Poster Number: CI-2.4

4D Convolutional Neural Networks for Multi-Spectral and Multi-Temporal Remote Sensing Data Classification

M. Giannopoulos, G. Tsagkatakis and P. Tsakalides

Signal Processing Lab, Institute of Computer Science - FORTH, Crete, Greece

Computer Science Department, University of Crete, Crete, Greece

Motivation

- Multi-Spectral (MS) & Multi-Temporal (MT) imaging
- **High-dimensional data**, time-series
  - Capture multi-dimensional dependencies & correlations
  - Extend current CNN architectures
- Remote Sensing (RS) data classification
  - Semantic segmentation
  - Land-cover classification, flood detection
  - Physical characteristics monitoring

Contributions

- Introduction of 4D-CNNs for MT-RS land-cover classification
- Effective exploitation of higher-order correlations without any information loss
- End-to-end learning of spatio-spectro-temporal features at the same time
- Demonstration of the 4D-CNN superiority over lower-dimensional CNNs and state-of-the-art methods

Proposed Method-Stacked Convolution

- Main objective**:** Perform 4D convolution
- How: Stacking multiple sequences of 3D convolutions along the last dimension

$$y_{k,l,m,n} = f\left(\sum_{c}^{C_{in}}\sum_{s=0}^{S-1}\sum_{t=0}^{T-1}\sum_{i=0}^{H-1}\sum_{j=0}^{W-1} w_{i,j,t,s}x_{c,(k+i)(l+j)(m+t)(n+s)} + b_{i,j,t,s}\right)$$

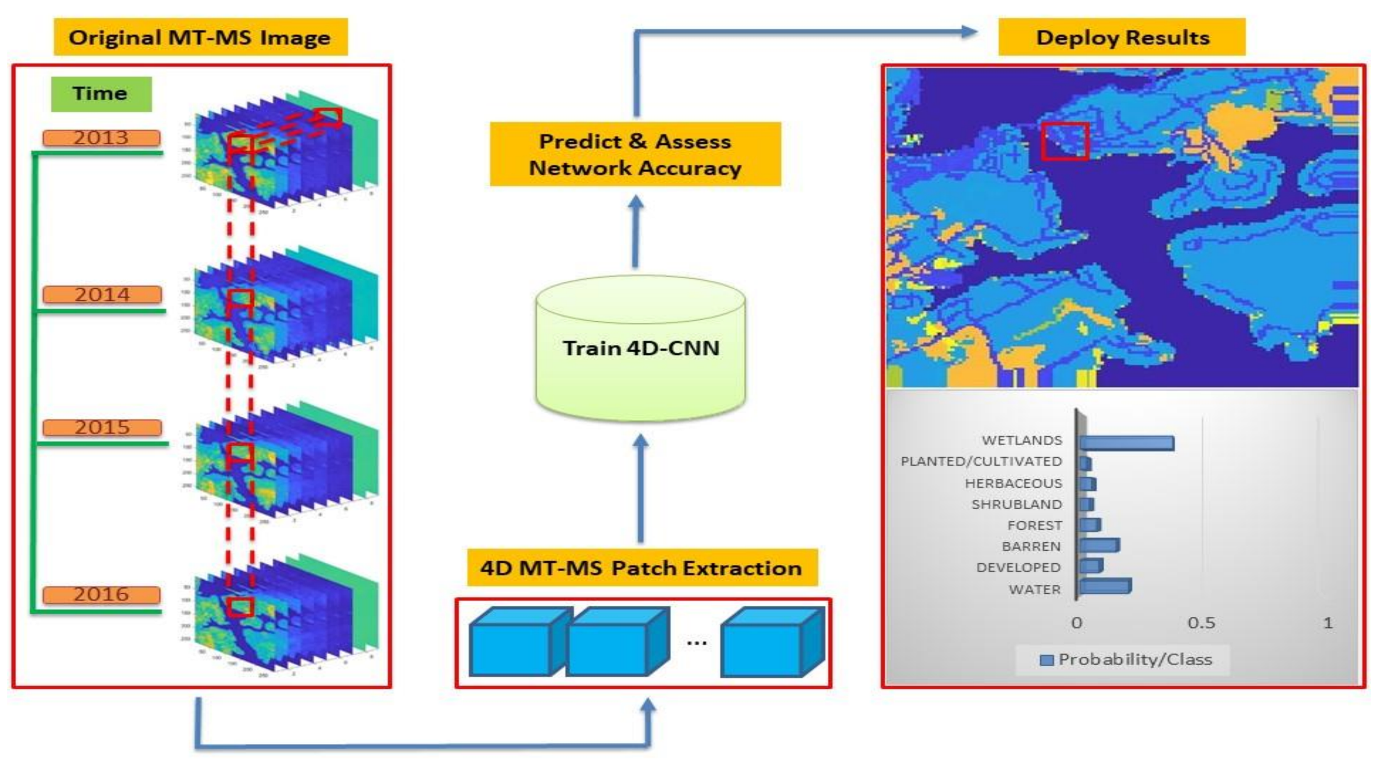
$$= f\left(\sum_{s=0}^{S-1}\left[\sum_{c}^{C_{in}}\sum_{t=0}^{T-1}\sum_{i=0}^{H-1}\sum_{j=0}^{W-1} w_{i,j,t,s}x_{c,(k+i)(l+j)(m+t)(n+s)}\right] + b_{i,j,t,s}\right)$$

