

DCCK & more

luca@psvl

agenda

- Panasonic
- PSVL
- DL@PSVL
- DCCK
- Q&A

Panasonic

Company Overview

President	Kazuhiro Tsuga
Foundation	March, 1918(incorporated in December, 1935)
Net Sales*	7,303.0 billion yen = ~\$72B
Number of Employees*	293,742
Number of Consolidated Companies*	538 (including parent company)

*as of March 31, 2013

Brief History

1918 Konosuke Matsushita founded Matsushita Electric Housewares Manufacturing Works. Sales of the Company's first product, an improved attachment plug, began.

1927 The "National" brand name was registered.

1935 The Company was reorganized and renamed Matsushita Electric Industrial Co., Ltd.

1949 The Company's shares were listed on the Tokyo Stock Exchange and the Osaka Securities Exchange.*

1951 The Company's shares were listed on the Nagoya Stock Exchange.

1959 Matsushita Electric Corporation of America was established. Following this move, the Company established bases in other parts of the world.

1971 The Company's shares were listed on the New York Stock Exchange.*

2002 The Company made Matsushita Communication Industrial Co., Ltd., Kyushu Matsushita Electric Co., Ltd., Matsushita Seiko Co., Ltd., Matsushita Kotobuki Electronics Industries, Ltd. and Matsushita Graphic Communication Systems, Inc. into wholly-owned subsidiaries.

2003 The Company adopted a business domain-based organizational style through restructuring. The Company made Matsushita Electronic Components Co., Ltd. and Matsushita Battery Industrial Co., Ltd. into wholly-owned subsidiaries.

2004 The Company made Matsushita Electric Works, Ltd. (later renamed Panasonic Electric Works Co., Ltd.), PanaHome Corporation and their subsidiaries into consolidated subsidiaries.

2008 The Company changed its name from Matsushita Electric Industrial Co., Ltd. to Panasonic Corporation. The Company planned to unify its corporate brands under the "Panasonic" name across the world.

2009 The Company made SANYO Electric Co., Ltd. and its subsidiaries into consolidated subsidiaries.

2011 The Company made Panasonic Electric Works Co., Ltd. and SANYO Electric Co., Ltd. into wholly-owned subsidiaries.

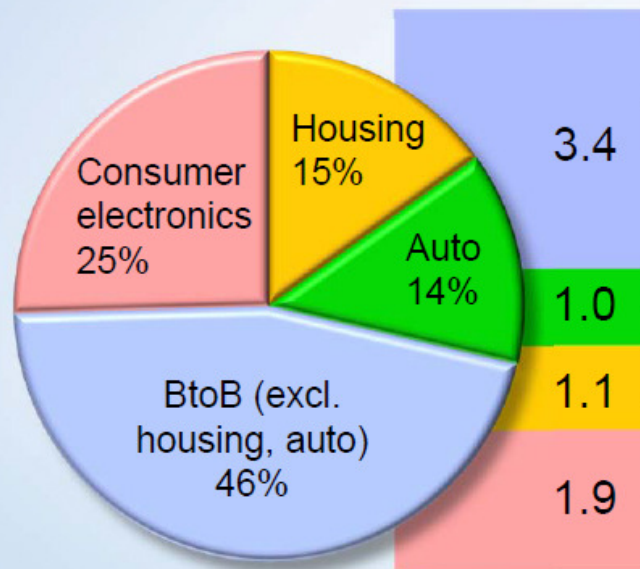
2013 The Company restructured its Group organization, and introduced divisional management and a 4 Divisional Company system on April 1.



But, B2B sales are higher now

(yen: trillions) = ~ \$9.8B

7.3 = ~\$72B



FY2013

BtoB (excl. housing, auto)

ICT device, industrial equipment
Avionics, lighting for commercial use,
cold chain equipment, business PC, ...

Auto

Car navigation, car AV equipment,
battery, sensor, camera, ...

Housing

Exterior, building materials, building
equipment, electric equipment,
housing (PanaHome), ...

PSVL

Last standing building 😊

Cupertino

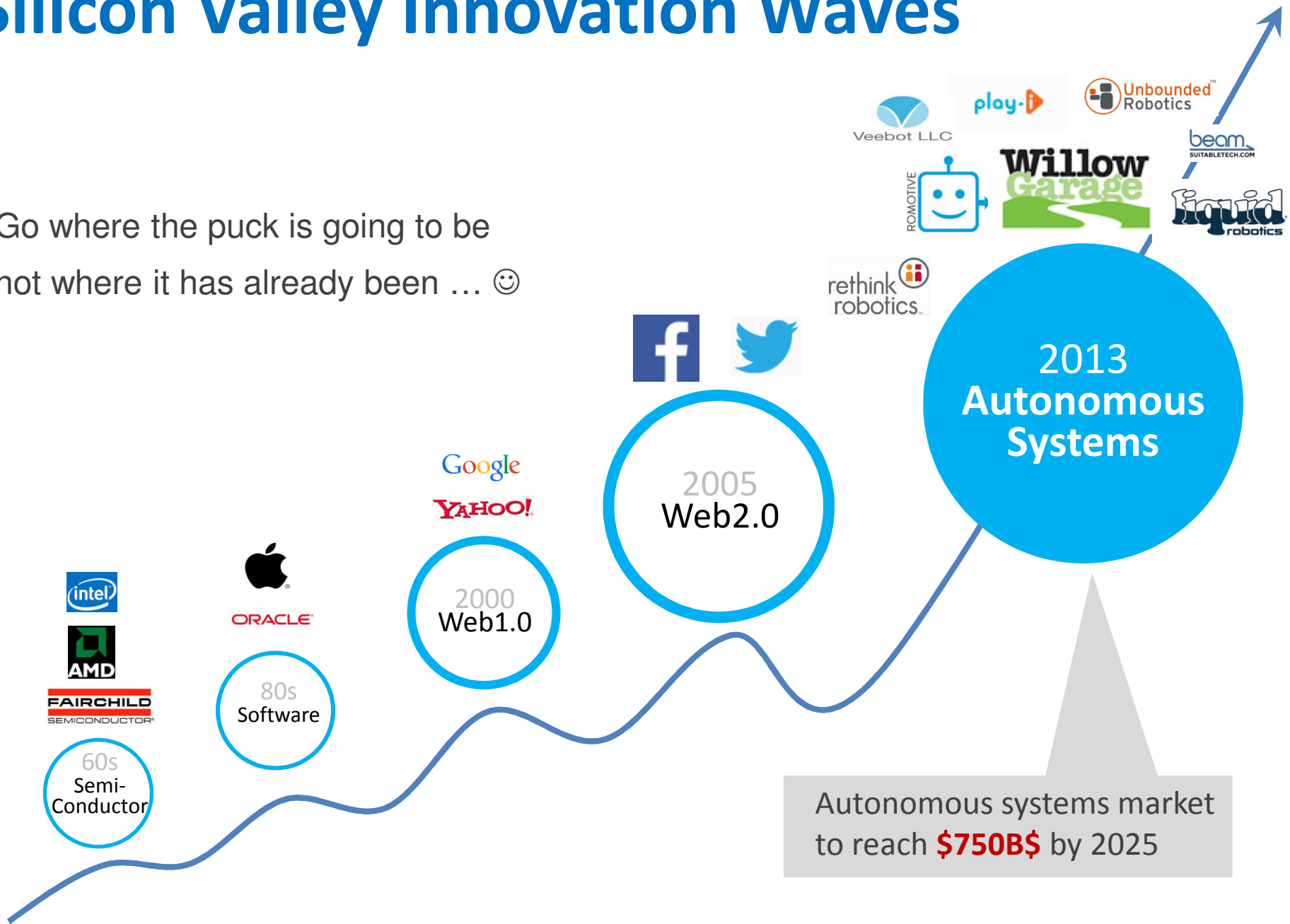


PSVL

Apple
Spaceship

Silicon Valley Innovation Waves

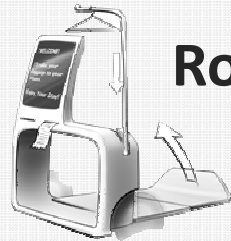
Go where the puck is going to be
not where it has already been ... 😊



Autonomous Systems Platforms

Common Technology Platform unifies three huge new markets

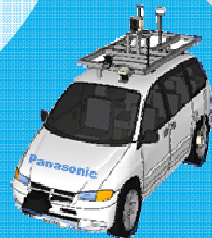
Autonomous Systems



Robots



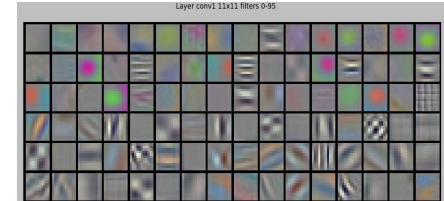
Drones



Autonomous Car

Key enabling technologies

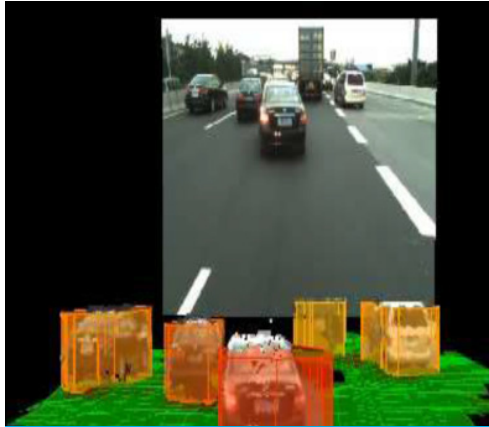
AI, broadly speaking



*AI - Deep Learning , 3D Environment Modelling , Object Detection and Path Planning

