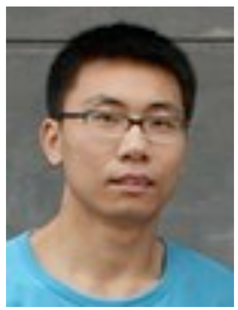
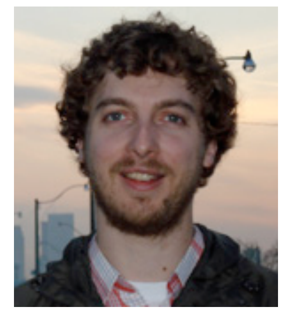
# Segmentation and structured deep learning






Mingyuan Jiu
Doct., soutenu
le 3.4.2014

Natalia Neverova
Doct. 2ième année

Graham W. Taylor,
Université de Guelph, Canada

Christian Wolf
Université de Lyon, INSA-Lyon
LIRIS UMR CNRS 5205

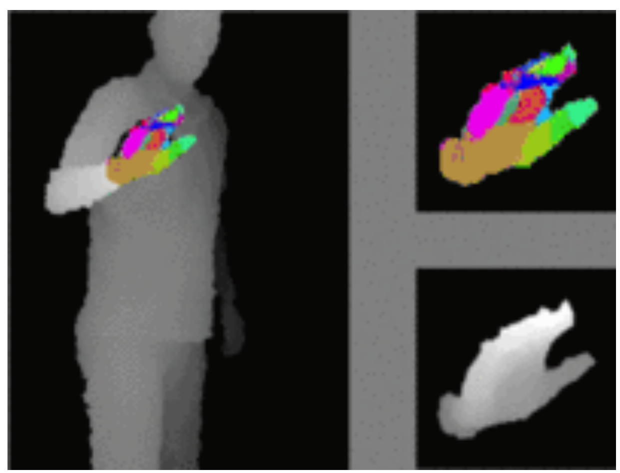
# Segmentation for visual recognition

## Applications:

- Pose estimation (body, hand)
- Semantic full scene labelling

Hard complexity constraints (real time!)