$$= f\left(\sum_{s=0}^{S-1} C_{3D} + b_{i,j,t,s}\right)$$

- Why going stacked: **Fast** GPU-based **implementation**, using **Tensorflow** primitives
- **Is it feasible?** Yes
  - Convolution: Linear operation → Summation order can change
  - Implementation: Further re-arrangement of sums' indices → Not separable convolution
- Also applied in semantic segmentation and video classification tasks

4D-CNN for MT-RS Land-Cover Classification

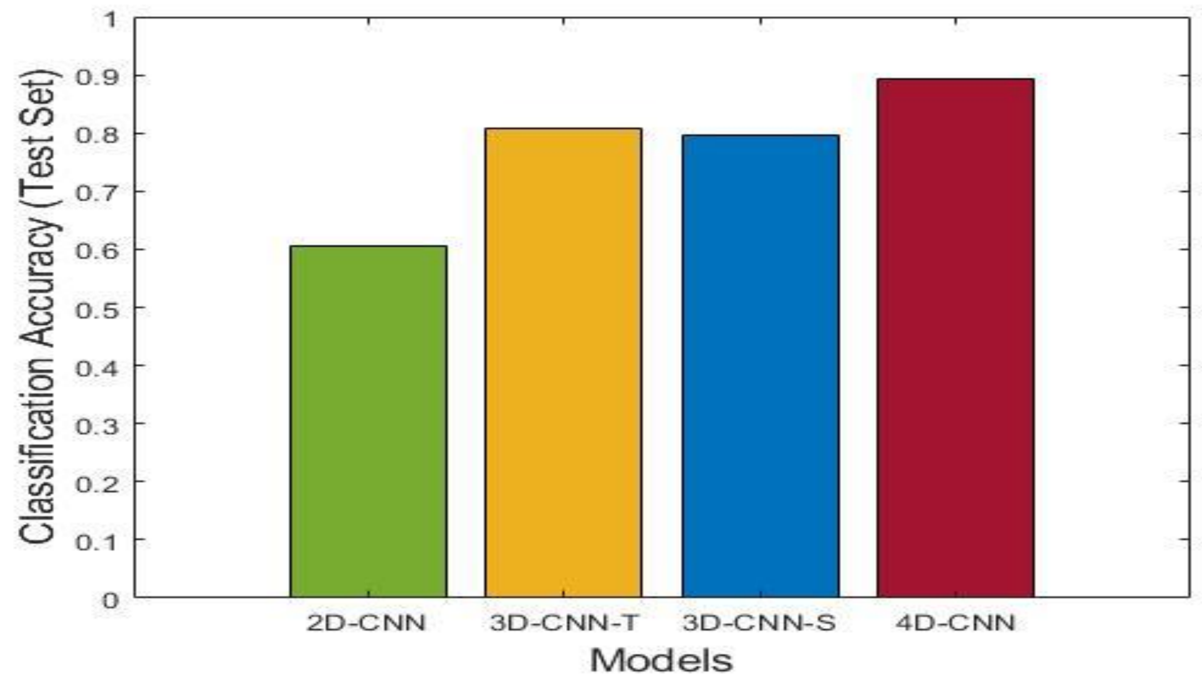
- Training samples: Overlapping 4D patches around each pixel of raw MT-RS imagery

