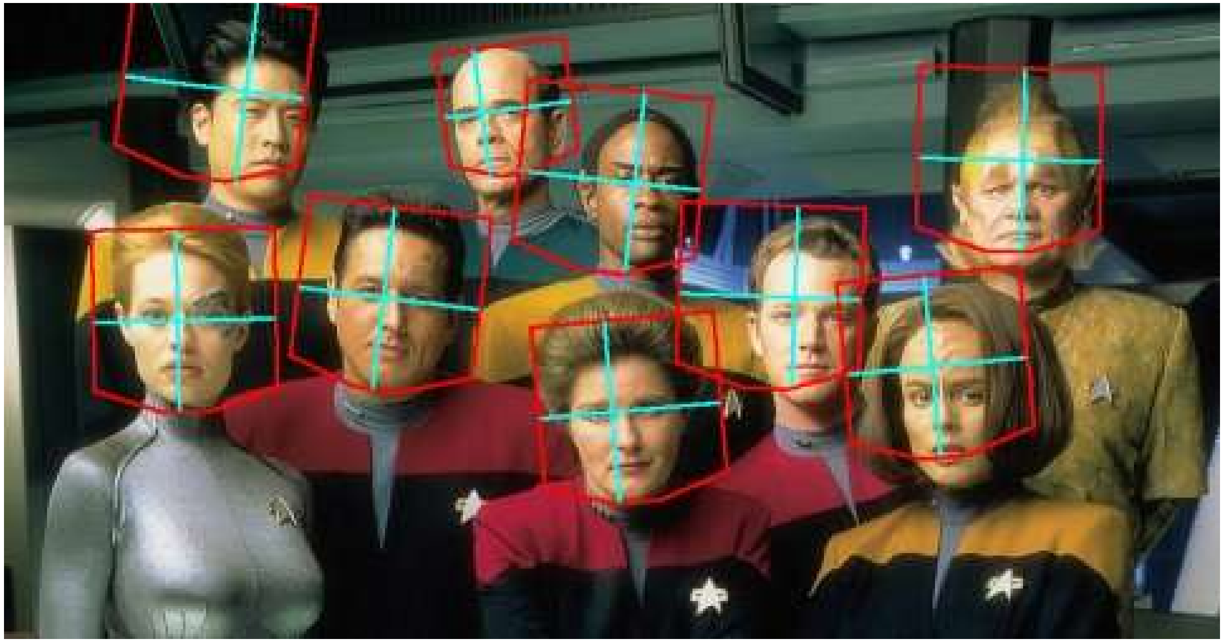
[Osadchy, Miller, LeCun, NIPS 2004]

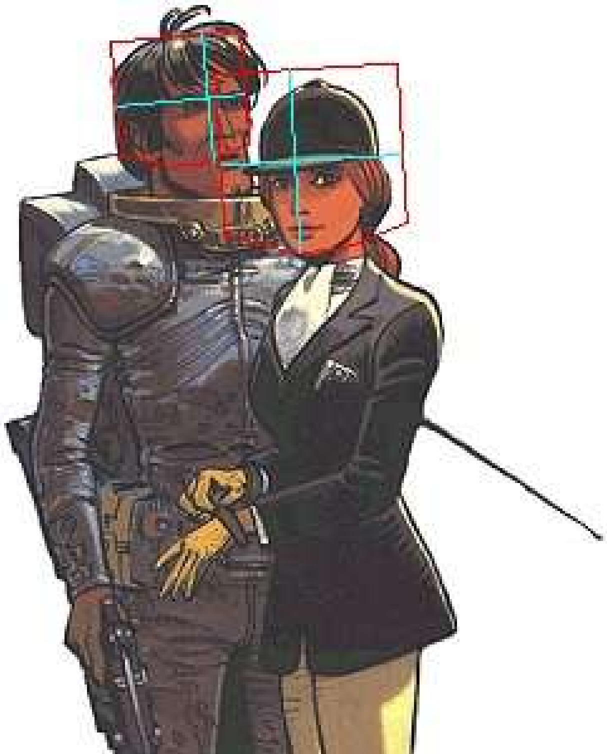


- **Training:** 52,850, 32x32 grey-level images of faces, 52,850 non-faces.
- Each training image was used 5 times with random variation in scale, in-plane rotation, brightness and contrast.
- 2^{nd} **phase:** half of the initial negative set was replaced by false positives of the initial version of the detector .