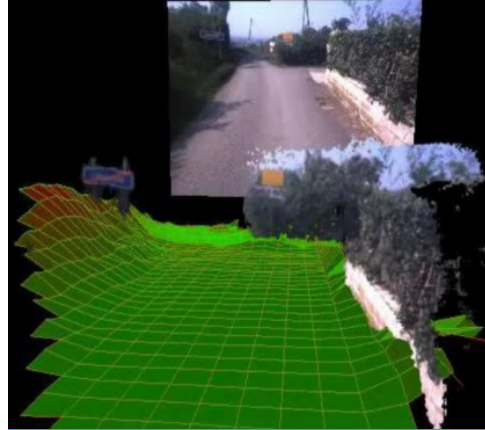
Autonomous Systems Technology Stack

Identifying Technology Opportunities for Autonomous Systems



Perception

Object Detection,
3D Environment Modelling,
Traffic Sign Recognition



Planning

Sensor Fusion, Path
planning, Trajectory
planning



Actuation

Breaking, throttle,
steering and motion
control



Deep Learning



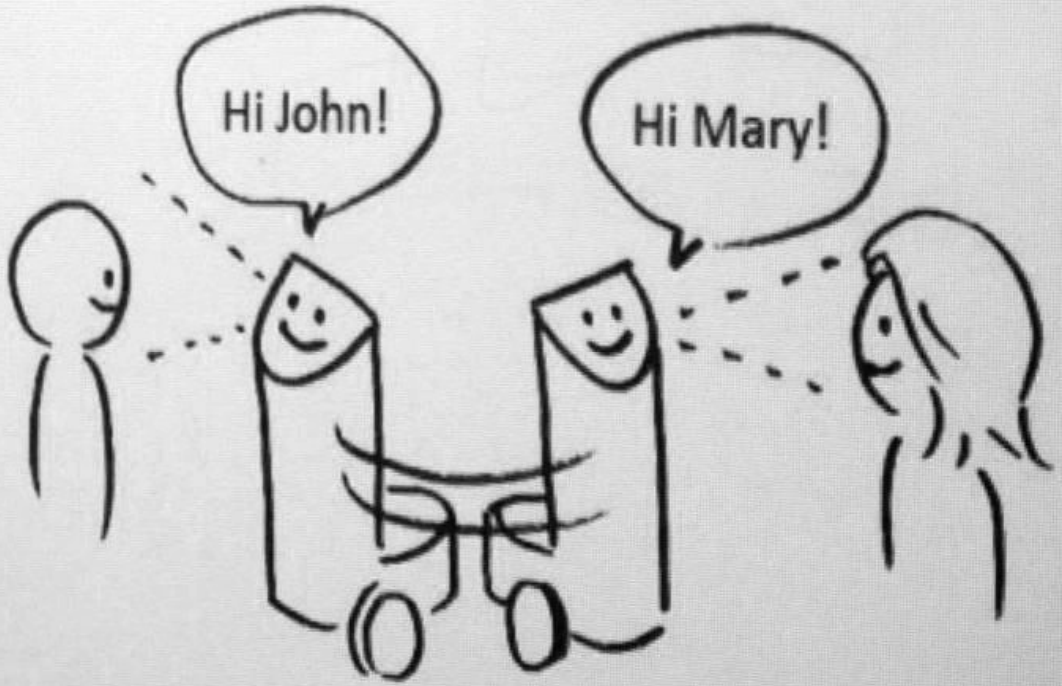
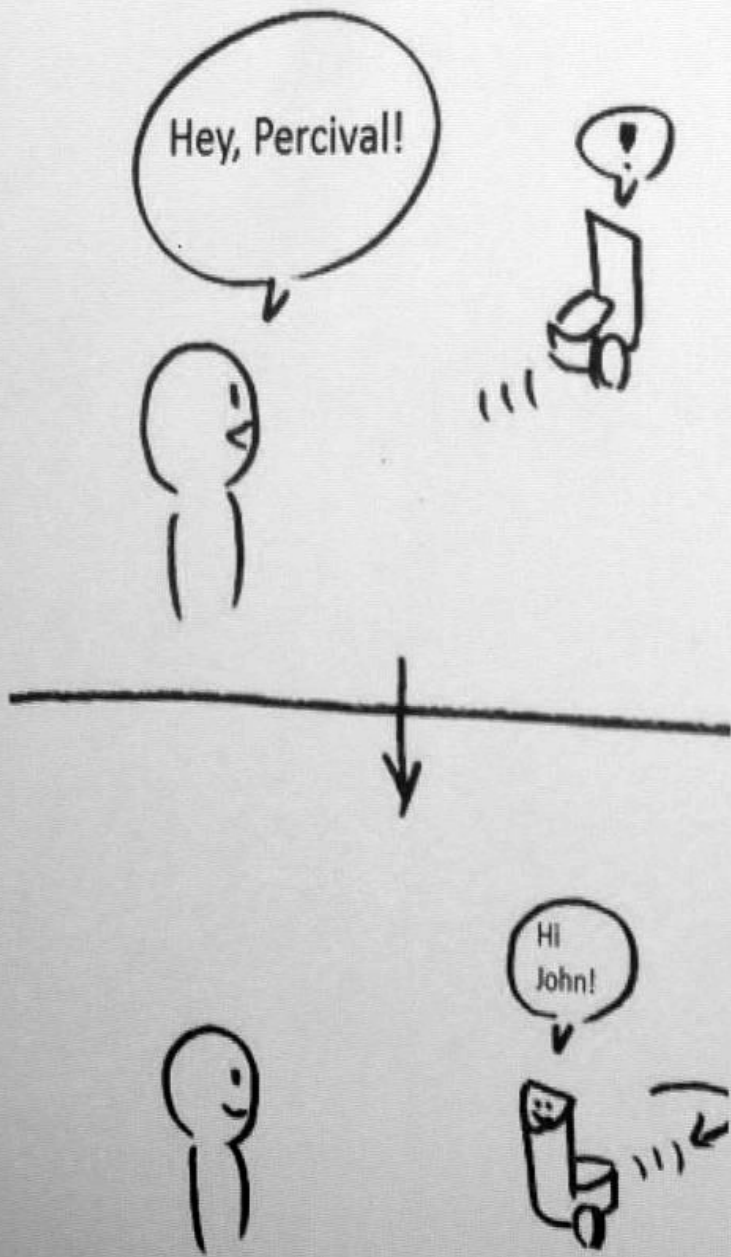
Applications on Learning in Planning
& Actuation





ROBOTHON

So we build stuff ...
Meet Percival



Face detection / Face identification

(Distance Estimation)



