STORM

Activités Learning / Deep Learning

IRIT
Institut de Recherche en Informatique de Toulouse
CNRS - INP - UT3 - UT1 - UT2J

STORM

STORM

- Team
  - 4 researchers
  - 10 PhD Students and post-doc
  - interns
- Objectives :
  - Domain-centered content creation frameworks for animated computer graphics.

Projects overview

On-going

- Latent Space estimation for path space analysis [simulation]
- Decision trees for transition sampling [simulation]
- Spiking Neural Network on the GPU [GPU programing]
- Deep Learning for point cloud processing [classification]

Perspectives

- Interactive tools for digital painting tools
- Low-data and low-energy

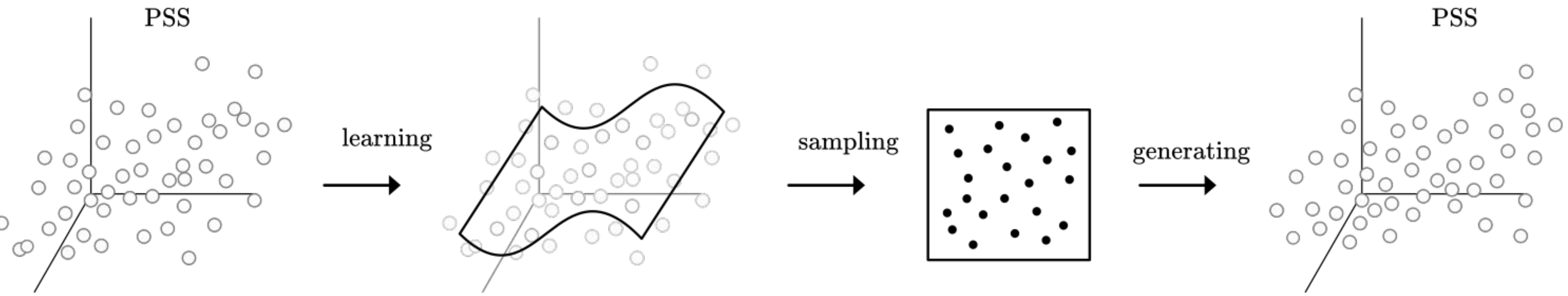
Latent-space estimation for path space analysis
[M. Paulin]

- Context
  - Physically based rendering : lighting simulation in path space
  - ANR project CaLiTrOp, IRIT - LIRIS - INRIA
- Objectives
  - Monte Carlo estimator of the Radiative Transfer Equation
    - Zero variance estimators
    - Faster convergence of the estimator
- Challenges
  - Sampling an infinite dimensional space from an unknown density

Latent-space estimation for path space analysis

[M. Paulin]

- Overview of the approach
  - Gaussian Process Latent Variable Model of the primary sample space $[0, 1]^D$

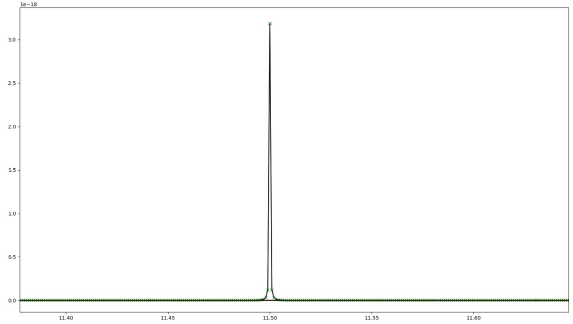


- Key ideas
  - Learn the latent space $[0, 1]^L$, with L << D, from several simulated paths
  - Sampling in latent space : easier than in primary sample space
  - Generating primary samples from latent space samples

Decision trees for transitions sampling

[M. Paulin, N. Mellado], Collaboration R. Fournier (Laplace)