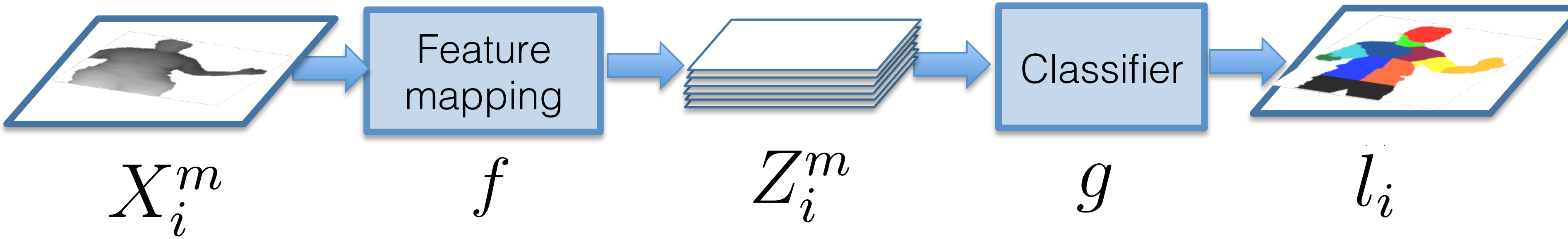
PhD of Mingyuan Jiu

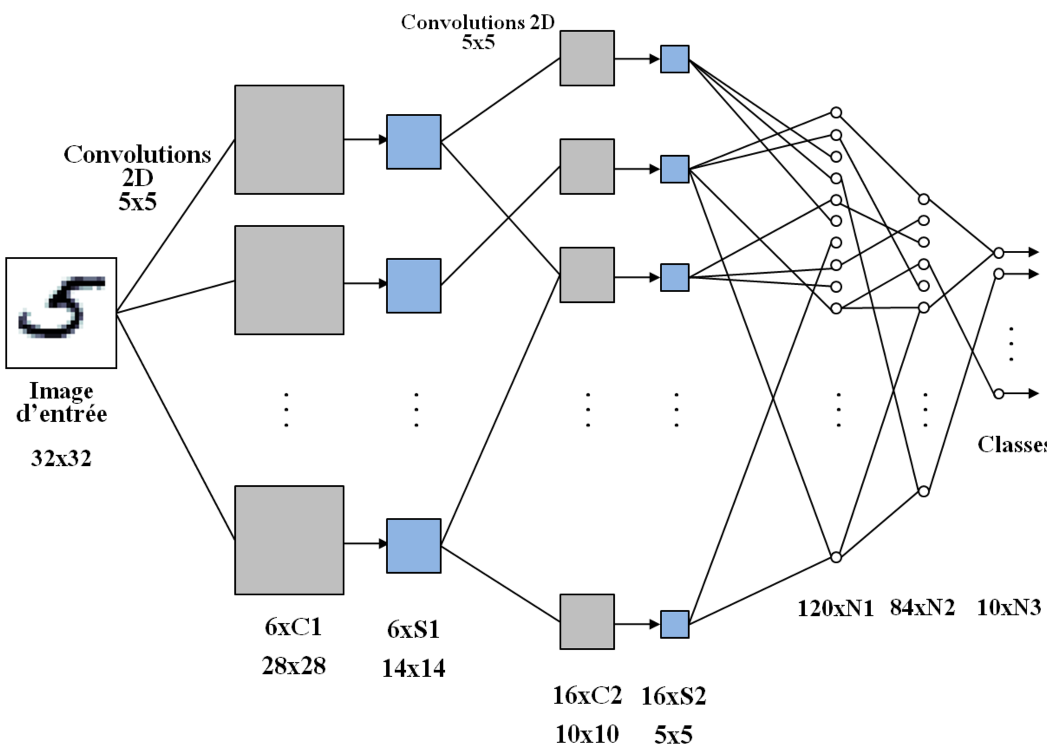


PhD of Natalia Neverova


PhD of Prisca Bonnet

# (Deep) representation learning

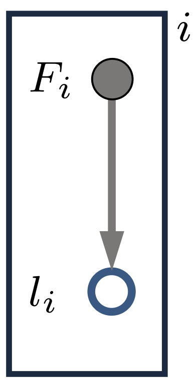


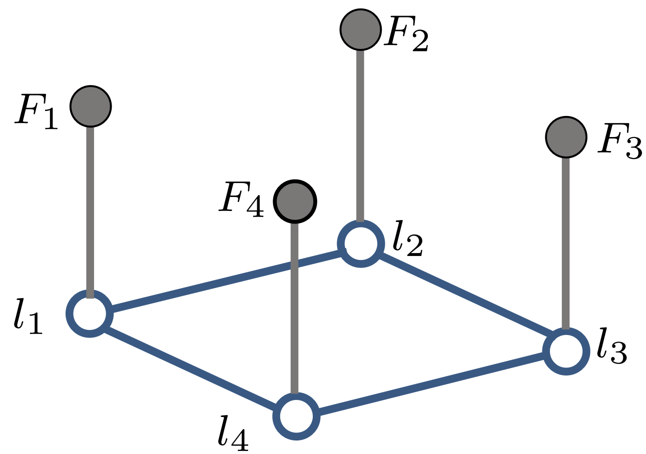



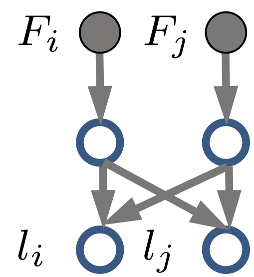
# Segmentation and spatial relationships

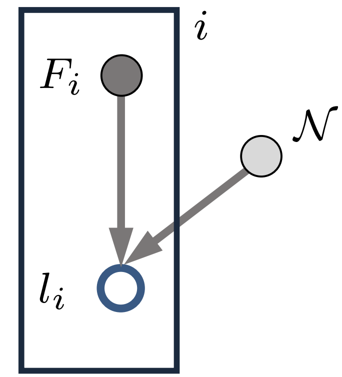
Features

Labels




**Non**

Pixelwise classification (independant)

[ICPR 2002a]

5e/43 à DIBCO 2009

**Oui**

MRF/CRF/BN. Inference of a global solution with high computational complexity

[IEEE-Tr-PAMI 2010]

[Neurocomputing 2010]

[ICPR 2010] [EG-W-3DOR 2008]

[ICPR 2002b] [ICPR 2008]

ANR Canada ANR Madras

**Oui**

Auto-context models

[Travaux en cours T. Kekec]

[Travaux en cours R. Khan]

Labex IMU-Rivière

ANR Solstice

**Oui**

Pixelwise classification. The prior improves the classifier

[Pattern recognition letters 2014]

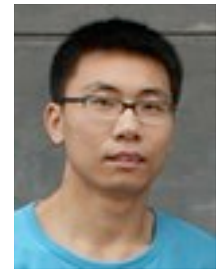
[Travaux en cours N. Neverova]

INTERABOT

# “Spatial learning”

## Application:

- Calculate human pose : set of joint positions
- Use an intermediate representation : body part segmentation



PhD of Mingyuan Jiu

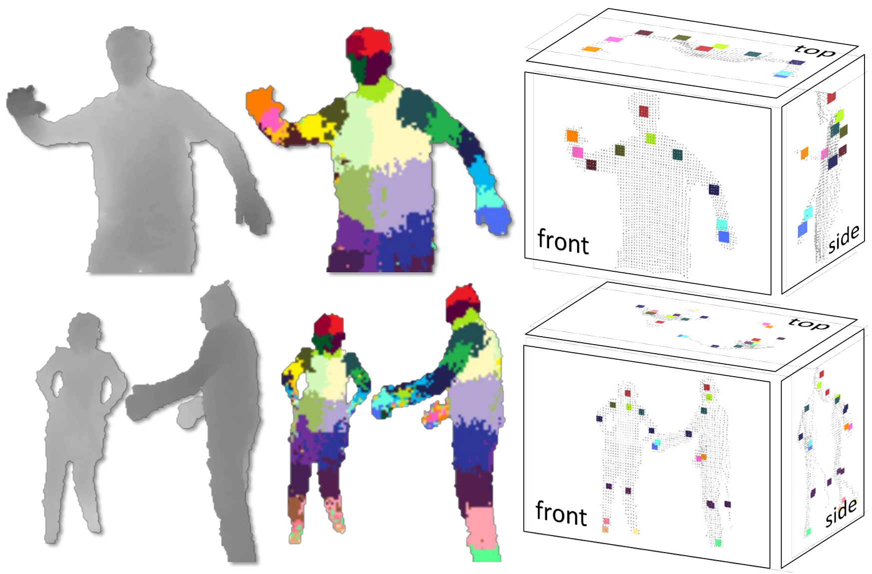
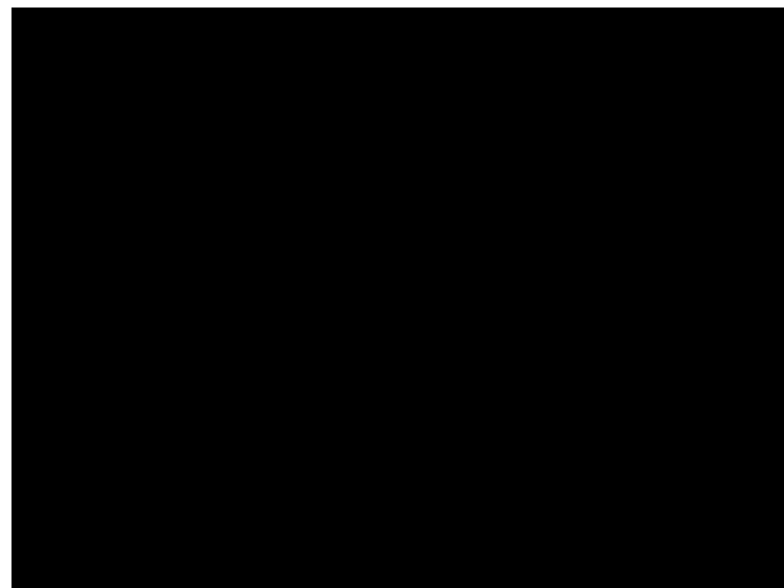