Visualization

Speech

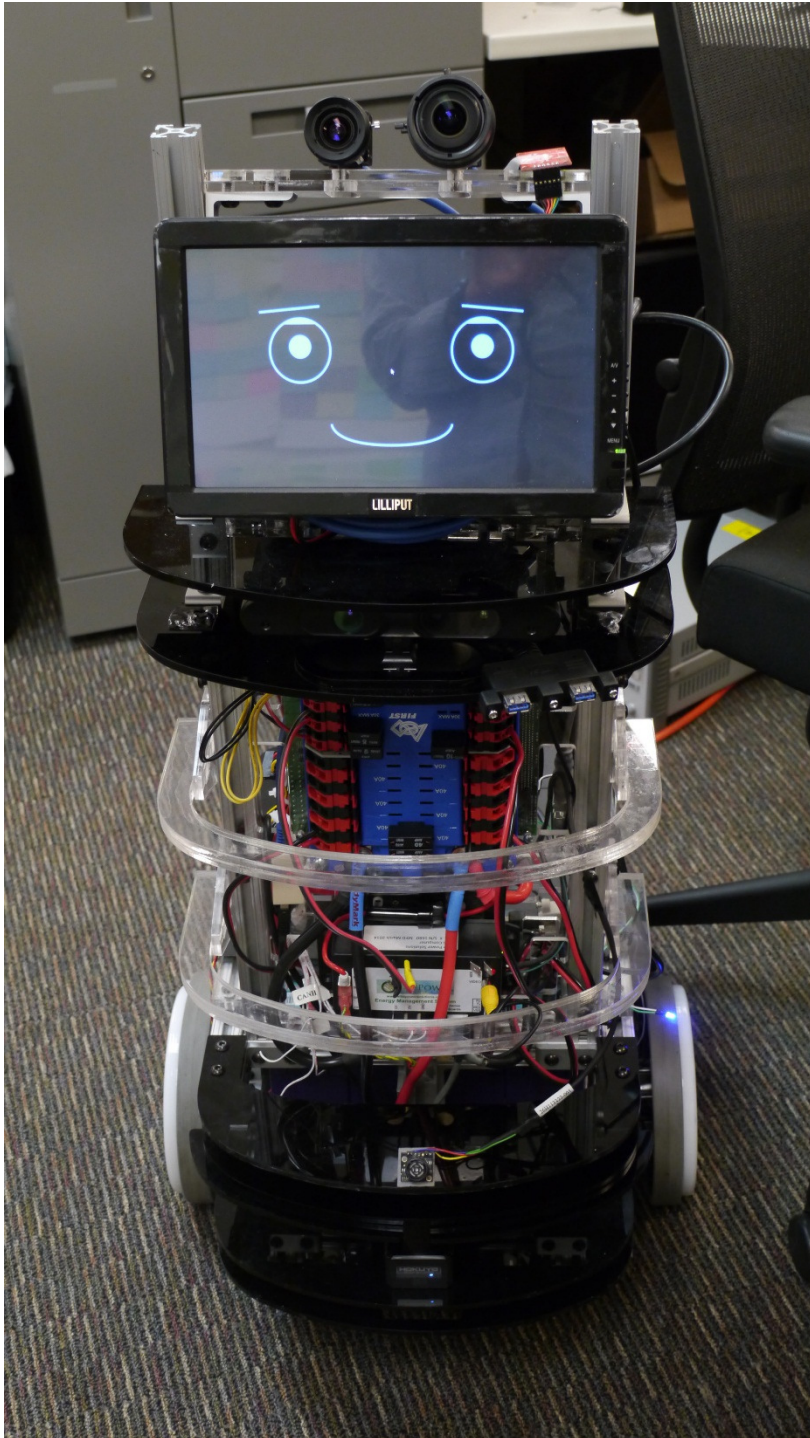
Sound

Convenience

Face Recognition & Depth Perception

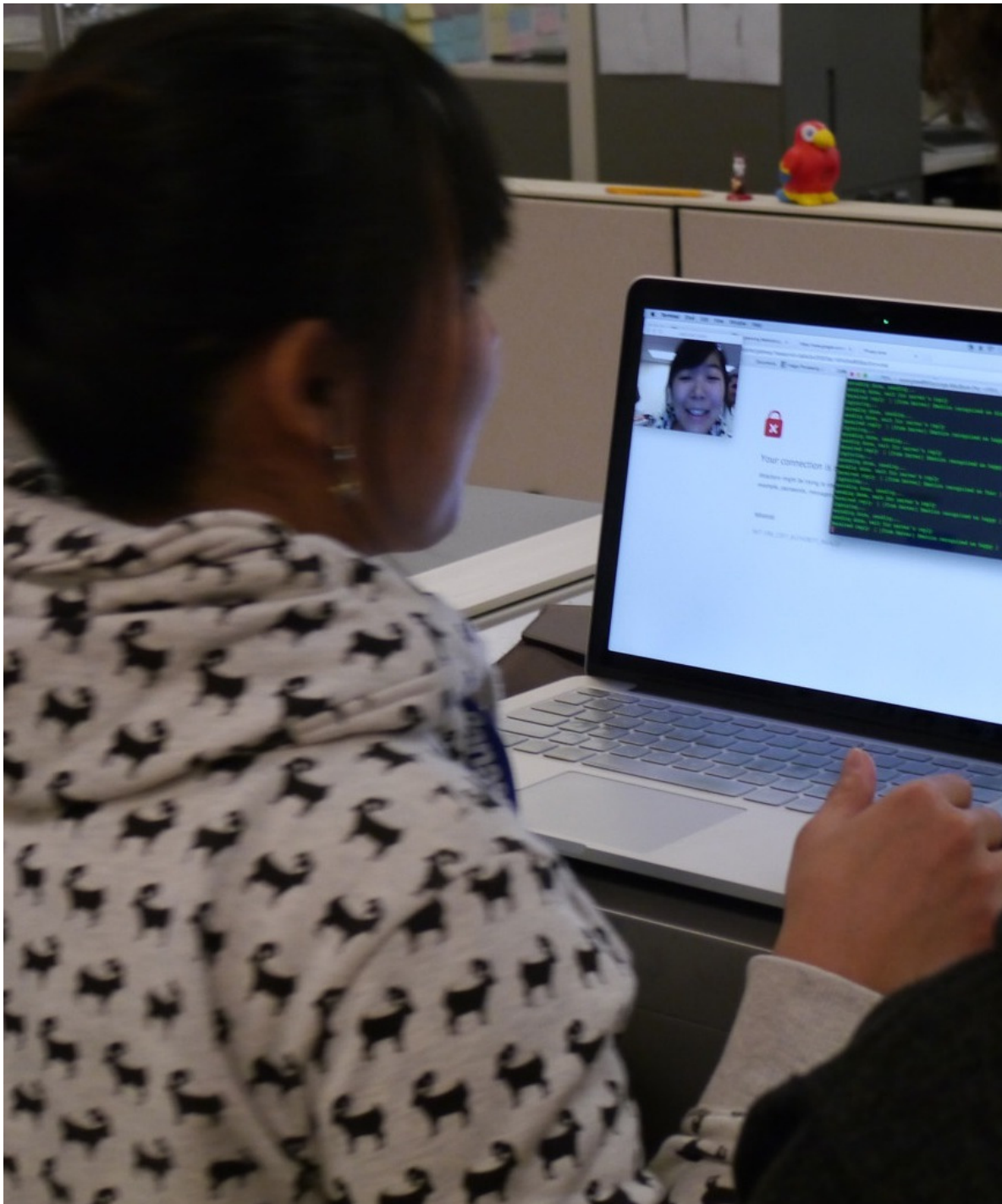
Touch

Emotion Recognition



- 'Hey percival' - Robot turn to face person when called.
- Souvenir photo - 'say cheese' interaction, saves photo.
- Integration





Emotional face recognition program that recognized when the person was angry, disgusted, scared, surprise, contempt sad or happy

DL@PSVL

A big thanks to LISA!

- You open source stance is helping making the world a better place 😊
- We are active users of Theano, Pylearn and Ground-Hog
- Also we are users of BV CAFFE ...

We work on

- Vision, with strong emphasis towards robotics and autonomous drive
- Language – specifically NLU (robotics too) and MT (Japanese is a focus here)

DCCK

Architecture, architecture, architecture

- Architecture very important for performance
- Usually architecture is optimized by hand
- Can we find a way to optimize architecture automatically?
- Also, why can't we vary architecture during training?

Optimizing Architecture

- It's hard, it's combinatorial
- But this is no excuse to just doing it manually
- And there are successful attempts in the past, also in other model typologies (HMM's)
- More recently teacher/student, dark-knowledge and FitNets approaches (related)

DCCK

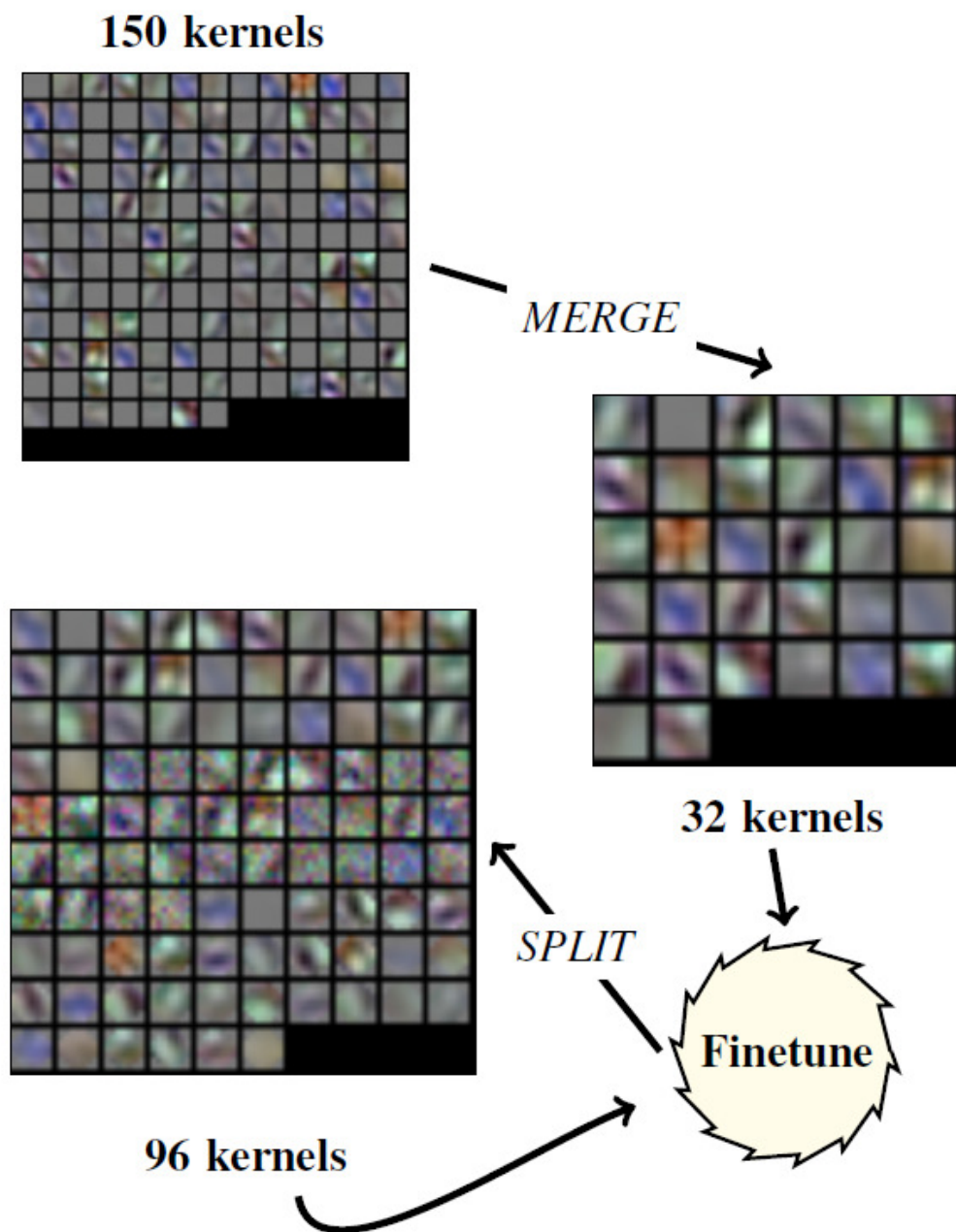
Deep Clustered Convolutional Kernels

Key point: vary architecture during training

1. Start from over-parameterized model

this has been shown to be easier to train but provides lots of redundant filters, excessively large model which possibly over-fit

2. Merge convolutional filters by clustering
3. SGD
4. Split convolutional filters and repeat



- Questions?
- How to split?
- How to merge (k-Means)?

