

Some methods to deal with data scarcity in Deep Neural Networks for medical images

Antoine Manzanera

ENSTA Paris



ÉCOLE STIC 2023 - Monastir, Tunisia
November 2023



Contributing Students (1/2)

This presentation is essentially based on the works of the following PhD students (1/2):

- **Clément Pinard** - cosupervised with Parrot (w. Laure Chevalley and David Filliat)
- **Josué Ruano Balseca** - cosupervised with UNAL Bogotá, Colombia (w. Eduardo Romero)
- **John Archila** - cosupervised with UIS Bucaramanga, Colombia (w. Fabio Martínez)



Clément



Josué



John

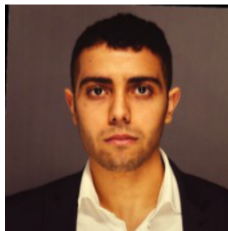
Contributing Students (2/2)

This presentation is essentially based on the works of the following PhD students (2/2):

- **Souhir Khessiba** - cosupervised with Université de Monastir, Tunisia (w. Asma Ben Abdallah and Ahmed Ghazi Blaiech)
- **Marwane Hariat** (w. David Filliat)
- **Juan Andrés Olmos** - cosupervised with UIS Bucaramanga, Colombia (w. Fabio Martínez)



Souhir



Marwane



Juan Andrés

Motivations: Dealing with data scarcity in DNN

- Deep Neural Networks are usually associated with *Data Greed*, i.e. the need for a huge quantity of data to train those networks.
- Many domains - such as Medicine - have strong constraints limiting data production:
 - ▶ Cost: of acquisition and annotation
 - ▶ Need for very high expertise
 - ▶ Ethics and Confidentiality issues
 - ▶ Difficulty to represent Rare classes

Presentation Outline

- 1 Introduction
- 2 Data augmentation
- 3 *Ad hoc* Features
- 4 Second order pooling
- 5 Numerical Phantoms
- 6 Self supervised learning

Why data greed?

Two main reasons why DNN training demands many data:

- ① *Combinatorics*: In spite of efforts to reduce the vanishing gradient effect, the amount of gradient that impacts the first layers (the closest to the input data) during the backpropagation is very small. Then a large number of training iterations is needed for the model to converge.
- ② *Variability*: The DNN needs to acquire a representative sample of the possible data, which is often impossible to meet.

Presentation Outline

- 1 Introduction
- 2 Data augmentation**
- 3 *Ad hoc* Features
- 4 Second order pooling
- 5 Numerical Phantoms
- 6 Self supervised learning

One straightforward solution?

Data augmentation is common in DNN training.

- + It is useful to address the *combinatorial* aspect of data greed.
- ± It may also cover one side of the variability, that we refer to as *aleatoric* (noise, illumination / colour, geometric transforms...)
- However it cannot deal with *epistemic* variability, i.e. it cannot compensate for the lack of knowledge about many types of data that have not been seen by the DNN.

Medical data augmentation? (1/2)

The use of data augmentation must be made with caution regarding its relevance with respect to the application:

- Add noise → supposes knowledge from image physics and sensor properties...
- Change illumination / colour → may make sense for some modalities (RGB,...)
- Geometric transforms (Rotation, Scale, Symmetry,...) → Pay attention to anatomical relevance...
- Generative models (Auto-encoders, GANs,...) → Pay attention to singularities...

Medical data augmentation? (2/2)

Generative models are able to produce globally realistic images, but excellent PSNRs or perceptual losses do not protect from annoying singularities:



Made with *Clipdrop...*

Presentation Outline

- 1 Introduction
- 2 Data augmentation
- 3 *Ad hoc* Features**
- 4 Second order pooling
- 5 Numerical Phantoms
- 6 Self supervised learning