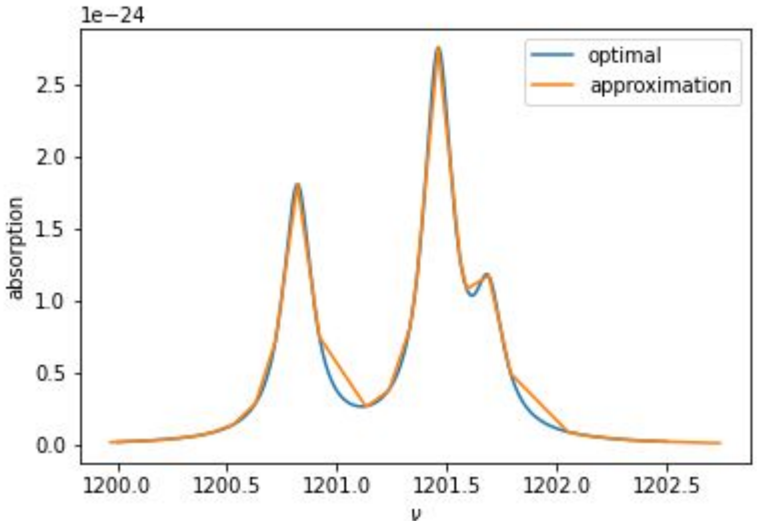
- Context
  - Monte-Carlo Radiative Forcings Computation: simulate radiative exchanges in the atmosphere
  - ANR project MCG-Rad, LMD/IPSL, Laplace, IRIT, Meso-Star
- Objectives
  - Speed-up sampling and improve Monte Carlo estimators
- Challenges
  - Large-scale:
    - absorption spectrums are composed of millions of transitions,
    - simulation of the whole earth atmosphere, for long period of times
  - Physical simulation: cannot approximate

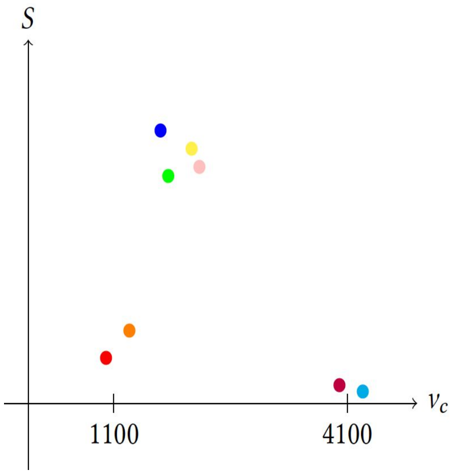


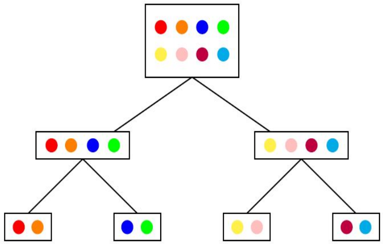
Decision trees for transitions sampling

[M. Paulin, N. Mellado], Collaboration R. Fournier (Laplace)

- Overview of the approach
  - Use decision trees to build hierarchical probability estimators
- Key ideas
  - Approximate the geometry of the probability density functions
  - Tree-based decomposition to reduce traversal complexity







Spiking Neural networks on the GPU

[L. Barthe, N. Mellado], Collaboration D. Longin, S. Torpes (CerCo)

- Context
    - Simulate human brain using spiking neural networks
- Objectives
    - Develop efficient GPU implementation for fast learning and processing (CUDA)
- Challenges
    - Network scale: trillions of neurons
    - Learning optimizes the connections
        - Non-regular [bad for GPU]

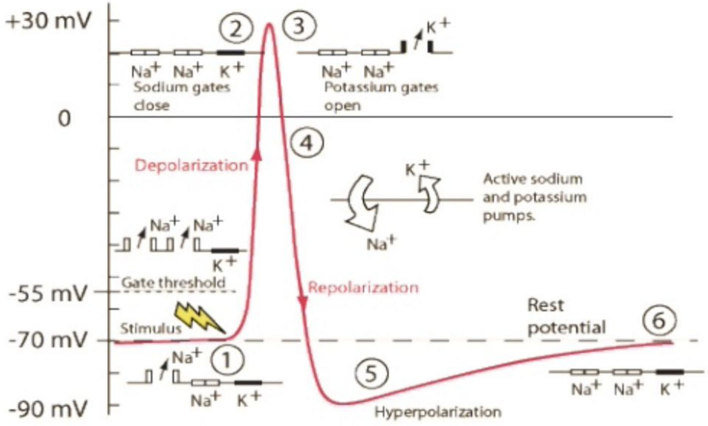
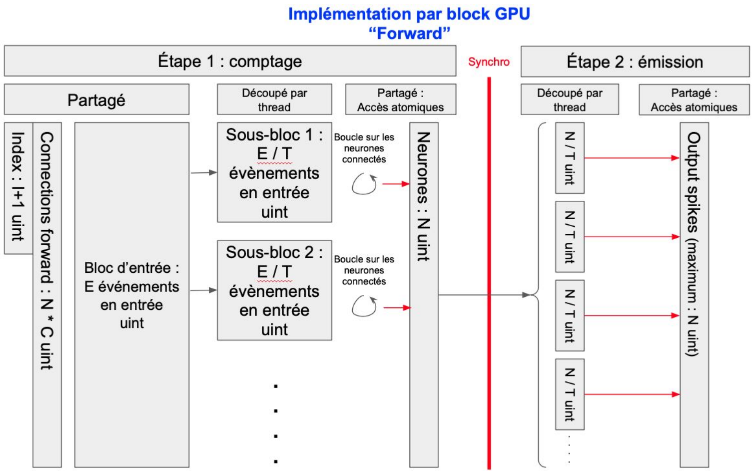