Figure : Shotton et al., CVPR 2011

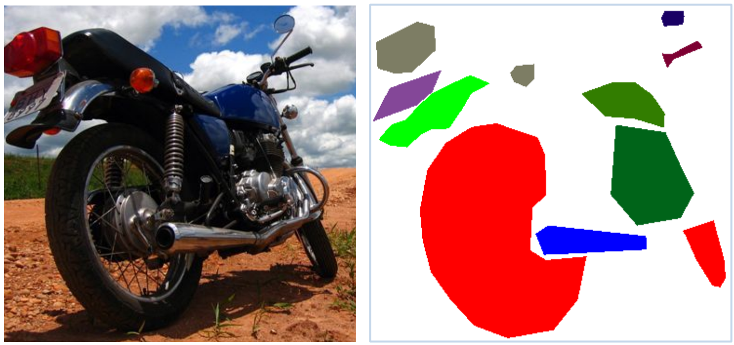


Jiu, Wolf, Baskurt, 2013

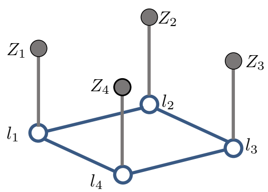
# Spatial relationships: labels

Additional information: neighboring pixels are likely

- to have similar labels, or
- to have labels which are adjacent in the object layout (!!)

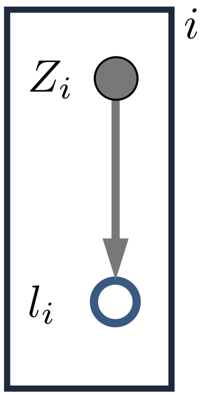
Could also be solved by MRF + discrete optimization




$$E(l_1, \ldots, l_N) = \sum_i U(l_i, Z_i) + \alpha \sum_{(i,j)\in\mathcal{E}} D(l_i, l_j)$$

# Structured models ... w/o structure

- It is not possible to include pairwise terms into a classifier which classifies pixels independently.
- Pairwise terms lead to combinatorial problems.
- Alternative strategy:
  - do not proceed by pairs
  - change the loss function for pixelwise classification
  - punish errors (classically), but:
  - punish errors less, if the misclassified label is a neighbor of the groundtruth label
- It will be shown that this strategy decreases “pure” classical (!!) classification error.



# Spatial deep learning

M images $\{X^1, \ldots, X^M\}$

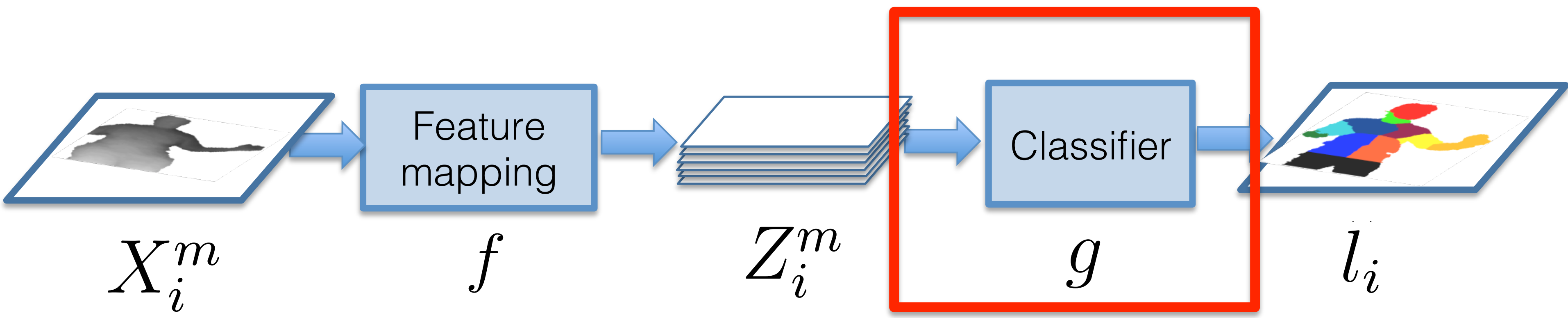
- A parametric function maps pixels *i* (and their receptive fields) to a feature representation