Merging

Purpose: remove redundancy

1. Cluster filters layer by layer, starting from the lower layer
2. Use cluster of the lower layer to constrain clustering of the following layer
3. Repeat for every convolutional layer in the network *

* Potential issue was observed with the last convolutional layer, before the FC layer

Splitting

Purpose: increase parameters

Use filter transformations that are inherent to the problem we are trying to model

For instance, in our case we use rotation, jittering and additive Gaussian noise

Experiments

Start from our best systems and best architecture (close to SOA) and see if we can improve architecture & accuracy

Tested on three vision benchmarks: MNIST, GTSRB, CIFAR10

MNIST sanity check

- Train our to convergence
- Merge/split (only first layer) and compare with same size, shows good promise

No.	STAGE	CONV1	CONV2	ERR(%)
1	ORIGINAL	100	50	0.82
2	ORIGINAL	200	50	0.78
3	ORIGINAL	300	50	0.75
4	SPLIT FROM [1]	200	50	0.58
5	MERGE FROM [4]	100	50	0.59

GTSRB

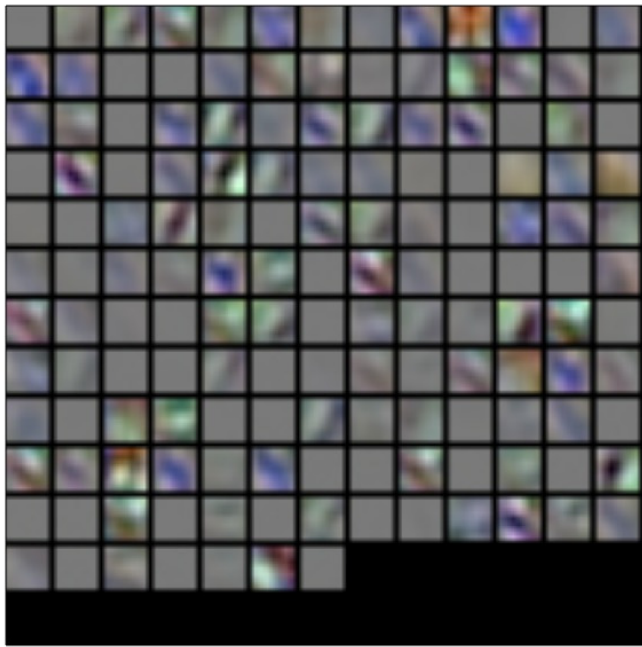
German traffic sign recognition benchmark

We have a SOA system based on 3-DNN's
(ensemble, similar to IDSIA system)

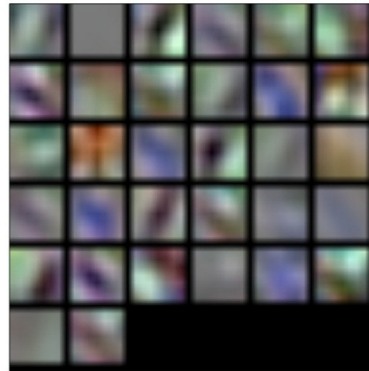
We also have a smaller / faster 1-DNN SOA
system

GTSRB - filters

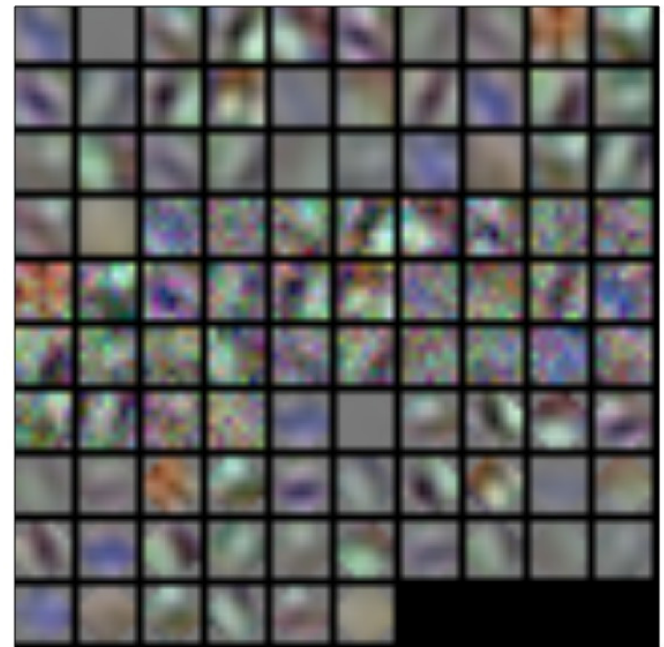
Conv1 initial: lots of redundancy



merge



Split: rotation/jitter & noise



GTSRB – 1-DNN

No.	STAGE	CONV1	CONV2	CONV3	ERR(%)
1	ORIGINAL	150	150	250	2.44
2	MERGE [1]	32	150	250	2.34
3	MERGE [2]	32	32	250	2.7
4	MERGE [2]	32	64	250	2.36
5	MERGE [3]	32	32	32	3.82
6	SPLIT [2]	64	150	250	2.5
7N	SPLIT [3]	32	64	250	2.25
8R	SPLIT [3]	32	64	250	2.15
9	SPLIT [1]	300	150	250	2.24
10	MERGE [1]	40	150	250	2.31
11	SPLIT [1]	150	300	250	2.27

* Same accuracy 2x or improved accuracy

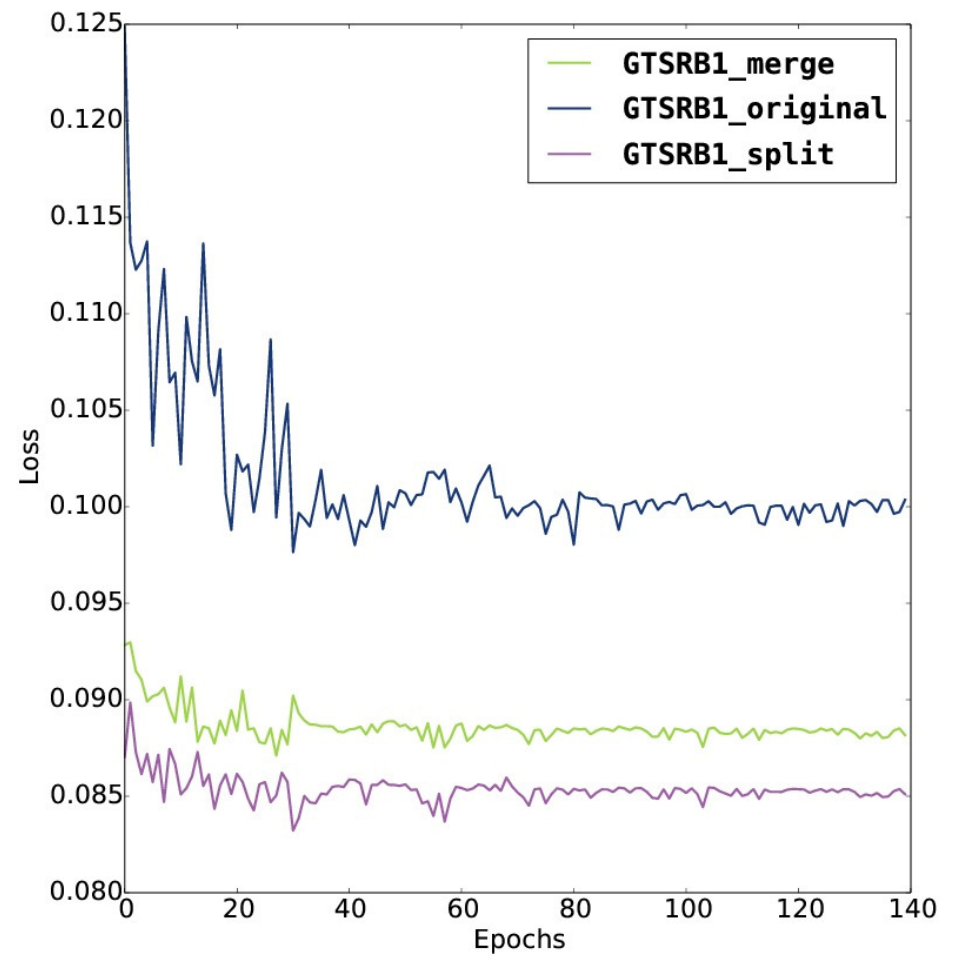
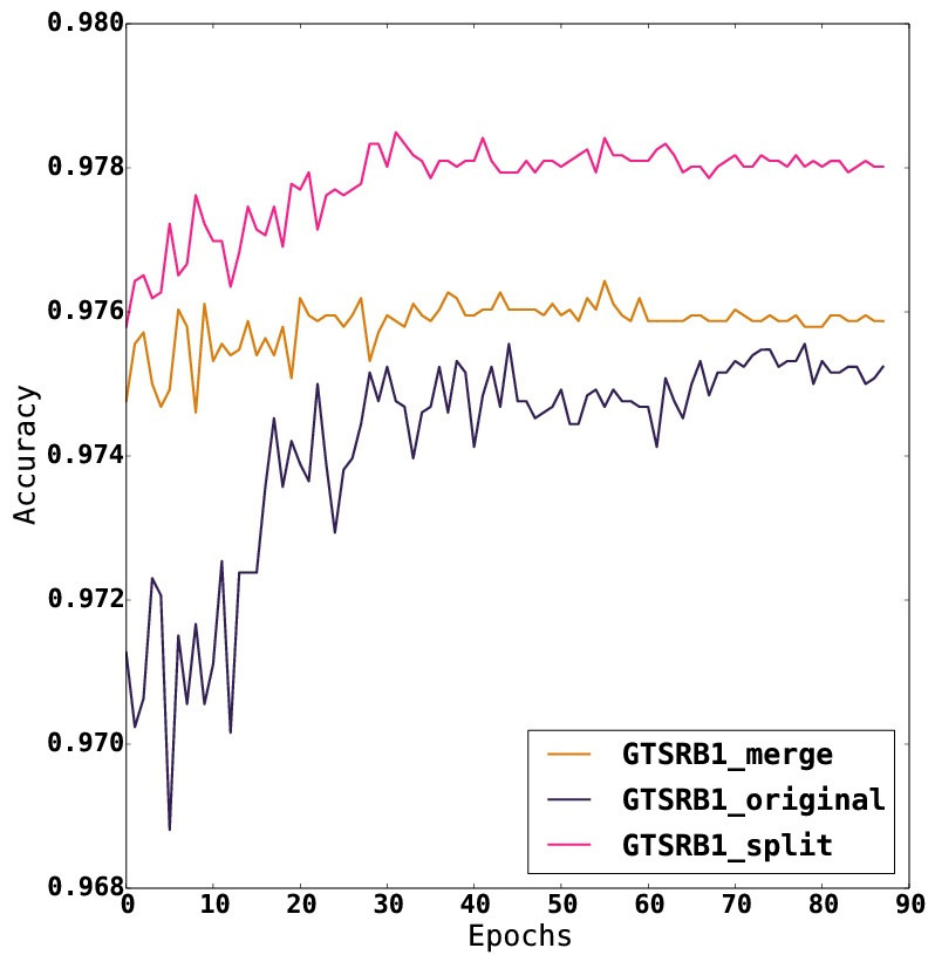
MODEL	STAGE	CONV1	CONV2	SPEED(MS)
1. SIMPLE	ORIGINAL	150	150	14.8
2. SIMPLE	MERGE [1]	32	150	14.1
3. SIMPLE	MERGE [2]	32	64	12.6
4. 3-DNNs	ORIGINAL	150	150	27.9
5. 3-DNNs	MERGE [4]	32	150	19.4