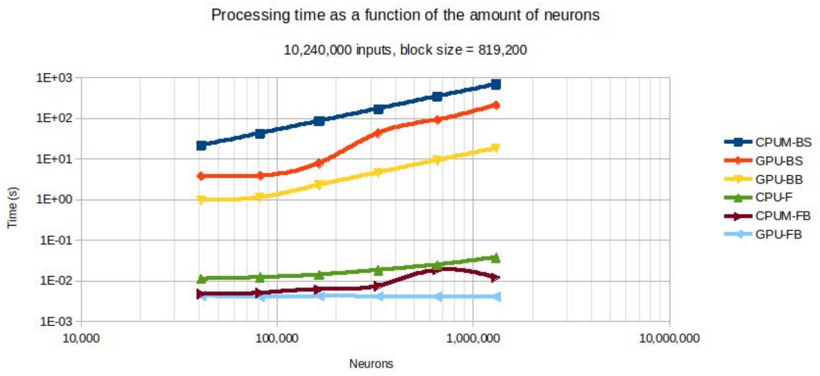
*Figure adopted from http://hyperphysics.phy-astr.gsu.edu/hbase/Biology/actpot.html

Spiking Neural networks on the GPU

[L. Barthe, N. Mellado], Collaboration D. Longin, S. Torpes (CerCo)

- Overview of the approach
  - Benchmark CPU and GPU algorithms
  - Optimize bottlenecks, memory access, cache, ...
- Key ideas
  - Low-level GPU programing requires specific algorithms and datastructures

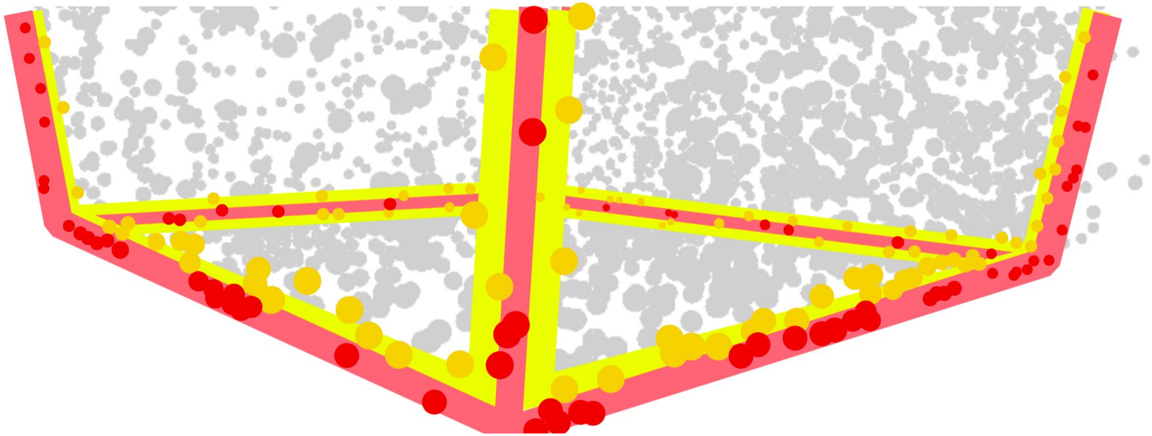




Deep learning for point cloud processing

[L. Barthe, N. Mellado], Collaboration T. Pellegrini

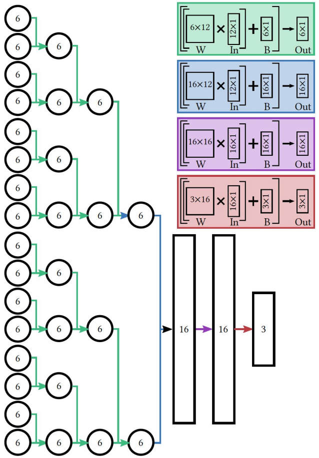
- Context
  - Analysis of acquired 3d point cloud (LiDAR, photogrammetry)
- Objectives
  - Point-wise classification according to geometric properties
- Challenges
  - Point clouds are unordered, irregular, noisy, and large