$$Z_i^m \in \mathbb{R}^Q$$

$$Z_i^m = f(X_i^m | \theta_f)$$

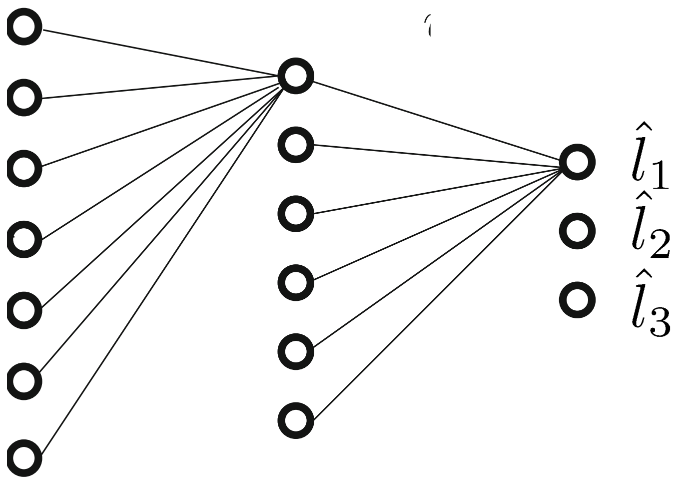
- A classifier predicts part labels

$$\hat{l}_i = g(Z_i^m | \theta_g)$$



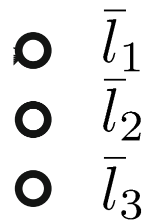
# Classical supervised learning

Stimulated network output:




$$\hat{l}_i = g(Z_i^m|\theta_g)$$

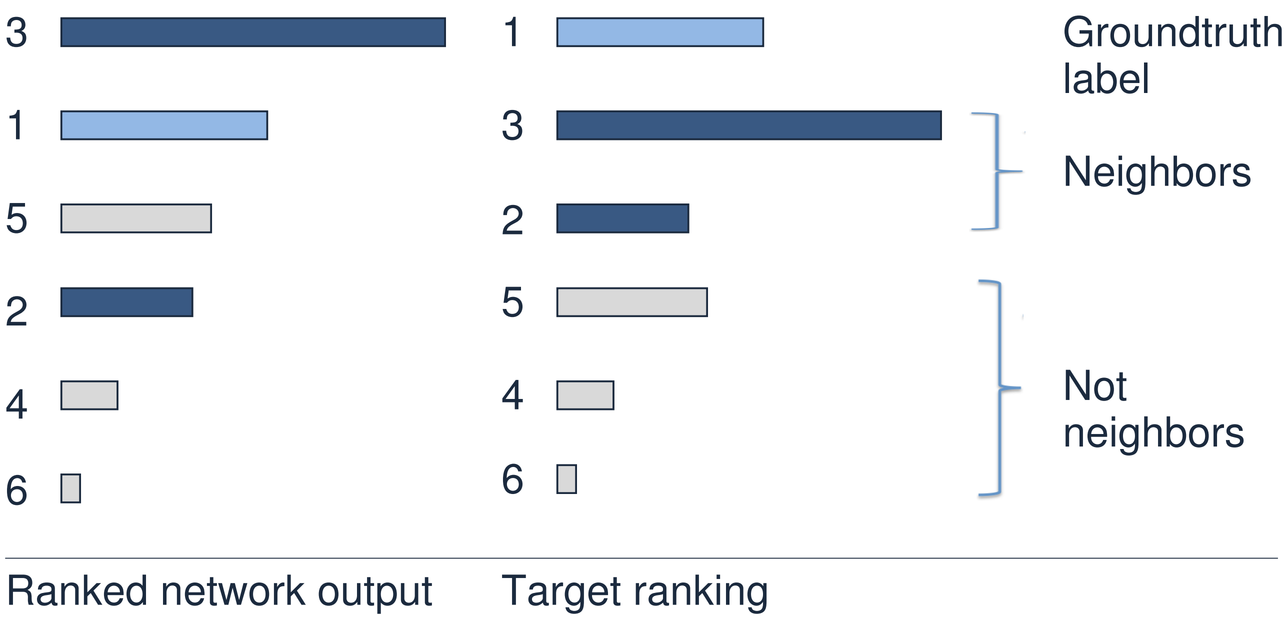
Target output (groundtruth):




Classical loss function: cross entropy

$$E(w) = -\sum_n \left\{ \bar{l}_n \ln \hat{l}_n \; + \; (1 - \bar{l}_n) \ln(1 - \hat{l}_n \right\}$$

# Learning to rank class labels





- The groundtruth class label is supposed to be ranked first (highest classifier response)
- The neighboring class labels are supposed to ranked next
- The non-neighboring class labels are ranked last
- The rankings inside the groups (gt, nb, non-nb) are irrelevant

# Learning to rank class labels

Similar to (Burges, NIPS 2006), the loss function is decomposed into terms over pairs. For each pair, differences in network output are mapped to probabilities :

$$o_{uv} = g(Z_{i,u}) - g(Z_{i,v})$$

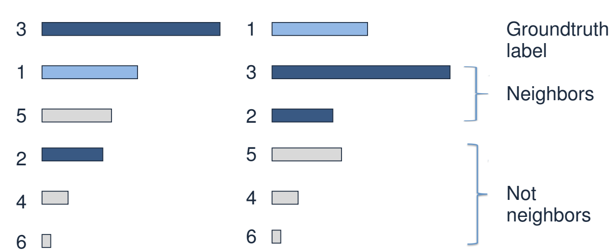
$$P_{uv} = \frac{e^{o_{ij}}}{1 + e^{o_{ij}}}$$

A target probability is defined according to desired ranking:

$\bar{P}_{uv}$ is set to $\lambda > 0.5$ if *u* is ranked higher than *v*, and $1 - \lambda$ otherwise.