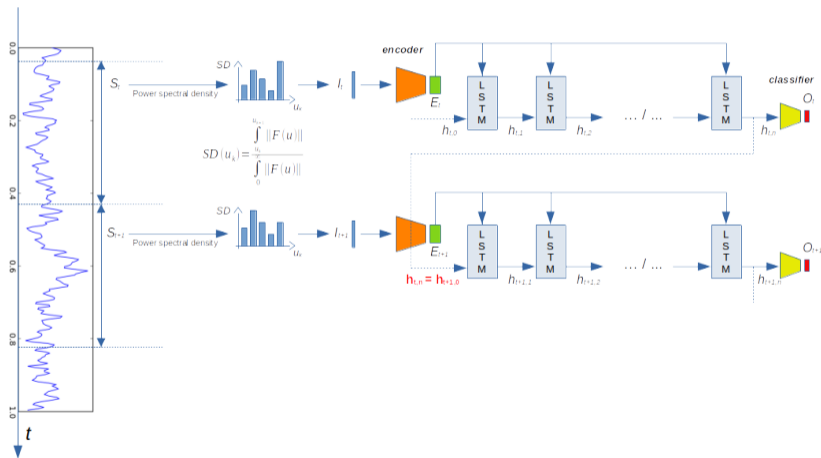
Ad hoc Features?

Replacing raw medical data by pre-computed features is a classic and still useful solution to address data scarcity since:

- it generally reduces dramatically the size of input data, and then the depth of neural network.
- it provides domain-interpretable data that can be "semantically" augmented more easily.

Example 1: Classifying EEG data

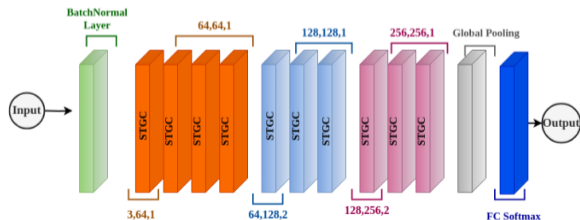
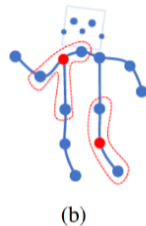
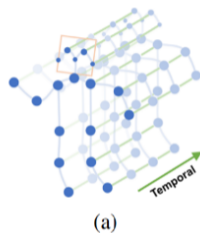
The input data is reduced to relative power of relevant frequency bands.



PhD work of
Souhir Khessiba
[Khessiba 22]

Example 2: Classifying Gait Videos

The input data is reduced to a 3d graph of human pose sequence.



PhD work of Souhir Khessiba

[Khessiba 23]

Presentation Outline

- 1 Introduction
- 2 Data augmentation
- 3 *Ad hoc* Features
- 4 Second order pooling**
- 5 Numerical Phantoms
- 6 Self supervised learning

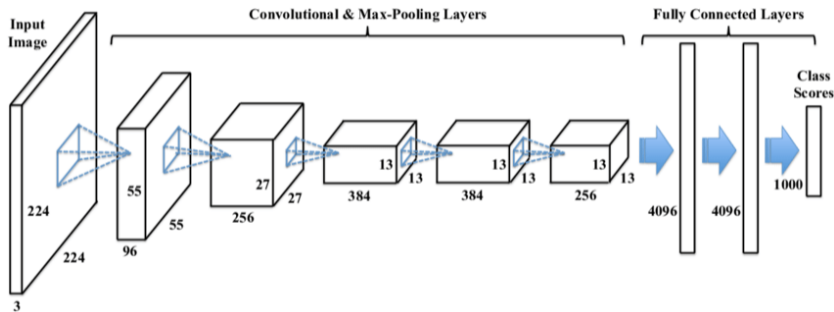
Second order pooling

Reducing the depth (and number of parameters) of a DNN can also be made through posterior processing, i.e. between the encoder and classifier modules of the network.

- One classic alternative to applying *Flattening* at the end of the encoder (i.e. last Convolutional layer) before Fully Connected (Classification) layers, is to use *Global pooling* as a way to summarize the features in a channel-wise manner.
- So the basic dilemma is: adding more layers with local pooling to reduce the size of the encoding, or using global pooling earlier to reduce each channel map to a single (max or average) statistics.
- One interesting tradeoff is: using an *earlier* and *richer* global pooling such as *second order (= covariance) pooling*.

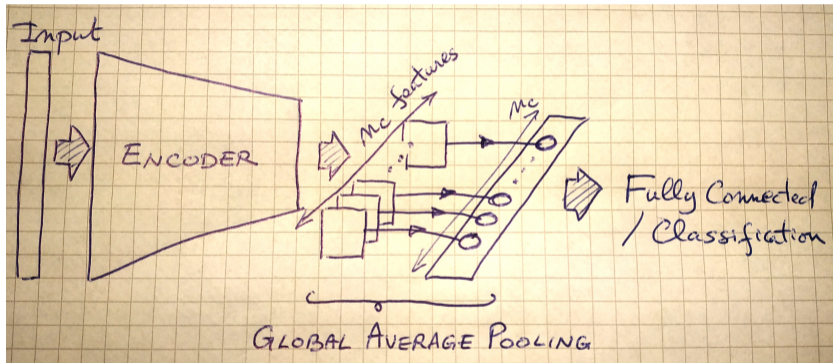
Classic classification network without Global Pooling

Classically, only iterated (layer-wise) *Local* pooling in the convolutional branch, then *flattening* features from the latent space to enter the Fully Connected branch (ex: AlexNet).