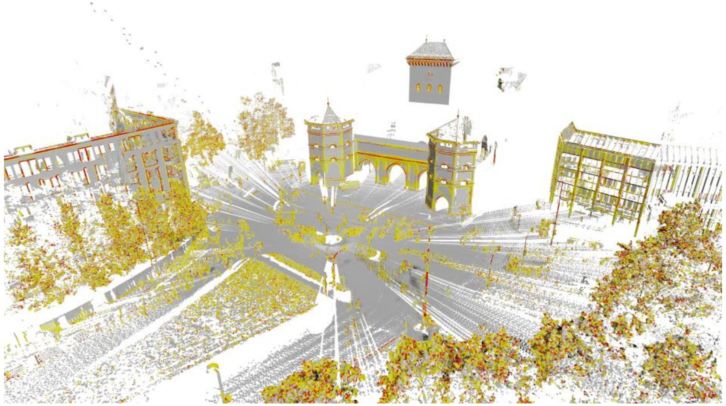


Deep learning for point cloud processing

[L. Barthe, N. Mellado], Collaboration T. Pellegrini

- Overview of the approach
  - Compute multi-scale feature vector
  - Use tailored network architecture
- Key ideas
  - Do not try to learn how to reconstruct surfaces
  - Compact network -> efficiency and low data





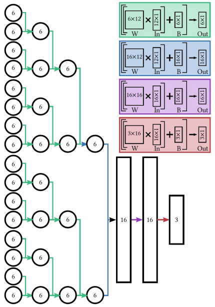
Perspectives

Low data and low energy

- J / #kpoints

Training	PCPNet (Default)	ECNet (EC)	PIENet (ABC)	GLS	CNN (average)	FC (average)	PCEDNet (average)
Time t_K	8.34*	7.32*	25.77*	0.05	14.68 (14.73)	3.97 (4.02)	**0.43 (0.48)**
Energy E_K	2167.42*	1831.05*	6443.78*	9.16	1321.43 (1330.59)	357.60 (366.76)	**38.60 (47.76)**

Table 12. Times t_K (2^{nd} row) and processing unit energy consumption E_K (3^{rd} row) required for processing 1K points when training the different networks (1^{st} row) denoted as *name(training dataset)*. *(average)* represents the average of the times obtained when training on the different datasets Default, ABC and SHREC. PCPNet is trained on an NVIDIA TITAN Quadro RTX 6000 GPU, and ECNet and PIENet are trained on an NVIDIA TITAN X GPU. The times and energy consumption for ECNet and PIENet are computed respectively from the statistics provided in [Yu et al. 2018] and [Wang et al. 2020].



Classification	CA	FEE	PCPNet	ECNet	PIENet (8K pts)	GLS	CNN	FC	PCEDNet
Time t_K	**0.015**	0.16	2.28*	1.32*	0.062*	0.023	0.043 (0.066)	0.0024 (0.0254)	**0.0026** (0.0256)
Energy E_K	**1.36**	14.79	592.87*	345.77*	15.63*	4.24	3.87 (8.11)	0.22 (4.46)	**0.23** (4.47)

Table 13. Times t_K (2^{nd} row) and processing unit energy consumption E_K (3^{rd} row) required for classifying 1K points with the different methods (1^{st} row). PCPNet and ECNet are run on an NVIDIA TITAN Quadro RTX 6000 GPU, and PIENet is run on an NVIDIA TITAN X GPU. The times and energy consumption for PIENet are computed from the statistics provided in [Wang et al. 2020].

- Learn from hundreds of samples

Interactive tools for digital painting tools

[D. Vanderhaeghe]

- Context
  - Digital painting tools for artistic creation
  - ANR JCJC Structures: hierarchical motion representation for stylized rendering
- Objectives
  - Develop interactive tools to help painterly animation
- Challenges
  - Detect "structures" in strokes
  - Define user-centered tools

Conclusion

- We are mostly users of deep learning tools
- We seek for fast and interactive approaches
- We develop our own implementations for better performances

Thanks !

Q&A