Output and target probability are compared with cross-entropy loss:

$$C_{uv} = -\bar{P}_{uv} \log P_{uv} - (1 - \bar{P}_{uv}) \log(1 - P_{uv})$$



# Results

| Input | Groundtruth | Random forest (Shotton et al., CVPR 2011) | ConvNet w/ DrLIM pretraining (Hadsell/ Chopra/ Lecun, CVPR 2006) +classical backprop | ConvNet w/ spatial pretraining + spatial backprop **(Our method)** |
|---|---|---|---|---|

CDC4CV Poselets dataset
(Holt et al., 2011)

# Experimental results: accuracy

| Methods | Accuracy |
|---|---|
| Randomized forest (Shotton et al., 2011) | 60.30% |
| Spatial Randomized forest (Jiu et al., 2013) | 61.05% |
| Single-scale (vanilla) ConvNet (LeCun et al., 1998) | 47.17% |
| Multi-scale ConvNet (Farabet et al., 2012) | 62.54% |

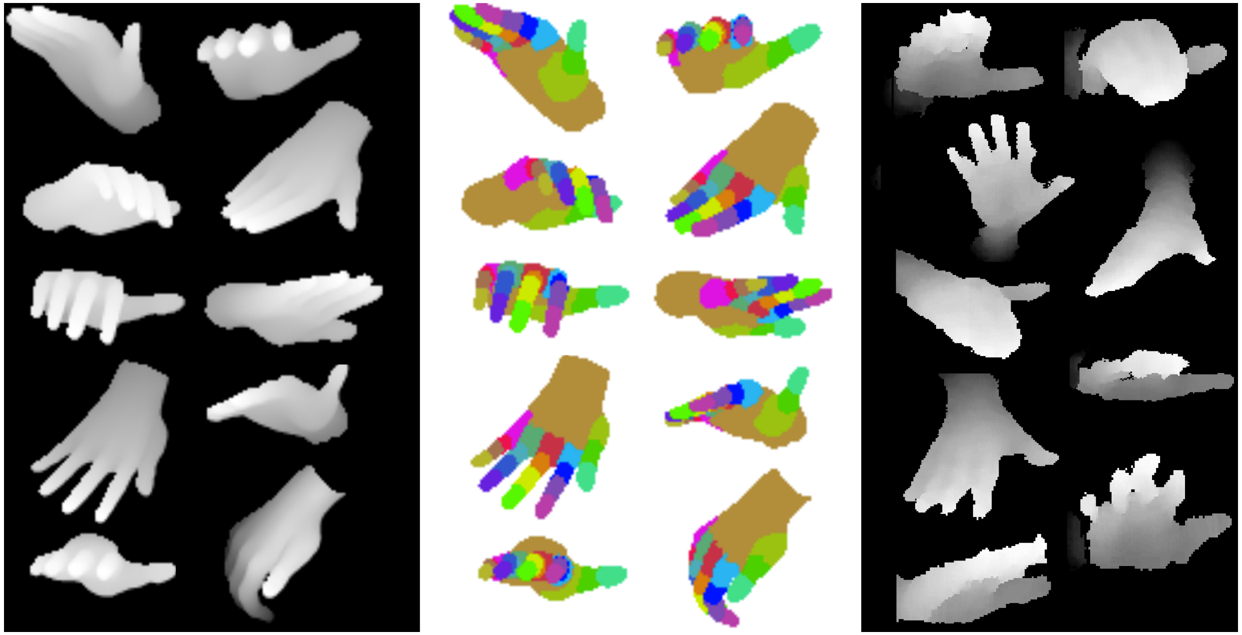
| Convolutional layers | LR | Fine-tuning | Accuracy |
|---|---|---|---|
| DrLIM (Hadsell et al., 2006) | classical | no | 35.10% |
| DrLIM (Hadsell et al., 2006) | spatial | no | 41.05% |
| spatial | classical | no | 38.60% |
| spatial | spatial | no | **41.65**% |
| DrLIM (Hadsell et al., 2006) | classical | yes | 64.39% |
| DrLIM (Hadsell et al., 2006) | spatial | yes | 65.12% |
| spatial | classical | yes | 65.18% |
| spatial | spatial | yes | **66.92**% |

CDC4CV Poselets dataset
(Holt et al., 2011)