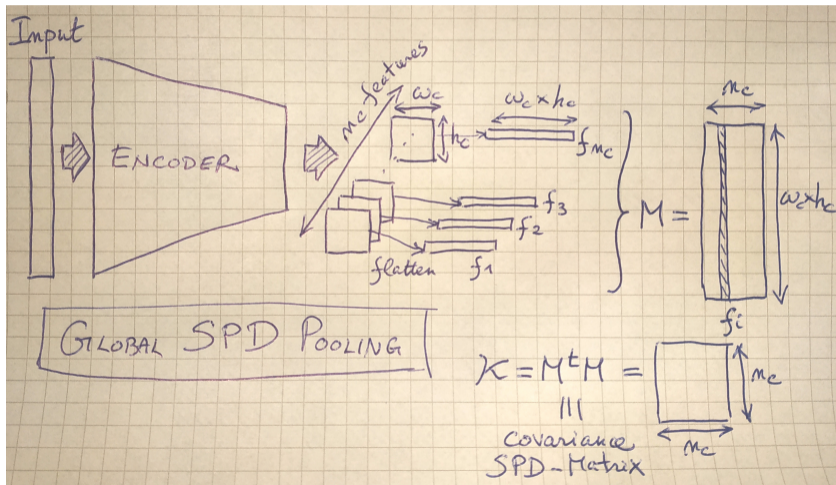
Scalar Global (e.g. Average) Pooling

Scalar Global Pooling summarizes each feature map to a single statistics (e.g. Average).



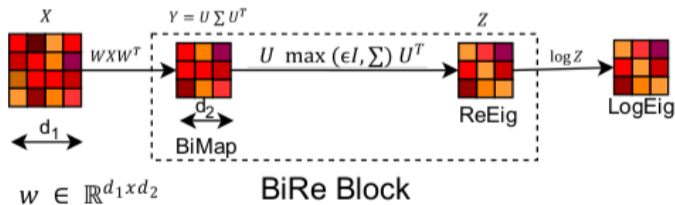
Second order (SPD) Global Pooling

Symmetric Positive Definite (SPD) Pooling summarizes the collection of feature maps by their covariance along the space dimensions.

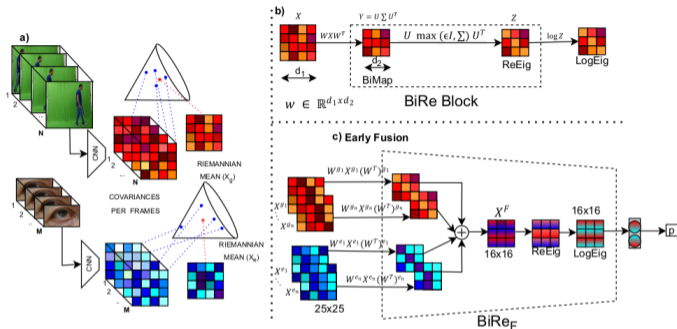


What after SPD Pooling? Riemannian Network!

Since SPD Matrices do not lie in Euclidean space, but in Riemannian manifold, they must be processed in a specific framework: (1) Bilinear Mapping (with learning weight matrix), (2) Rectified Eigenvalues (counterpart of ReLU layer), and (3) Logarithm Eigenvalue (to backproject in Euclidean space for classification).

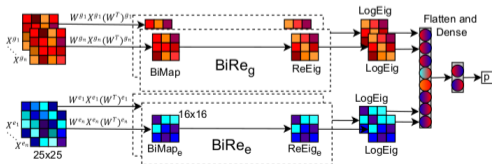


Parkinson Prediction by Riemannian Multimodal Fusion



PhD work of John Archila
[Archila 23]

d) Intermediate Fusion



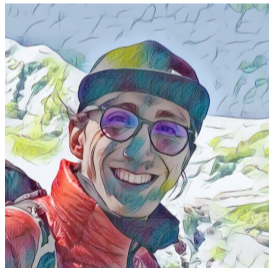
SPD pooling + Riemannian Network

SPD pooling is a tradeoff between (later, iterated) local pooling and (earlier) scalar global pooling, that allows to reduce the number of layers (and parameters) by providing a more elaborate statistics describing correlations between features.

