Deep Learning Approach to Charged Particle Track Pattern Recognition

HEP Trk.X Kickoff Meeting, Fermilab, 1-3 Nov 2016

Jean-Roch Vlimant

Approach

Learn the parameters of the track each hit belong to

- Very similar to an Hough Transform approach : just learning the transformation instead of imposing it
- Similar to associative memory track parameters lookup

Hough Transform

Hough algorithm

Discretised maximum likelihood optimisation over

$$L(n|\{x_i\}) = \sum_i \int \mathrm{d}n\, \delta(d(n, x_i))$$

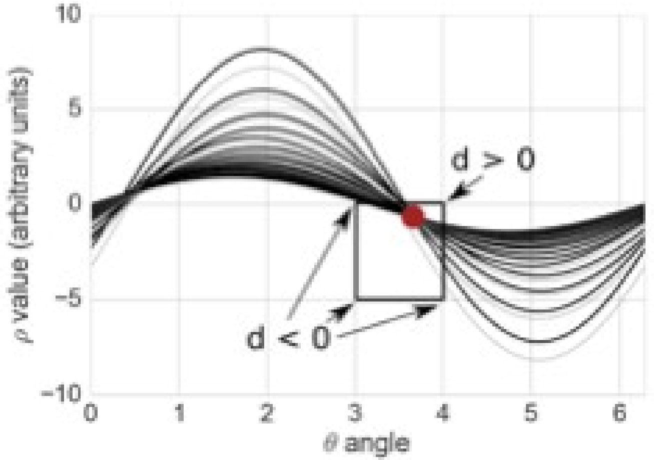
where d is the distance measure of track to hit.
Typically carried out as

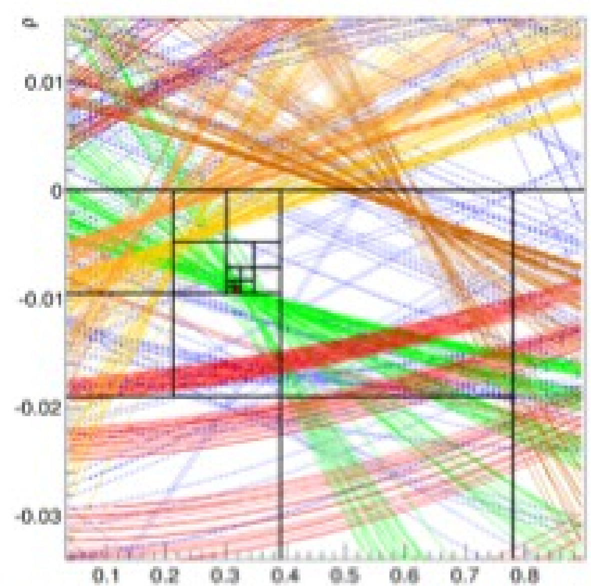
- grid search
- *Fast Hough* bisecting each dimension

over small volumes dn of the parameter space evaluating only the signs of d on the edges.

Refinements

- Weighting of hits versus tracks e.g. on distance d or prior distributions
- Priorisation of search areas
- Overlapping volumes





Oliver Frost on behalf of the Belle II collaboration | DESY | 2016-02-22 | Page 4 / 23

Connecting The Dots 2016 https://indico.hephy.oeaw.ac.at/event/86/

In a Nutshell

Input

Pixel signal

ANN

Target

Track parameter of track through that pixel

- Data is represented in a 1D vector indexing pixels over the barrels in the natural ordering.
- Size N = sum(layer size)
- Binary signal On/Off

- The output has the same size as the input

What Was Tried

- Stack of 4 pixel modules, small number of tracks (4 to 10 due to small setup) small number of pixel (fully connected network size limitation) gave 60-70% hit matching accuracy.
- Train on the track “rank” in the event. Much harder problem than finding track parameters.
- Regression on normalized curvature of the track. More reasonable approach.
- Helix data generator in a simple barrel pixel only detector. Does not require to store a huge amount of data, it's generated on-demand.
- Fully connected layers so far. Nothing else because of the un-conventional layer layout.
- Training with masking pixel with no signal to avoid “training on zeros”