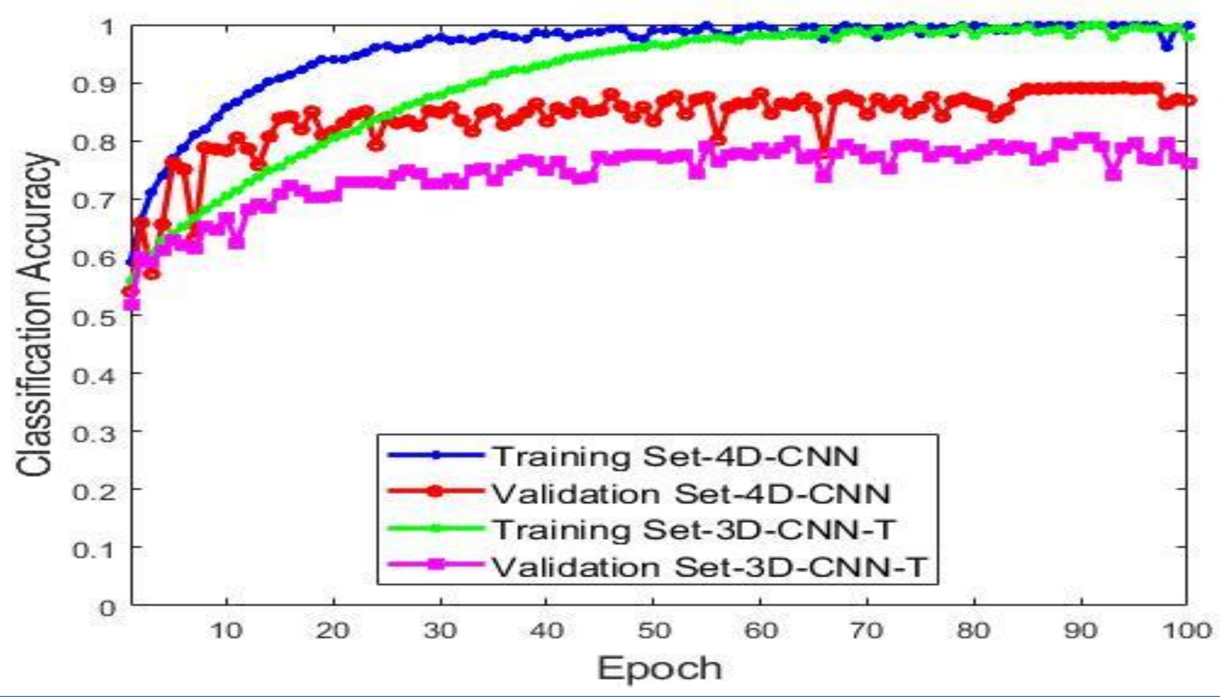


- **Model architecture:** Fully-Convolutional (FC) network
  - No need for considering scaling/translation factors
  - FC networks perform remarkably in similar tasks (e.g. Hyper-Spectral pixel-level classification)
- Model topology
  - Stacks of (Conv-BN-ReLU) layers
  - Kernel size equal to 3 across every dimension, "same" padding
  - Loss-function: Categorical cross-entropy, Optimizer: Adam
  - Batch-size=128, Epochs: 100