- SPD pooling + Riemannian Net can be thought as characterising the *stationary* part of data. It was used before to extract the *pictural style* of paintings, by separating the content (contingent, subject dependent), from the style (immanent, artist dependent).



Fauvist Clément



Cartoonised Josué



Pointillist John

(Made with *fotor*
Neural Style Transfer)

SPD pooling + Riemannian Network

SPD pooling is a tradeoff between (later, iterated) local pooling and (earlier) scalar global pooling, that allows to reduce the number of layers (and parameters) by providing a more elaborate statistics describing correlations between features.

- SPD pooling + Riemannian Net can be thought as characterising the *stationary* part of data. It was used before to extract the *pictural style* of paintings, by separating the content (contingent, subject dependent), from the style (immanent, artist dependent).



Expressionist Souhir



VanGoghised Marwane



Charcoalised Juan Andrés

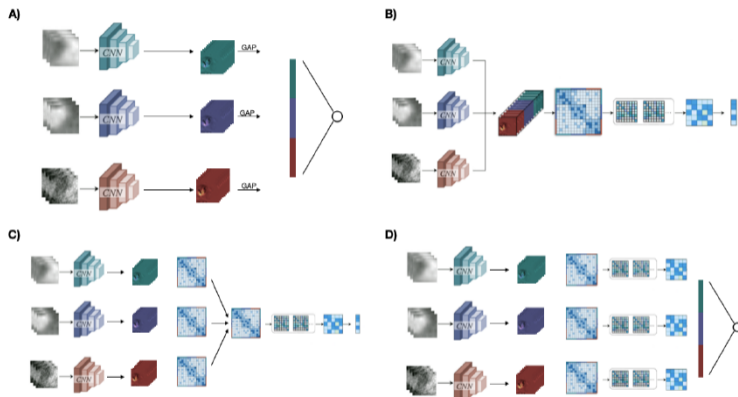
(Made with *fotor*
Neural Style Transfer)

SPD pooling + Riemannian Network

SPD pooling is a tradeoff between (later, iterated) local pooling and (earlier) scalar global pooling, that allows to reduce the number of layers (and parameters) by providing a more elaborate statistics describing correlations between features.

- SPD pooling + Riemannian Net can be thought as characterising the *stationary* part of data. It was used before to extract the *pictural style* of paintings, by separating the content (contingent, subject dependent), from the style (immanent, artist dependent).
- **In the same way we have used it in Parkinson prediction from gesture videos, as a way to separate the "content" (instruction for gesture) and the "style" (patient's pathology).**

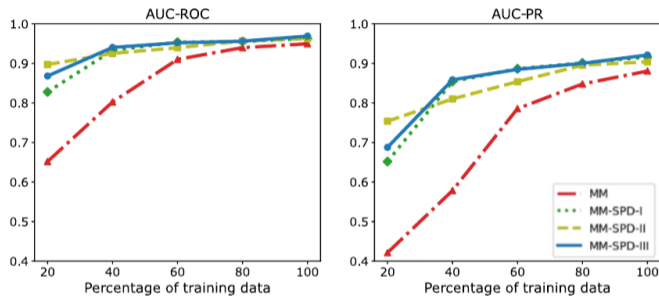
Prostate Lesion Classification from BP-MRI using Riemannian Fusion



- A GAP Baseline
- B Early SPD Fusion
- C Intermediate SPD Fusion
- D Late SPD Fusion

PhD of Juan Andrés Olmos - **[Olmos 23]**

Prostate Lesion Classification from BP-MRI using Riemannian Fusion



The SPD-Riemannian Network performance is much less sensitive to the amount of training images.