GTSRB – ensemble

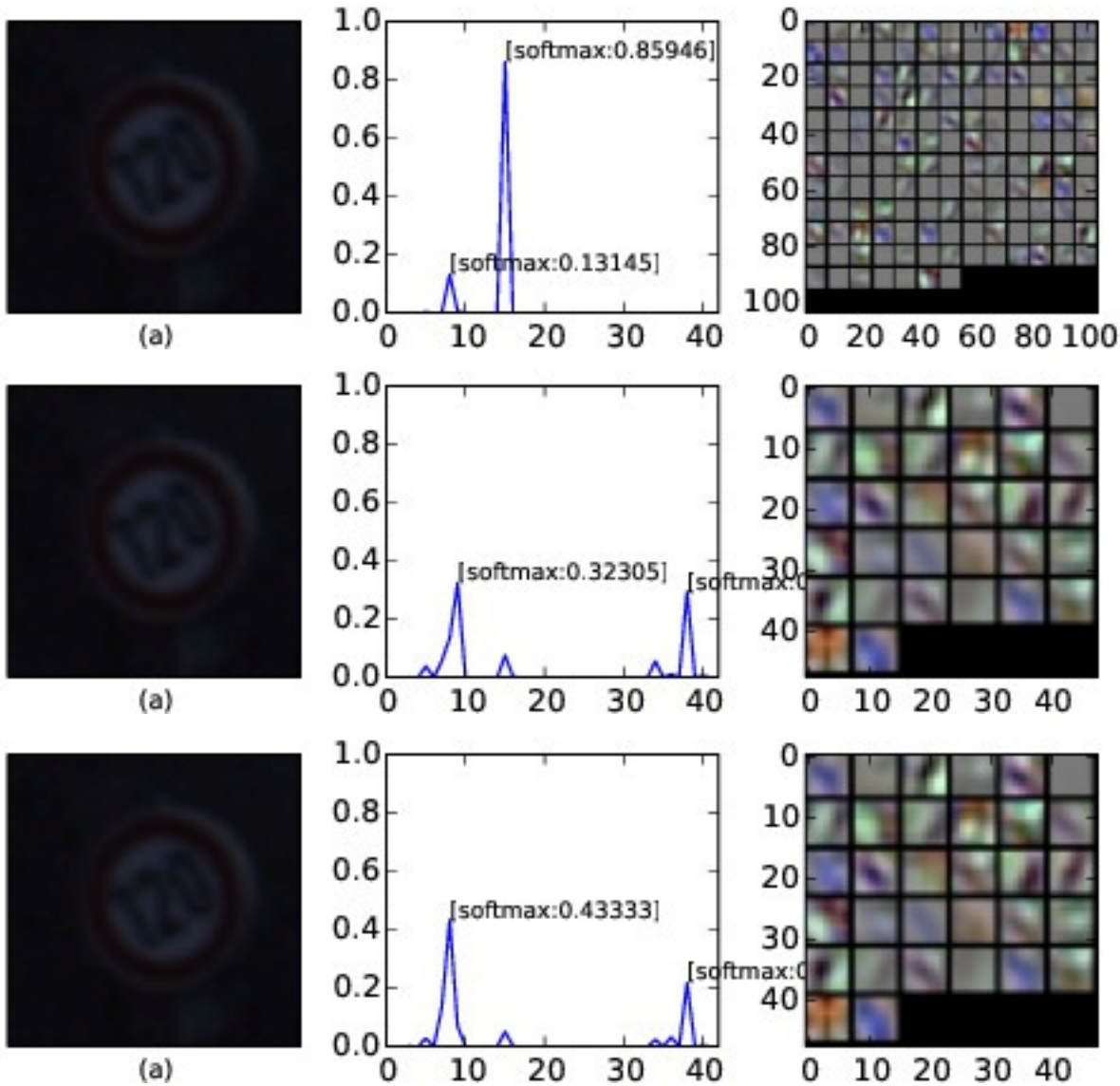
Smaller ensemble architecture with improved accuracy

No.	STAGE	CONV1	CONV2	CONV3	ERR(%)
1	ORIGINAL	150	150	250	1.24
2	ORIGINAL	16	150	250	1.67
3	MERGE [1]	32	150	250	1.18
4	MERGE [1]	16	150	250	1.25
5	SPLIT [1]	300	150	250	1.21
6	SPLIT [3]	64	150	250	1.15

GTSRB plots



GTSRB - outlier



* Initially mis-classified, after merging, jittering (rotation) and SGD is correctly classified

CIFAR-10

* Small improvement

No.	STAGE	CONV1	CONV2	CONV3	ERR(%)
1	ORIGINAL	192	192	192	10.4
2	SPLIT [1]	384	192	192	10.29
3	SPLIT [1]	576	192	192	10.25
4	MERGE [3]	192	192	192	10.2
5	SPLIT [1]	192	192	384	10.04
6	SPLIT [1]	192	384	192	10.04
7	MERGE [6]	192	192	192	10.28

Self critique

- Would be good to show we can converge to good architecture when starting from random architecture (or basic architecture)
- Should compare (or combine?) with teacher/student methods
- Scale up and test on large datasets

MT&NLU

Machine Translation

- RNN systems provide competitive performance while avoiding a huge chain of component (word segmentation, Tagging, reordering)
- Problematic:
 - RNN Systems require large models or/and limitation of lexicons are required
- Goal:
 - Study the impact of the tokenization in RNN for JP-EN

Machine Translation JP-EN

- Tokenization: An issue in Japanese
 - Lot of tools try to solve this issues
- Proposal:
 - Remove tokenization and lets the model learn it
- Benefit:
 - Avoid OOV words in Japanese?