# Hand part segmentation

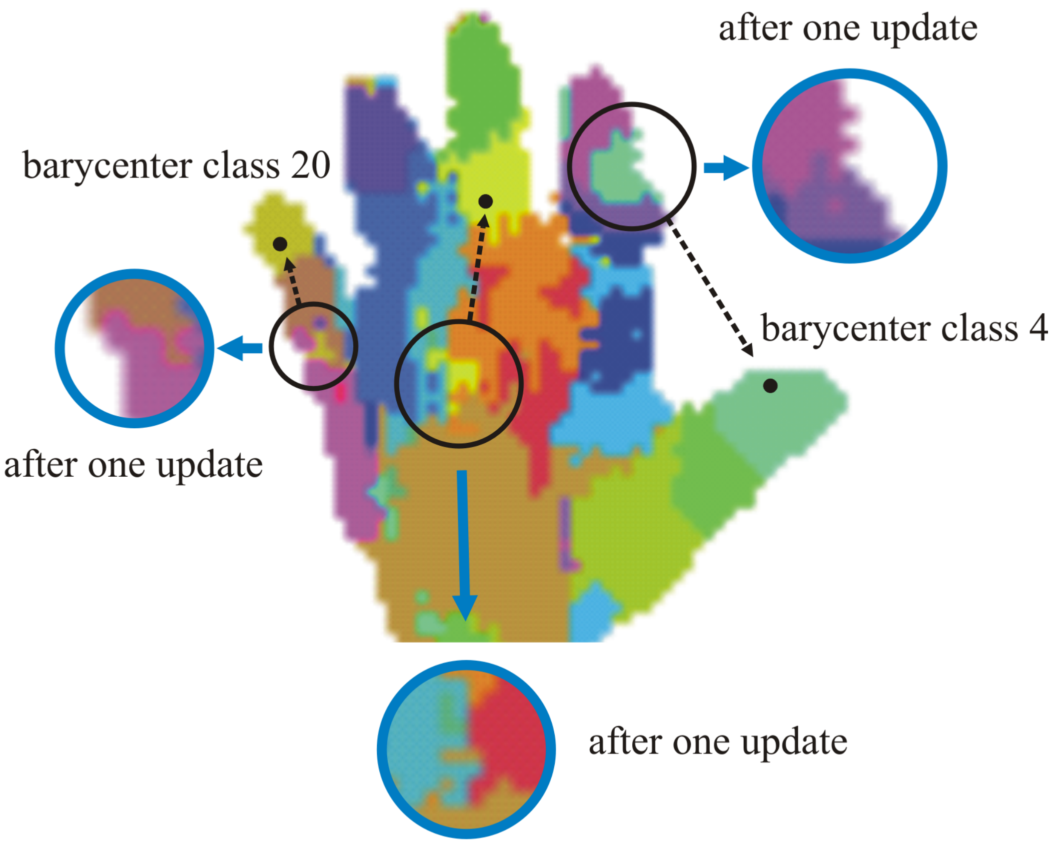
- Structured Deep learning
- Real time necessary
- Training set: 600.000 frames
  - labelled synthetic data
  - Unlabelled real data

PhD of Natalia Neverova

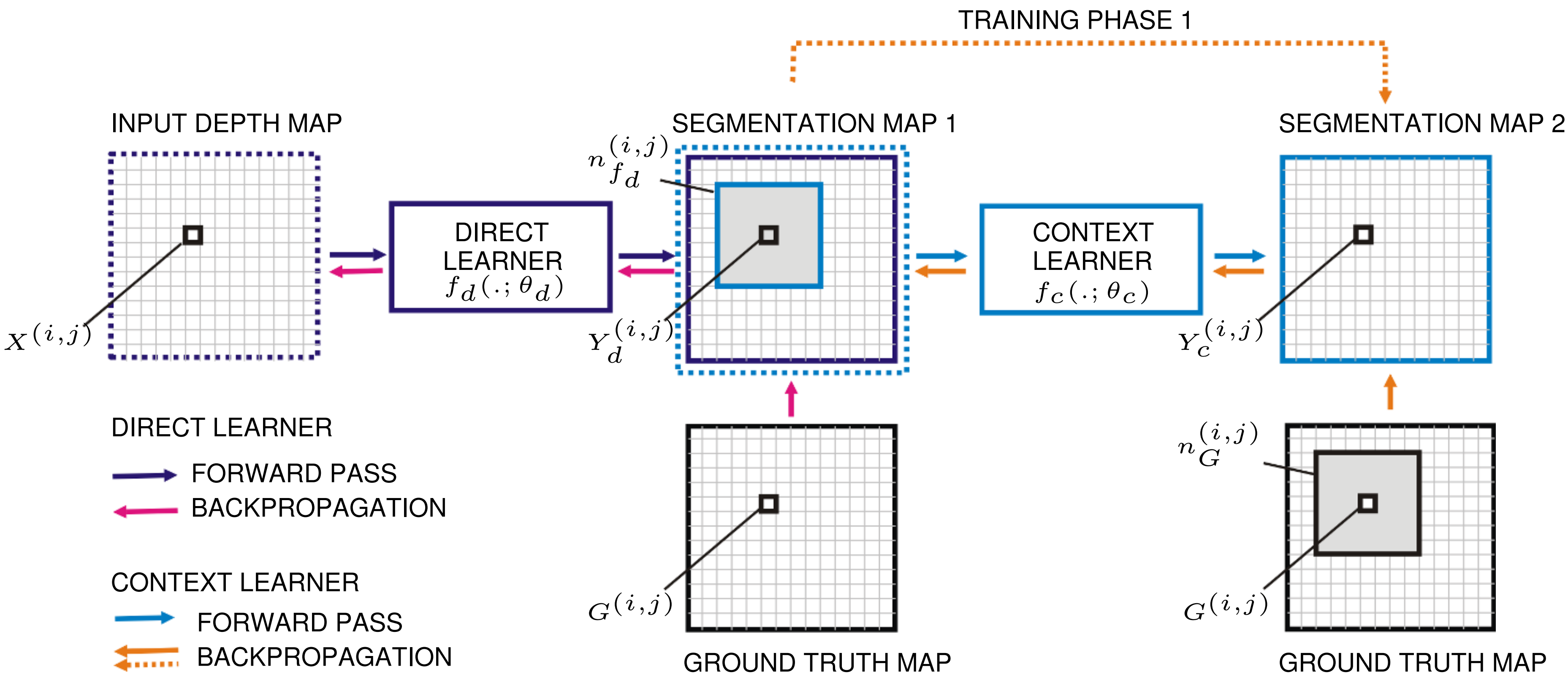


# Structural information

- A single region is supposed to exist for each label
- Unconnected outlier pixels are identified and punished
- No regularization during testing: pixelwise classification