PhD of Juan Andrés Olmos - **[Olmos 23]**

Presentation Outline

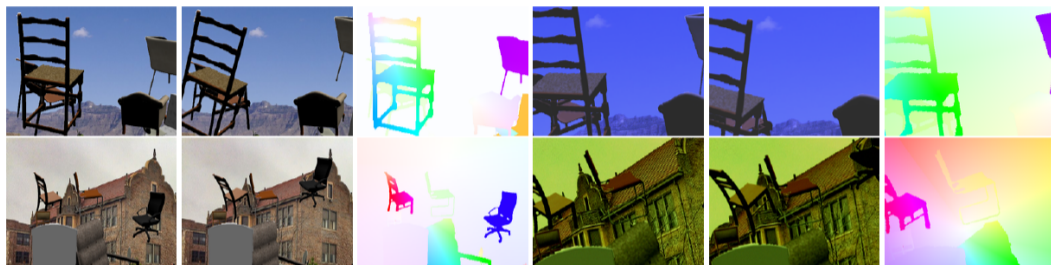
- 1 Introduction
- 2 Data augmentation
- 3 *Ad hoc* Features
- 4 Second order pooling
- 5 Numerical Phantoms**
- 6 Self supervised learning

Numerical Phantoms

We refer here to Numerical Phantom as a synthetic model designed to produce training data *at will*.

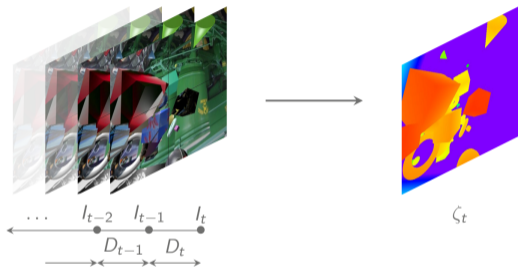
- The numerical phantom is designed to contain as much variability and difficulty as possible that can be found in real data for the target task.
- This does not necessarily imply physical realism, it may even be the contrary. Fine examples from Computer vision are: *Flying Chairs* and *Still Box*.

Flying Chairs for training Optical Flow predictors



The FlyingChairs synthetic dataset **[Fischer 15]** provides dense annotations on scenes integrating different level of typical optical flow difficulties: homogeneous areas, thin objects, holes, occlusions, large speed range, etc. Furthermore the data can be easily augmented (on the right).

Still Box for training Depth Map predictors in flying Drone context

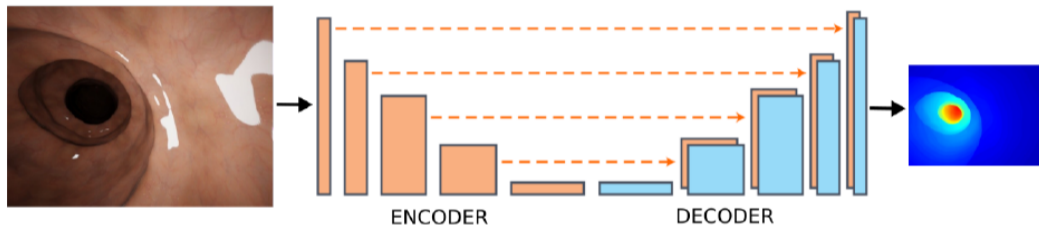


Supervised learning of depth from synthetic sequences

[Pinard 17]

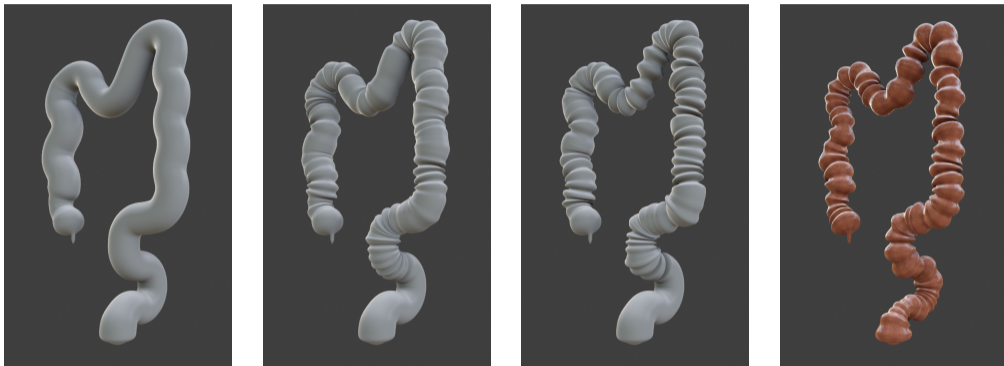
- Unrealistic scenes \leftrightarrow Abstraction of the context
- Focus on geometry / motion, not on appearance / context
- Trained on rotationless movement, at a constant speed