4D-CNN Parameter Tuning

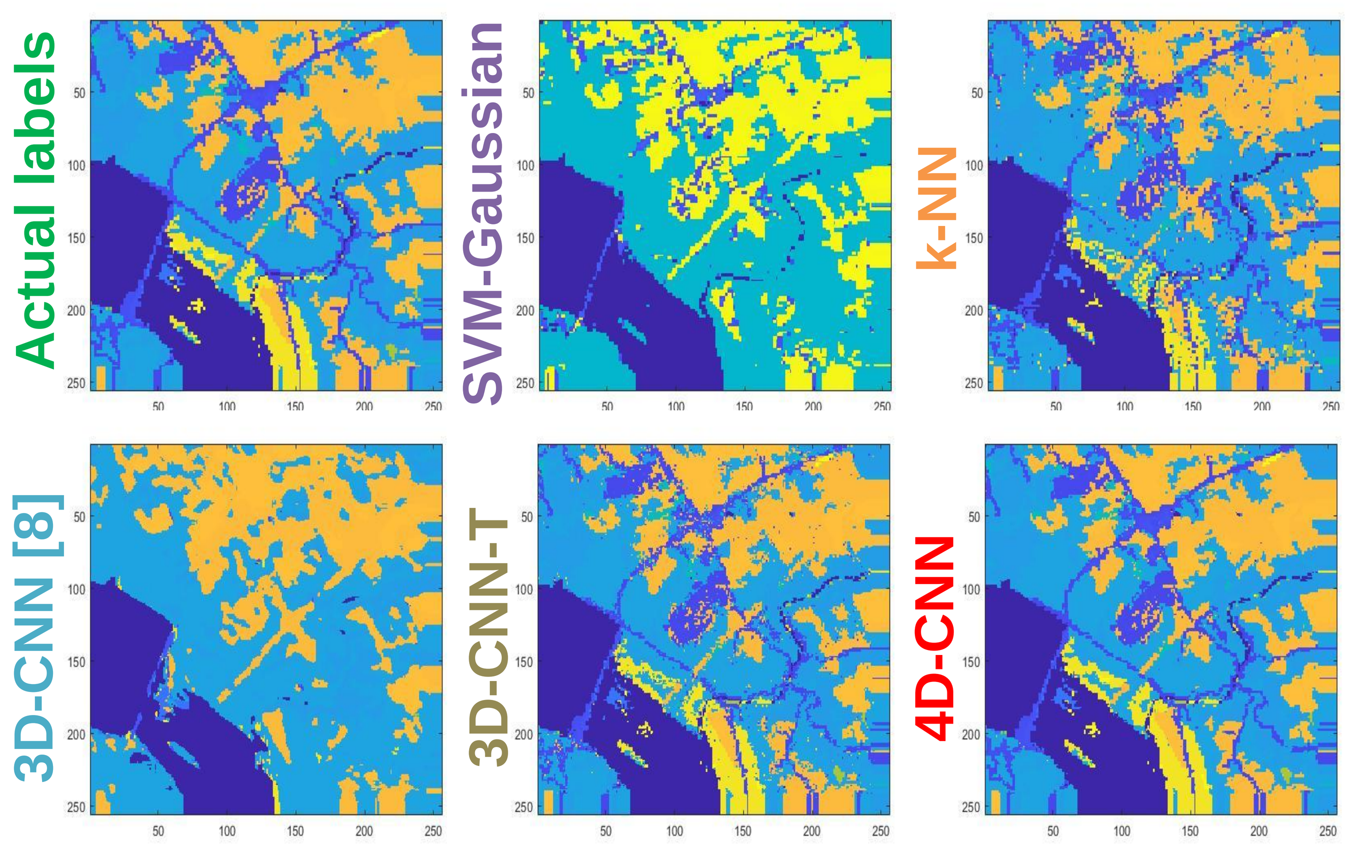
- Input data dimensionality impact
  - 2D-CNN: Exploit only spatial information
  - 3D-CNN-T: Exploit spatial & temporal information
  - 3D-CNN-S: Exploit spatial & spectral information
  - **4D-CNN**: Exploit spatial & spectral & temporal information
- **Hyper-parameter optimization**
  - #Stack-of-layers in each CNN architecture (i.e. 2, 3, 4)
  - Spatial patch-size of training samples (i.e. 5x5, 7x7)





Comparison to State-of-the-Art and Machine Learning Methods

Model	Accuracy	Time	F1-Score
k-NN	0.7443	0.000	0.5747
SVM-Gaussian	0.5945	2.006	0.4995
3D-CNN [8]	0.7097	6.003	0.5090
2D-CNN	0.6049	4.928	0.3624
3D-CNN-T	0.8080	14.558	0.6509
3D-CNN-S	0.7953	27.435	0.6552
4D-CNN	**0.8916**	**107.619**	**0.7796**



Experimental Setup

- Dataset Description: IEEE GRSS Data Fusion Contest dataset
  - Landsat-8 MS imagery (9 spectral bands), collected once a year (2013-2016)
  - USGS NLCD labels of 2016, 15 different land-cover classes (spatial resolution=30m)
  - 2250 different tiles over Maryland, USA
- **Pre-processing steps**
  - Selection of a tile with no undefined label values
  - Spatial padding & sub-sampling to create MT-RS imagery of spatial size 256x256
  - Spatial patch-extraction around each pixel, across all spectral bands & time-instances
  - Created data: 256x256=65536 samples, of size (p,p,9,4) each
  - Data-split: Training/Validation/Test → 60%/20%/20%, random
- Platforms
  - Software: Python-Tensorflow-Keras, Hardware: NVIDIA Quadro P4000 (8Gb RAM)

Conclusions

- **Efficient classification** of MT-RS data via 4D-CNNs
- **Clear improvements** over lower-dimensional CNNs, machine learning and state-of-the-art methods
- **Robustness** towards class-imbalance regimes

Acknowledgments

This work was funded by the Hellenic Foundation for Research and Innovation (HFRI) and the General Secretariat for Research and Innovation (GSRI), under the HFRI Faculty Grant 1725 (V4-ICARUS), and by the CALCHAS project (contract no.842560) of the H2020 Framework Program of the European Commission.