# Learning context



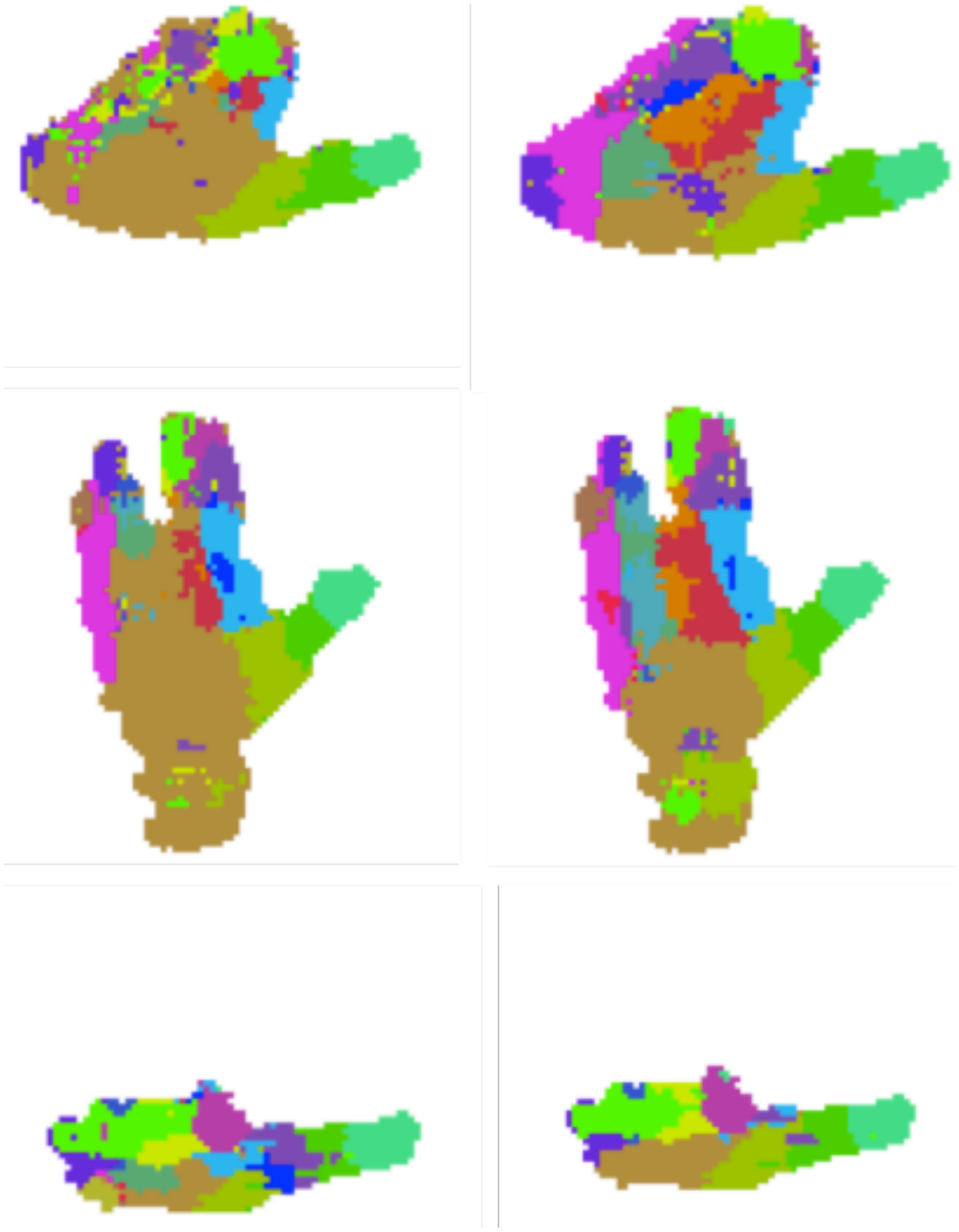
# Results

On 50 manually annotated frames (real data)

| Loss function | Training data | Test data | Accuracy | Average per class |
|---|---|---|---|---|
| $Q_{sd}$ (supervised baseline) | synth. | synth.<br>real | 85.90%<br>47.15% | 78.50%<br>34.98% |
| $Q_{sd} + Q_{loc} + Q_{glb}$<br>(semi-supervised, ours) | all | synth.<br>real | 85.49%<br>**50.50%** | 78.31%<br>**43.25%** |

| Terms | $Q_{loc}$ | ${Q_{glb}}^{+}$ | ${Q_{glb}}^{+} + {Q_{glb}}^{-}$ | $Q_{loc} + {Q_{glb}}^{+} + {Q_{glb}}^{-}$ | $Q_{sd}$ |
|---|---|---|---|---|---|
| Requires labels | no | no | no | no | yes |
| Gain in % points | +0.60 | +0.36 | +0.41 | **+0.82** | +16.05 |