Learning Shape from Shading for Automated Colonoscopy



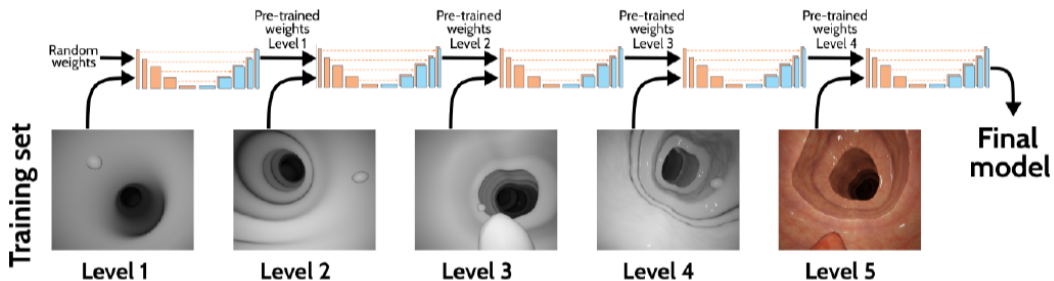
Images from synthetic videos are used to train a CNN using a loss function based on the ground truth depthmap **[Ruano 23]**

Curriculum Learning Shape from Shading for Automated Colonoscopy



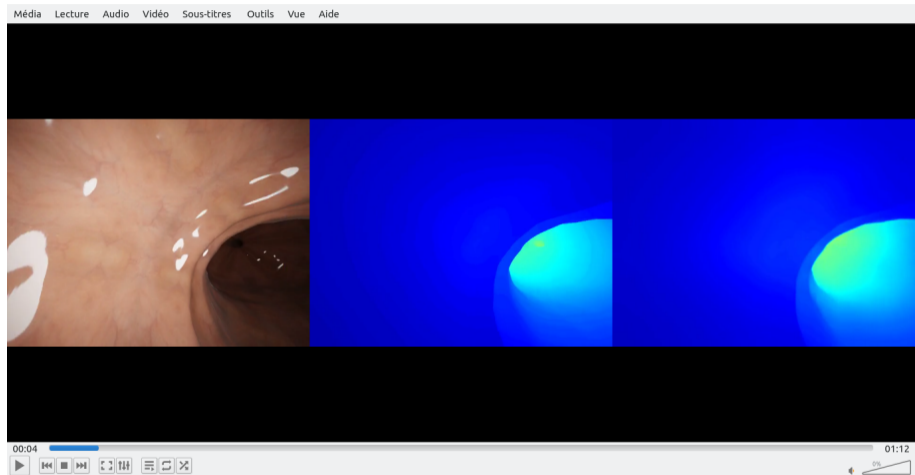
Synthetic exploration videos are created from a hierarchy of synthetic colons of increasing complexity [Ruano 23]

Curriculum Learning Shape from Shading for Automated Colonoscopy



The training is performed with progressive complexity [Ruano 23]

SfSNet on Synthetic Videos



ShapeFromShadingNet on Synthetic Test Videos **[Ruano 23]**

SfSNet on Real Videos



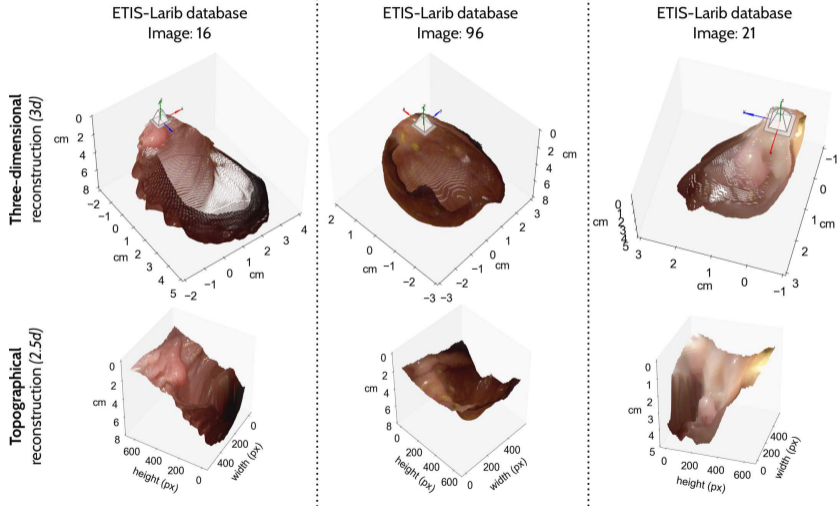
ShapeFromShadingNet on Real Videos [Ruano 23]. Single images seem to be sufficient in such particular context!