Machine Translation

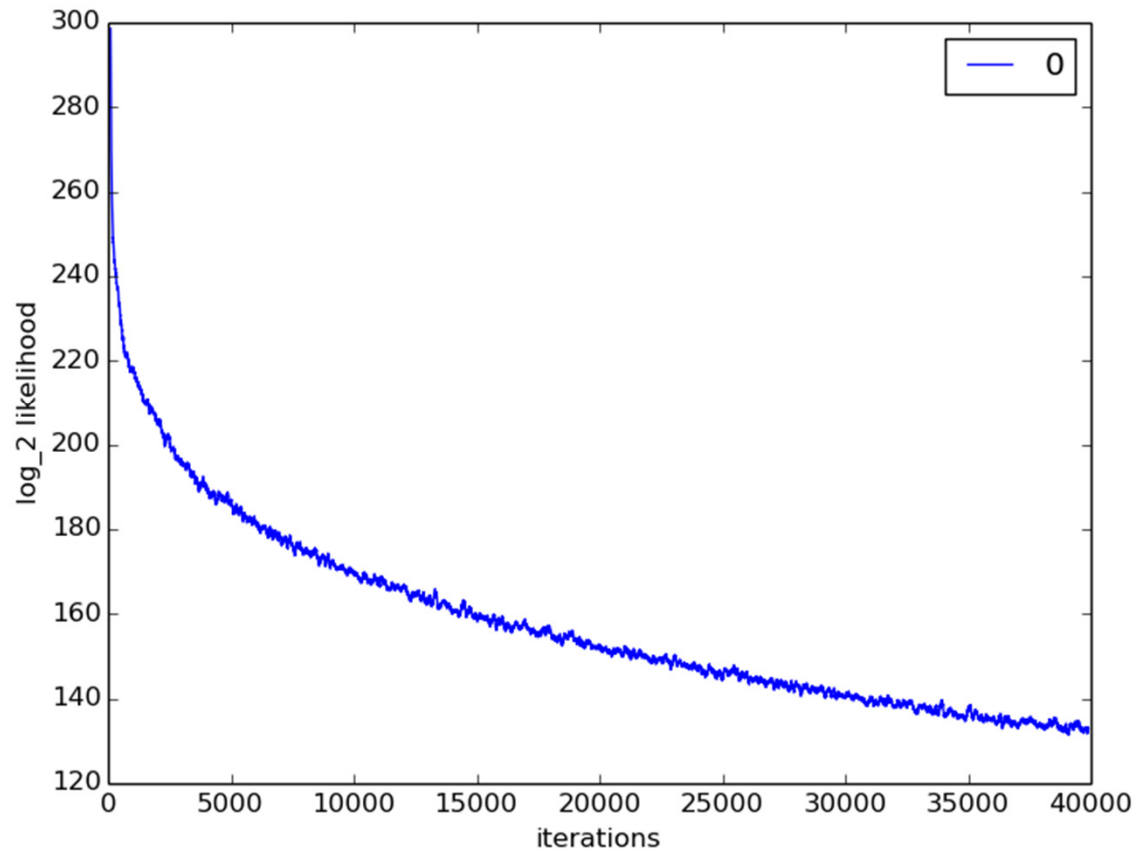
- Corpus ASPEC:
 - 3 millions of parallel sentences from abstract of scientific English-Japanese papers.
- ASPEC campaign:
 - Baseline systems (online system + Organizer baseline) are between ~ 11 and ~15 of BLEU
 - Best system without external data: 23.5 BLEU (Travatar-based Forest-to-string)

Machine Translation – 1st experiment

- Experiment:
 - 2*RNNs Decoding-Encoding
 - Lexicon Input (JP): 3333 words (no tok.)
 - Lexicon Output(EN): 40000 words
 - Embedding vector dimension: 1000
 - Titan Black:
 - Memory usage: between 1.6 Gb and 4.5 Gb

Machine Translation ASPEC JP-EN

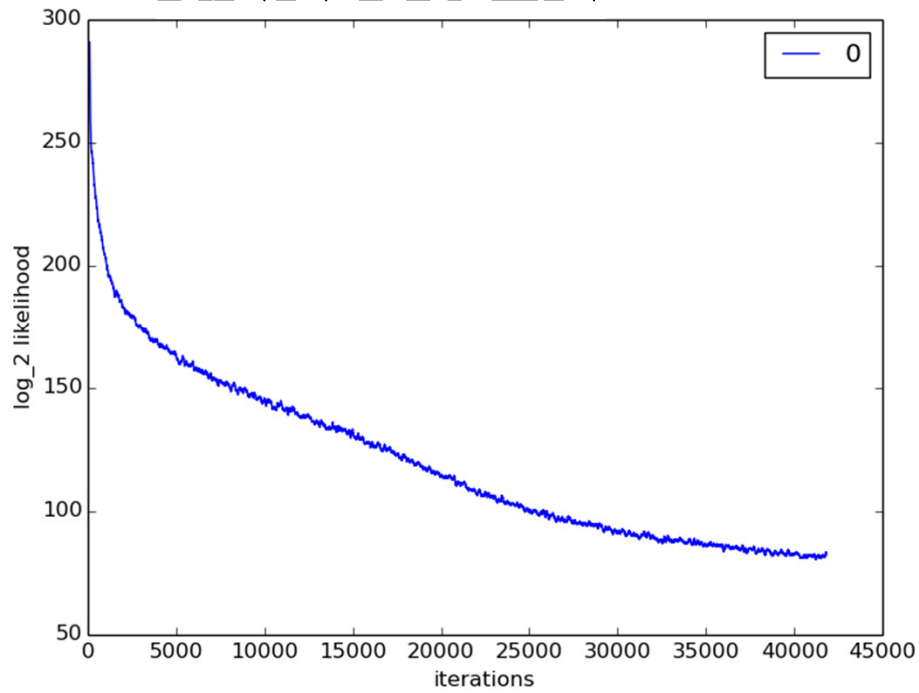
- Perplexity:



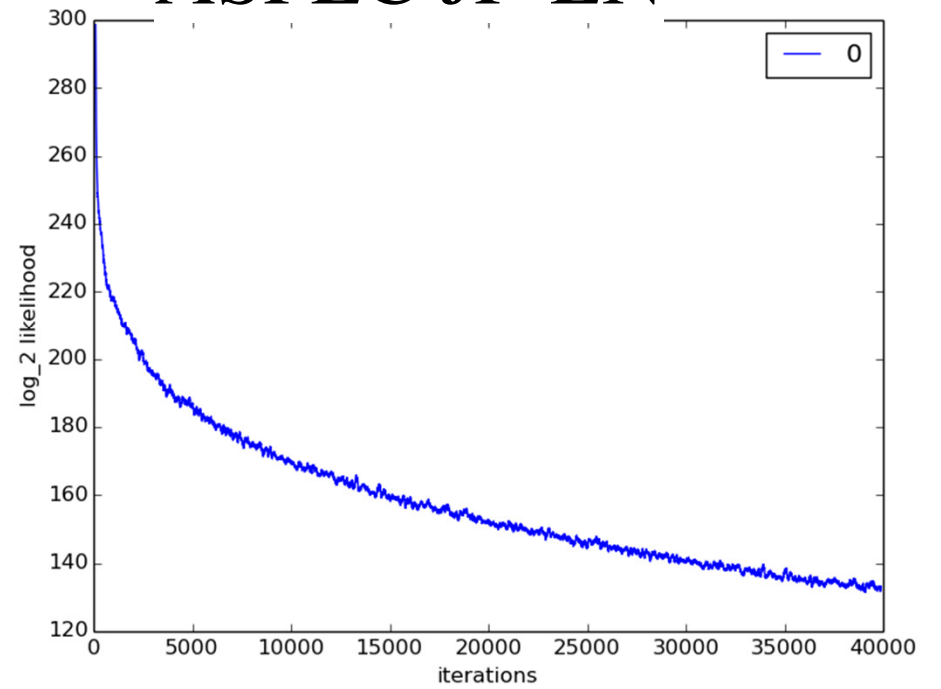
- Slower convergence than FR-EN ... but still works!

Machine Translation JP-EN/FR-EN

RNN FR-EN



ASPEC JP-EN



Machine Translation JP-EN/FR-EN

• Some examples:

S: 本論文では,多成分非理想係気液平衡問題の新しい計算法を提案する。

T: This paper presents a new computation procedure for multicomponent nonideal vapor - liquid equilibrium problems.

O: This paper proposes a new calculation method for the problem solving the problem of the boundary condition.

S: 最小検出歪みは 10^{-5} であった。

T: As a result , it was found that its minimum detection strain was 10^{-5} .

O: The minimum detectable strain was found to be 10^{-5} .

S: 多層薄膜による各種のセンサを実現しているとした。

T: Various sensors have been realized with multilayer thin films .

O: Various sensors by multi - layer thin films have been realized .

S: 1995年3月,マリアナ海溝最深部10,911mに着底し,初期の性能を実証した。

T: In March , 1995 , it was successful in landing on the deepest bottom of 10911m of the Mariana Trench , and its expected performance was proved .

O: In March 1995 , the UNK Trough was exposed to the depth of 10 , UNK and the initial performance was verified .

Machine Translation JP-EN/FR-EN

- Some issues with acronyms and rare example:
 - 10,911m -> 10,UNK
 - Chemical formula:
- But easily recoverable
- Some sentences are rotated:
 - Grammatical structure of Japanese is inversed with the English
 - More frequent for long sentences

S: ... けるSm2Fe17N3磁 ...
T: ... of Sm2Fe17N3 magnets ...
O: ... of UNK UNK magnets ...