Face Detection: Results

Data Set->	TILTED		PROFILE		MIT+CMU	
False positives per image->	4.42	26.9	0.47	3.36	0.5	1.28
Our Detector	90%	97%	67%	83%	83%	88%
Jones & Viola (tilted)	90%	95%	x		x	
Jones & Viola (profile)	x		70%	83%	x	

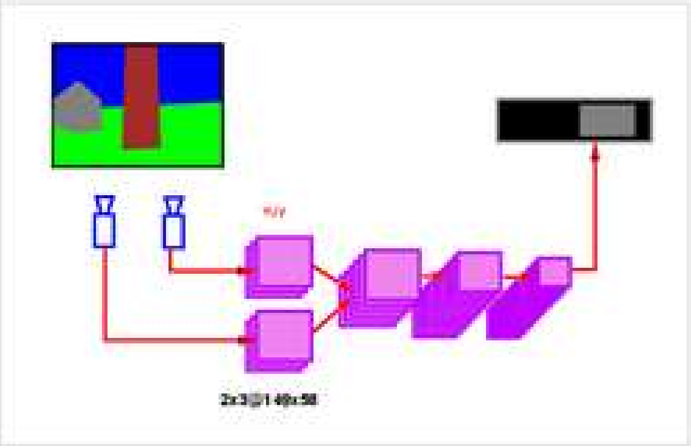




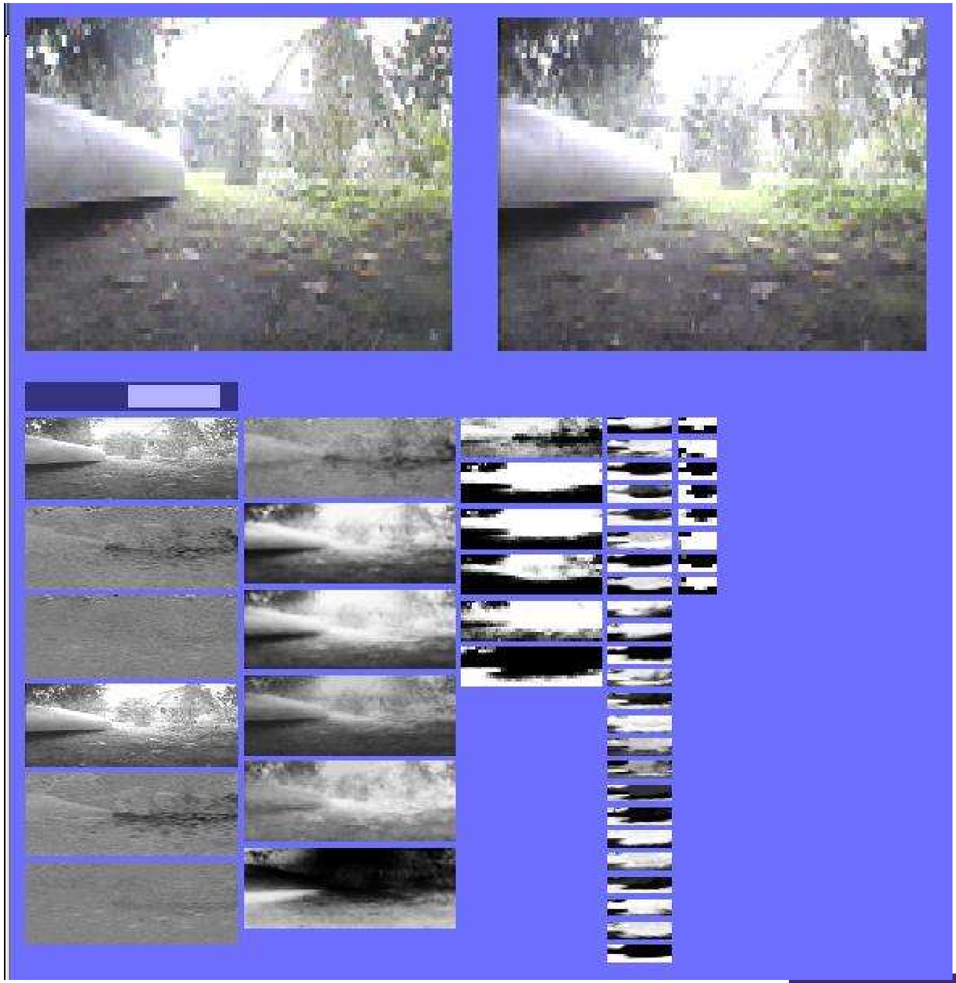
Face Detection: Results

Face Detection with a Convolutional Net

Visual Navigation for a Mobile Robot



- Mobile robot with two cameras
- The convolutional net is trained to emulate a human driver from recorded sequences of video + human-provided steering angles.
- The network maps stereo images to steering angles for obstacle avoidance



Invariant Object Recognition

- The old feed-forward architecture can do more than expected.
- Full invariance to viewpoint and illumination for detecting and recognizing objects can be learned discriminatively by a simple feed-forward architecture.
- With only 5 training instances from each category, the model can detect and recognize new instances with high accuracy.
- The model outperforms "traditional" template-based classifiers operating on raw pixels or on PCA features.
- The system takes advantage of the binocular input.
- The convolutional net architecture is generic, and can be applied to a variety of vision tasks with essentially no change.
- Feature tuning produces very parcimonious systems with only a small number of feature detectors at each layer.
- Invariance can be achieved with "deep" architectures, containing mutiple, successive layers of feature detection and feature integration/subsampling (Hubel/Wiesel'62, Fukushima'72, LeCun'89, Ullman'02, Riesenhuber/Poggio'02).