3d reconstruction from depth maps

Back-projection from the depth map Z :

$$M = Z(m)K^{-1}m$$

[Ruano 23]



Presentation Outline

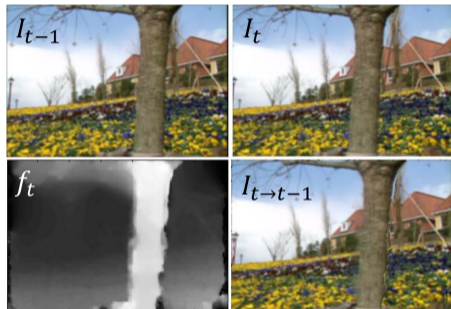
- 1 Introduction
- 2 Data augmentation
- 3 *Ad hoc* Features
- 4 Second order pooling
- 5 Numerical Phantoms
- 6 Self supervised learning**

Self supervised learning

Self supervised learning is based on an *indirect evaluation* of the loss function, through a *secondary task*, that the network can perform *itself*, without supervision. For example:

- Photometric loss based on image warping for *optical flow* prediction.
- Photometric loss based on image backprojection-reprojection for *depth maps* prediction.
- Any examples from medical images detection/segmentation?

Self-supervised Learning Optical Flow Prediction



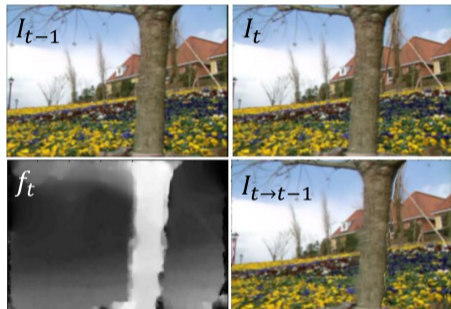
Self-supervised learning (or fine tuning) can be made on real images by using a photometric loss function, that quantifies the difference between an image and its prediction based on the optical flow:

$$\mathcal{L}_{ph} = \|I_{t-1} - I_{t \rightarrow t-1}\|,$$

with:

$$I_{t \rightarrow t-1}(\mathbf{x}) = I_t(\mathbf{x} + f_t(\mathbf{x})).$$

Self-supervised Learning Optical Flow Prediction



However, additional difficulties occur:

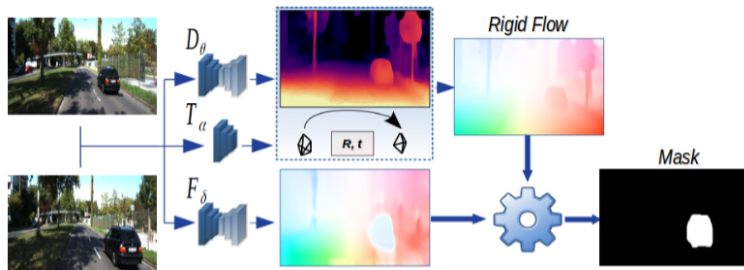
- Homogeneous areas
- Occlusion areas

This implies - among other - a finer modelling of the loss function, for example:

$$\mathcal{L}_{zh} = ||(I_{t-1} - I_{t \rightarrow t-1}) || \nabla I_{t-1} || ||$$

$$\mathcal{L}_{occ} = \min (||I_{t-1} - I_{t \rightarrow t-1}||, ||I_t - I_{t-1 \rightarrow t}||)$$

Self-supervised training of Optical Flow, Odometry and Depth

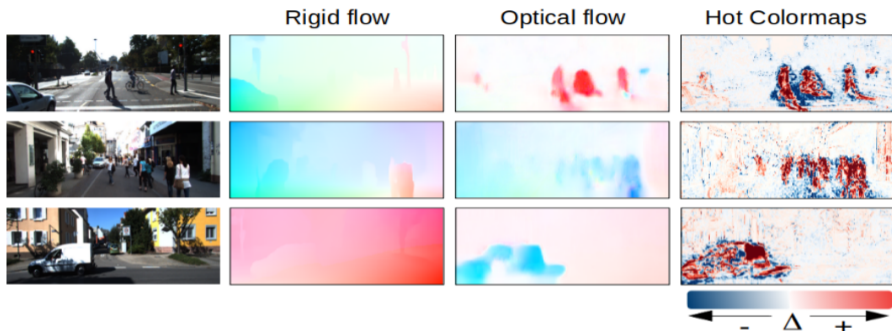


CoopNet [Hariat 23]