Machine Translation – 1st experiment

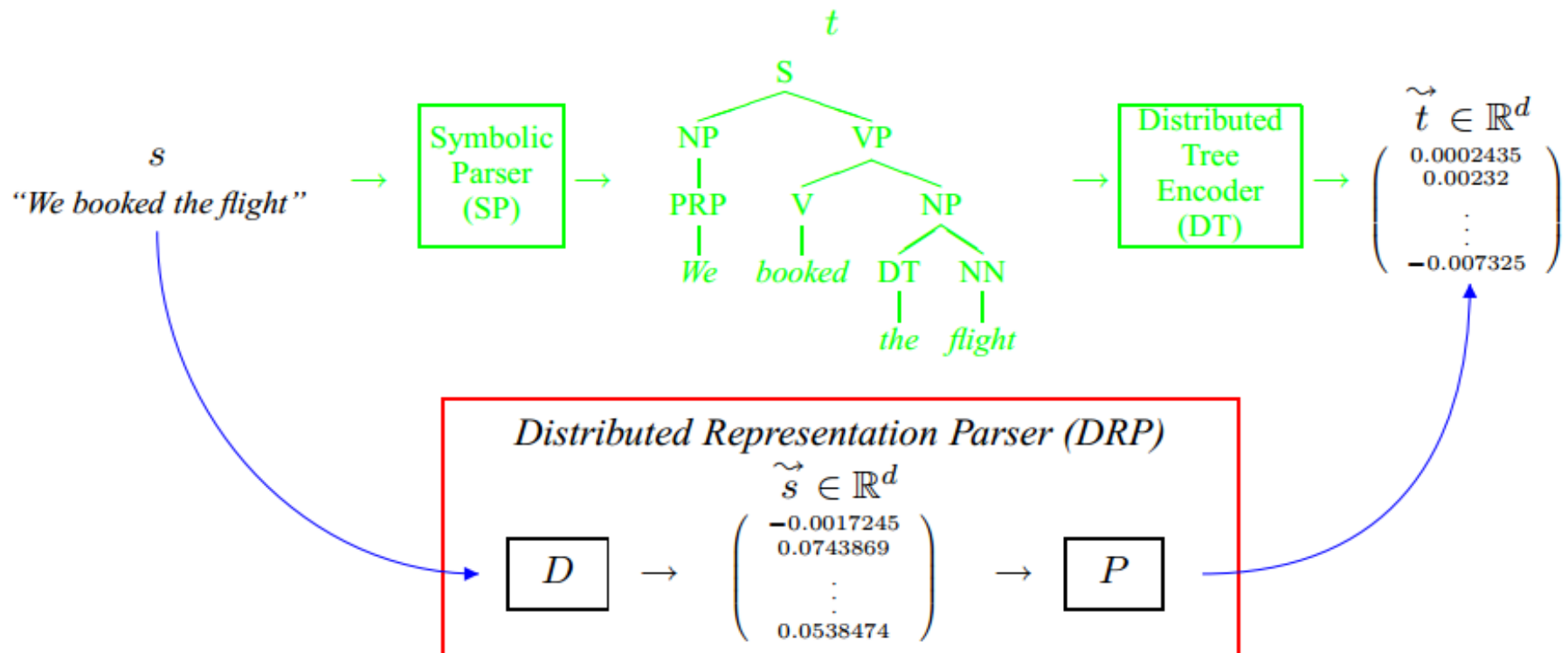
- Results:
 - 17 of BLEU on “raw output” without post-processing (just 12 best Beam search decoding)
 - Better than the baseline, maybe + 4BLEU with post processing
- Actual work:
 - Confusion Network decoding with RNNLM
 - Acronym or rare examples recovery
- Parallel model for OOV?
- Combination of Tree based method? Distributed Tree Kernel?

Distributed Tree Kernel

- Goal:
 - Reduce the time and the complexity of tree kernel
- from University of Rome, Prof. Zanzotto (ICML'13)
- Applications:
 - Question classification, Textual entailment....

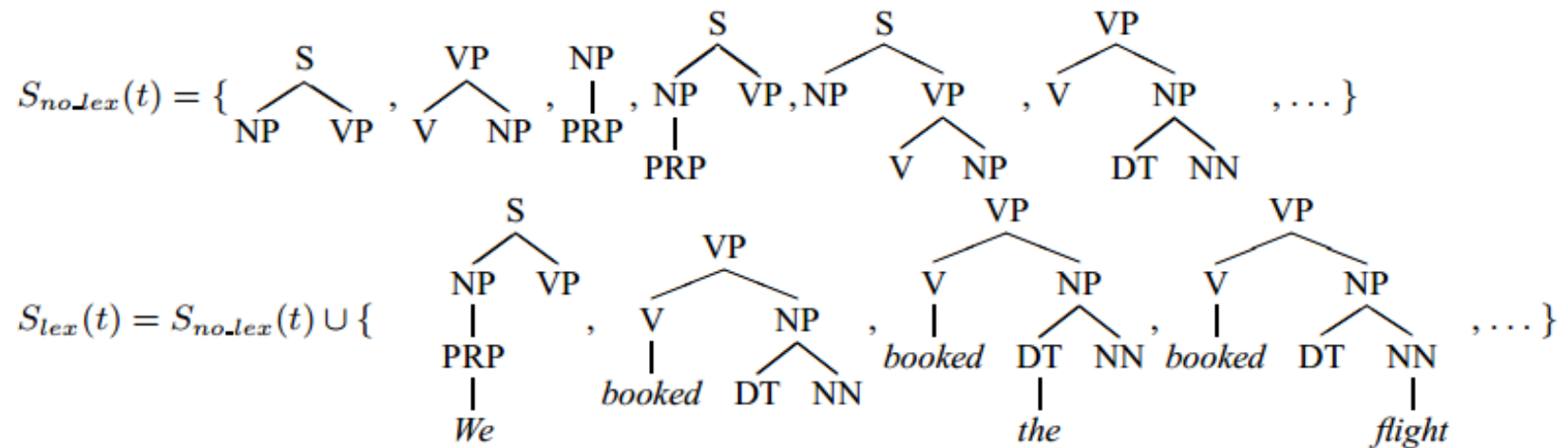
Distributed Tree Kernel

- Encoding of the morphosyntactic structures:



Distributed Tree Kernel

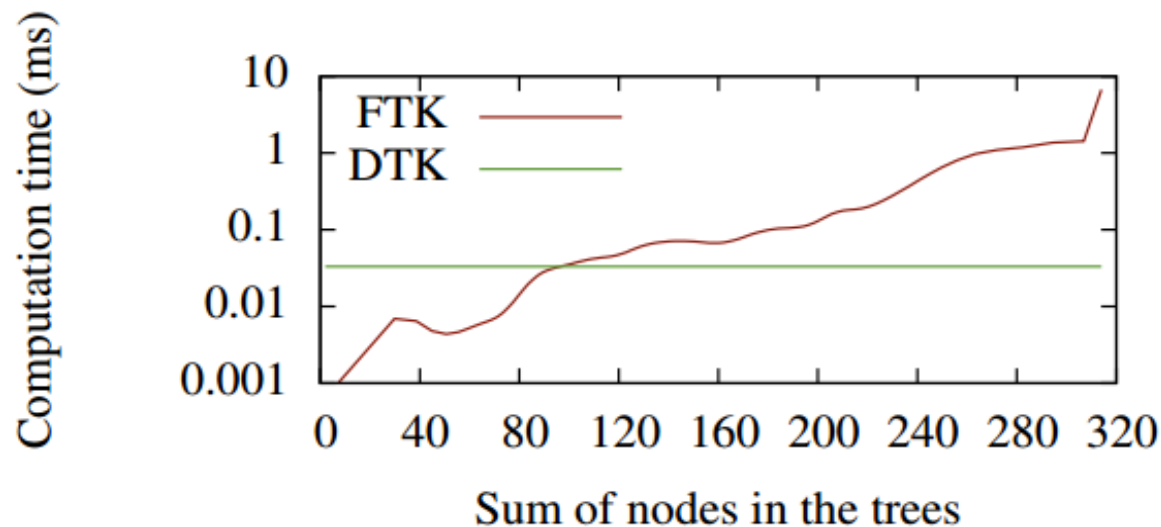
- Encoding of the morphosyntactic structures:



- With or without lex

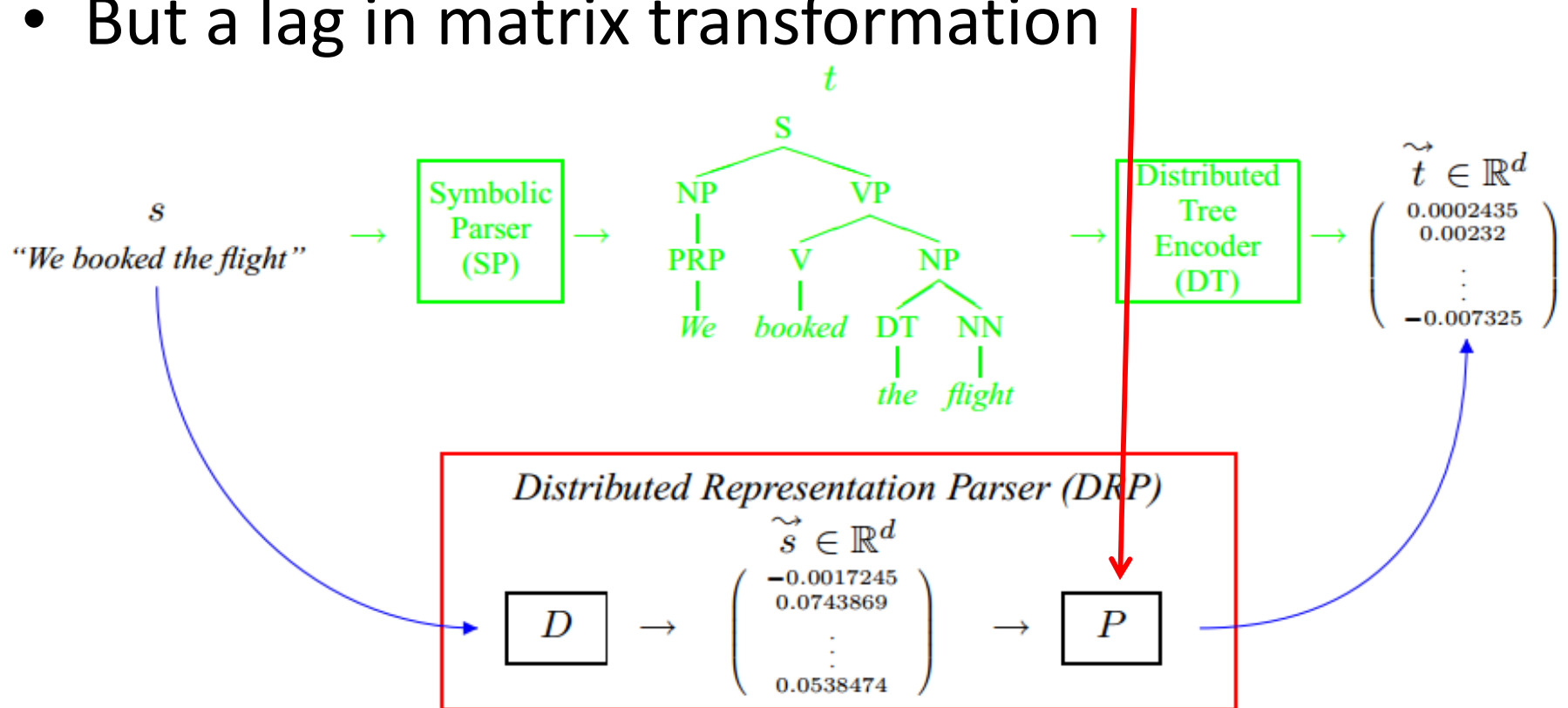
Distributed Tree Kernel

- Vector dimensions of s :
 - 4096 or 8192 with an average similarity up to 90 compare to state-of-the-art method
- Computation efficient:



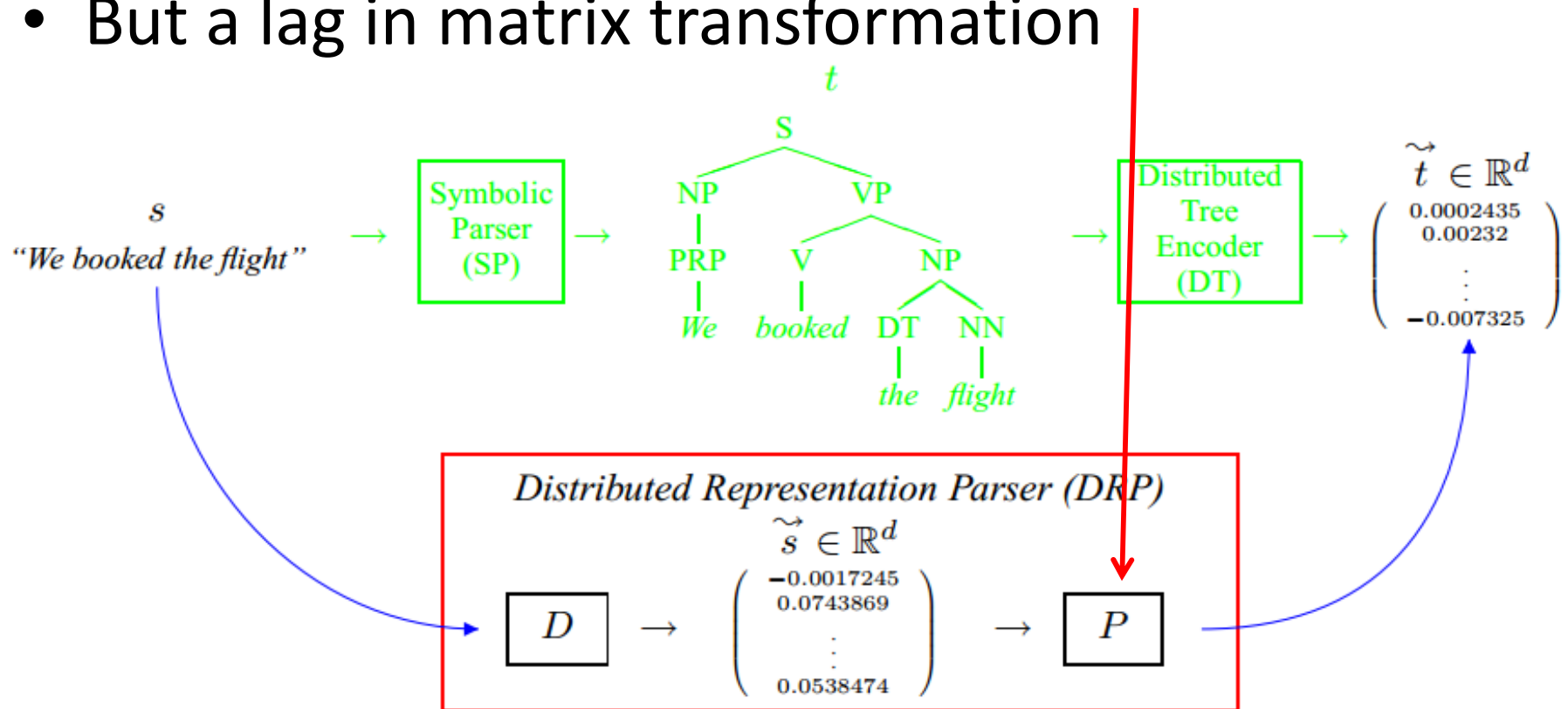
Distributed Tree Kernel

- But a lag in matrix transformation



Distributed Tree Kernel

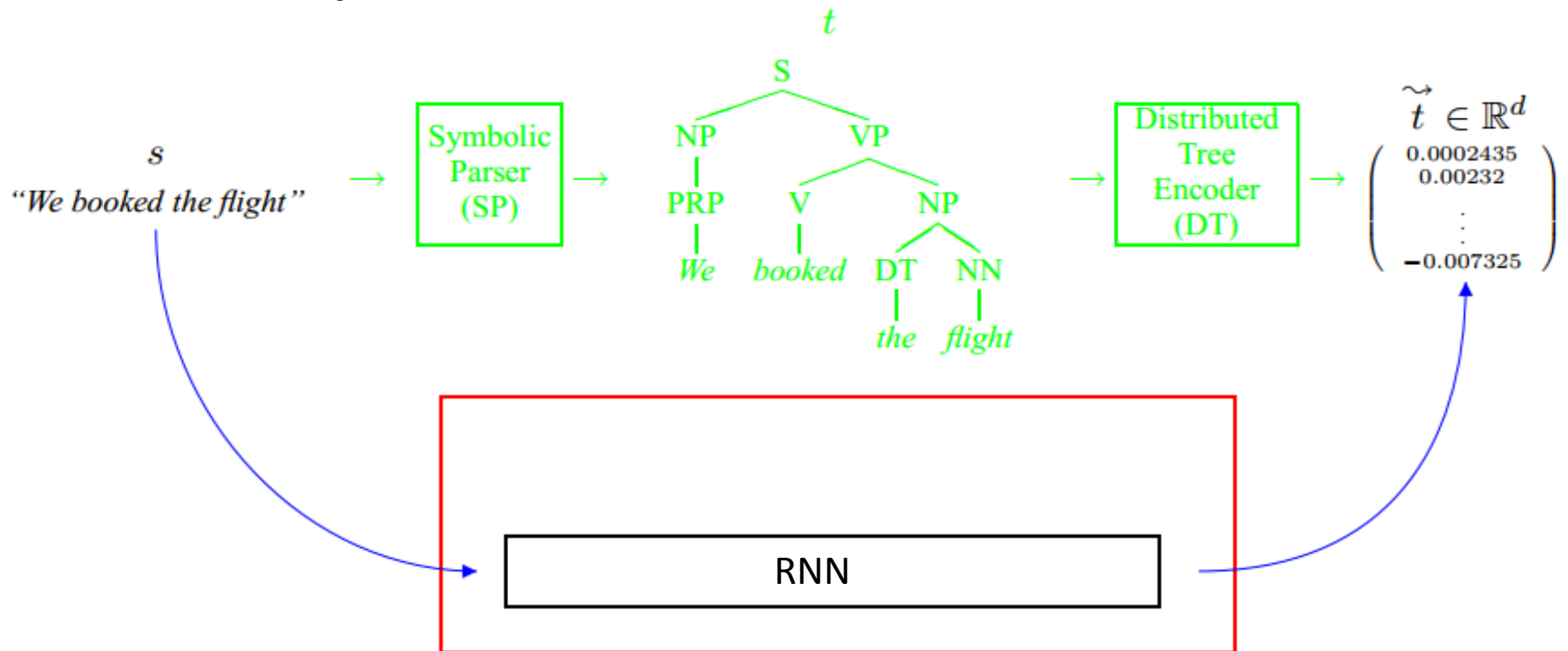
- But a lag in matrix transformation



- Replace P by a Encoder-Decoder

Distributed Tree Kernel

- Next step:



- Distributed RNN?