Path Forward

- Step back and demonstrate in lower dimensionality
- Add other track parameters (eta, phi, rho) to target
- Address the size of the model with model parallelisation (tensorflow)
- Address training time with data parallelisation (mpi_learn, spark, see next slides)
- Clustering in the track parameter space. Combined approach "a la" scene labeling (see next slides)

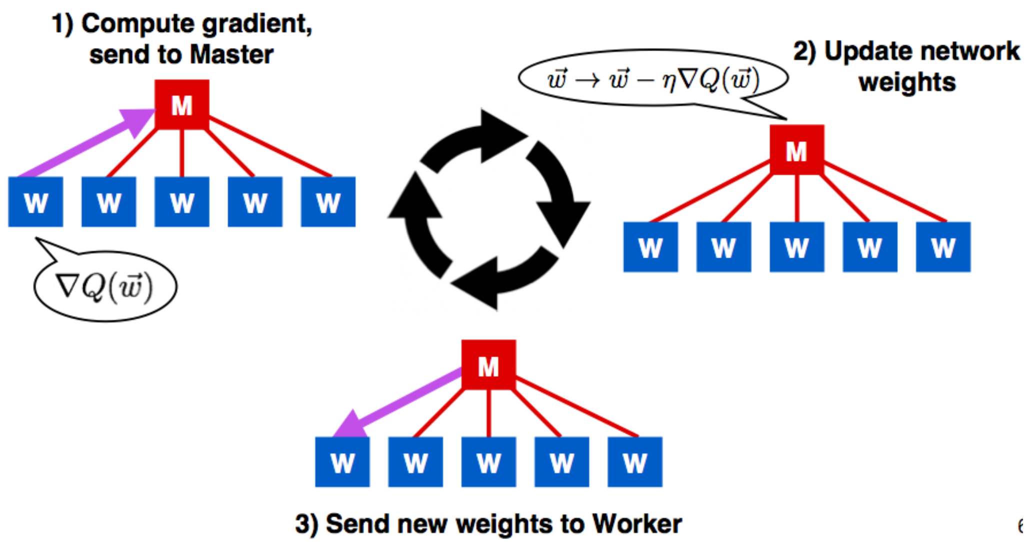
Scene Labeling

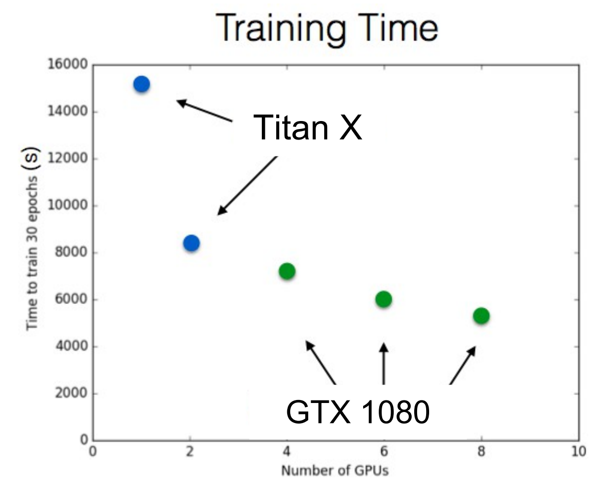
Farabet et al. ICML 2012, PAMI 2013

Scene Labeling

From talk of LeCunn at CERN

Distributed Learning





- Deep learning with elastic averaging SGD https://arxiv.org/abs/1412.6651
- Revisiting Distributed Synchronous SGD https://arxiv.org/abs/1604.00981
- Implementation with Spark and MPI for the Keras framework https://keras.io/