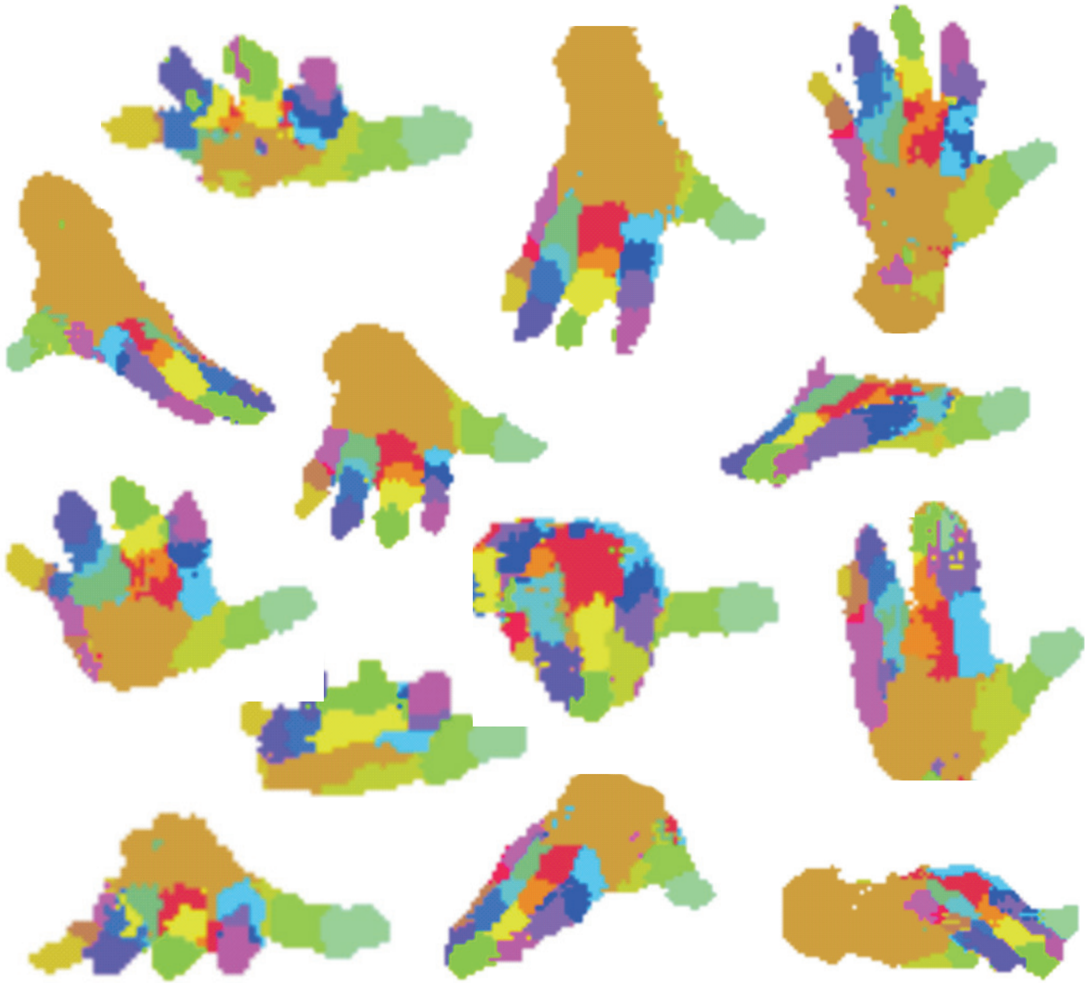
LIRIS cnrs



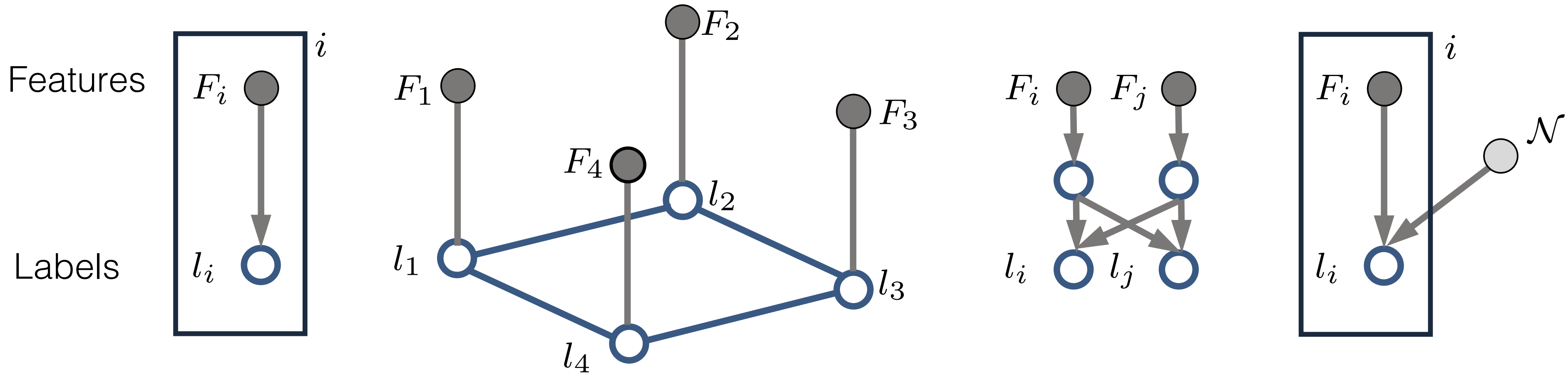
# Results on real images : one step of unsupervised training

LIRIS cnrs

# Results on real images



# Conclusion




- Many applications need highly efficient (real time) segmentation algorithms
- Traditional graphical models are unsuited
- Including structural terms into training (as opposed to testing) can help