The joint estimation of camera displacement (rotation \mathbf{R} and translation \mathbf{t}) and depth map D_θ allows, by backprojection of the first image in 3d, then reprojection onto the second view, to predict the *rigid flow*, which is the apparent velocity field under rigid assumption scene (i.e. only due to camera motion), defined as:

$$[\mathbf{K} | \mathbf{O}_4] [\mathbf{R} | \mathbf{t}] D_\theta(\mathbf{m}) \times \mathbf{K}^{-1} \mathbf{m} - \mathbf{m}$$

CoopNet: Joint training of Optical Flow, Odometry and Depth



CoopNet [Hariat 23]

The CoopNet network is trained based on the difference between the photometric losses from the optical flow and from the depth networks:

$$\Delta(\mathbf{m}) = \mathcal{L}_{\text{photo}}^{\text{depth, odometry}} - \mathcal{L}_{\text{photo}}^{\text{flow}}$$

Conclusion: Some methods amongst others...

- Data scarcity: major problem, in particular for medical applications.
- *Annotated* data scarcity or *Raw* data scarcity?
- Presented Methods today:
 - ▶ **Data augmentation**: Pay attention to meaning and to singularities!
 - ▶ **Ad hoc Features**: Smaller Networks, more interpretable
 - ▶ **Riemannian Networks**: More compact, more expressive
 - ▶ **Numerical Phantom**: Powerful expression, helps interpretability
 - ▶ **Self-Supervised Learning**: Find meaningful secondary tasks!
- Other methods to be considered:
 - ▶ Unsupervised learning
 - ▶ Few-shot / Zero-shot learning
 - ▶ Transfer learning

References (1)

 **[Khessiba 22]** S. Khessiba and A.G. Blaiech and A. Ben Abdallah and K. Ben Khalifa and A. Manzanera and M.H. Bedoui

Convolution Neural Network Hyperparameter Optimization for Vigilance Classification
11e Workshop Applications Médicales de l'Informatique : Nouvelles Approches (AMINA 2022). Monastir, Tunisia. November 2022.

 **[Khessiba 23]** S. Khessiba and A.G. Blaiech and A. Ben Abdallah and R. Grassa and A. Manzanera and M.H. Bedoui

Improving Knee Osteoarthritis Classification with Markerless Pose Estimation and STGCN Model

The IEEE International Workshop on MultiMedia Signal Processing (MMSP 2023). Poitiers, September 2023.

References (2)

 **[Archila 23]** J. Archila and A. Manzanera and F. Martínez

A Riemannian multimodal representation to classify Parkinsonism-related patterns from non-invasive observations of gait and eye movements

Submitted, 2023

 **[Olmos 23]** J.A. Olmos and A. Manzanera and F. Martínez

Improving Prostate Cancer Lesion Classification from Second-Order Networks


Working Document, 2023

 **[Ruano 23]** J. Ruano Balseca and M. Gómez and E. Romero and A. Manzanera

Leveraging a realistic synthetic database to learn Shape-from-Shading for estimating the colon depth in colonoscopy images

Submitted, 2023

References (3)

 **[Fischer 15]** P. Fischer, A. Dosovitskiy, E. Ilg, P. Häusser, C. Hazırbaş, V. Golkov, P. van der Smagt, D. Cremers, T. Brox

FlowNet: Learning Optical Flow with Convolutional Networks

Proceedings of the International Conference on Computer Vision (ICCV), Dec. 2015, pp 2758–2766

 **[Pinard 17]** C. Pinard and L. Chevalley and A. Manzanera and D. Filliat

End-to-end depth from motion with stabilized monocular videos

Int. Conf. on Unmanned Aerial Vehicles in Geomatics (UAV-g) Bonn, pp. 67-74, 2017

 **[Hariat 23]** M. Hariat and A. Manzanera and D. Filliat

Rebalancing gradient to improve self-supervised co-training of depth, odometry and optical flow predictions

IEEE Winter Conf. on Applications of Computer Vision (WACV). Waikoloa, 2023