Muon trigger for mobile phones

on behalf of the CRAYFIS collaboration

For the 22nd International Conference on Computing in High Energy and Nuclear Physics,
CHEP 2016, October 10-14, hosted by SLAC and LBNL

Maxim Borisyak[1,2], Mikhail Usvyatsov[2,3,4], Michael Mulhearn[5], Chase Shimmin[6] and Andrey Ustyuzhanin[1,2]

[1] National Research University Higher School of Economics
[2] Yandex School of Data Analysis
[3] Moscow Institute of Physics and Technology
[4] Skolkovo Institute of Science and Technology
[5] University of California, Davis
[6] University of California, Irvine

CRAYFIS experiment

Cosmic RAYs Found In Smartphones Experiment

CRAYFIS experiment proposes usage of private mobile phones for observing Ultra-High Energy Cosmic Rays:

- high energies: $> 10^{18}$ eV;
- distributed world-wide observatory;
- mobile phone's camera as cosmic rays detector;
- cluster of mobile phones as intensive air shower detector.

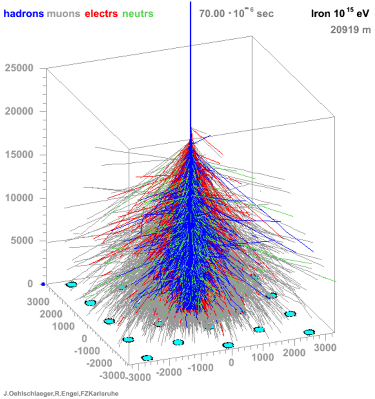


Illustration of an intensive air shower produced by iron ion, 1 PeV, CORSIKA simulation, by J. Oehlschläger and R. Engel.

Challenges

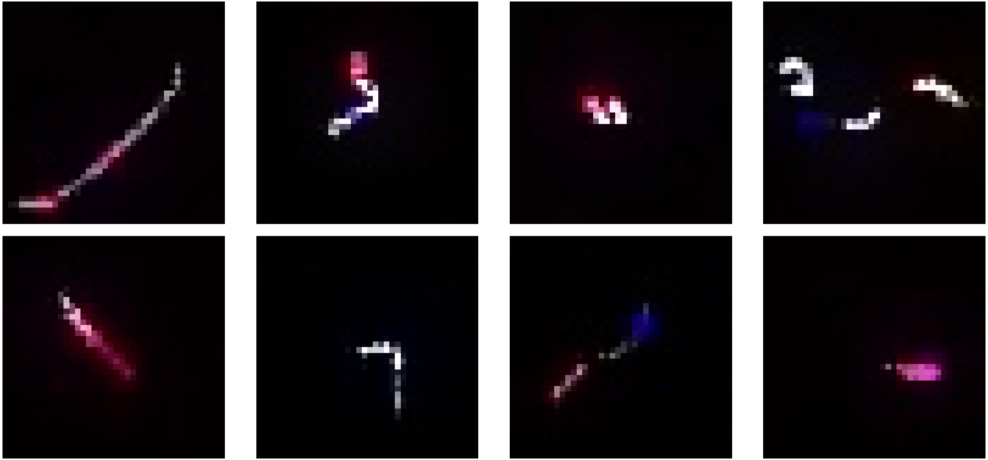
CRAYFIS:

- low event rate is expected:
  - background cosmic rays: less than once per minute;
  - UHECR: less than once per year;
- an intensive air shower from UHECR occurs in less than microseconds;
- high frame rate is required (~ 10 Hz).

Mobile phones:

- a typical mobile phone has relatively low computational power and storage space;
- cosmic ray trigger is a crucial part of the experiment.

Photon Events Examples



Examples of photon events. Images obtained under exposure to a radioactive source. Images are 40×40 pixels in size.

Muon Trigger

Main assumption

Minimally ionizing particles leave traces with brightness comparable to the level of intrinsic camera noise.

Trigger strategies:

- simple strategies are not sensitive to muon events, thus result in either:
  - large pollution rates: consume a lot of storage space;
  - large loss rates: low sensitivity;
- shape recognition is required.

Muon Trigger

Visual Pattern Recognition

Convolution Neural Networks:

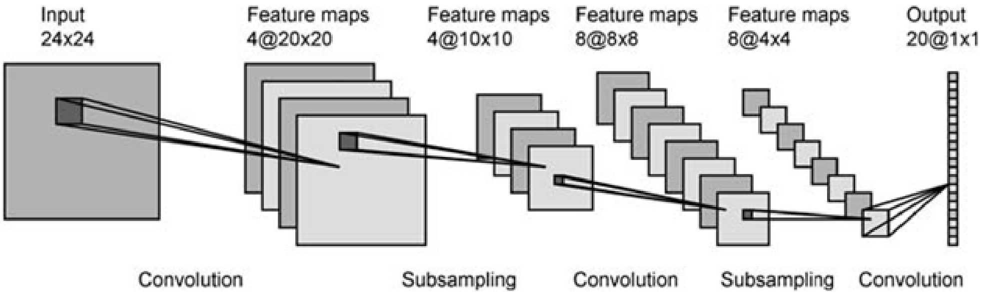
- state-of-the-art algorithms for wide range of visual pattern recognition tasks;
- require considerable computational resources (typically, powerful GPUs).

Alternative approaches:

- various domain-specific algorithms;
- **Viola-Jones cascades** [Viola and Jones, 2004].

Convolutional Neural Networks

Convolutional Neural Networks is a generic model for visual pattern recognition.



A CNN is represented as a set of convolutional filters organized in layers. Each filter is convolved with the result of the previous layer to obtain more high-level representation of image.

Cascade Architecture for Convolutional Networks

Suggested approach
Adopt cascade architecture for Convolutional Neural Networks.

Cascade architecture features:

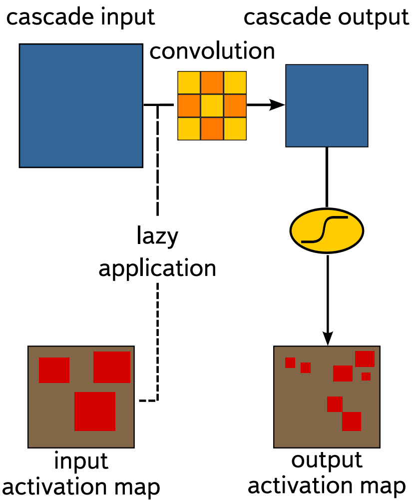
- cascades equivalent to a sequence of triggers;
- first cascades are to discard most obviously noisy areas on early stages;
- last cascades are to refine region of interest (particle traces).

Convolutional Neural Networks:

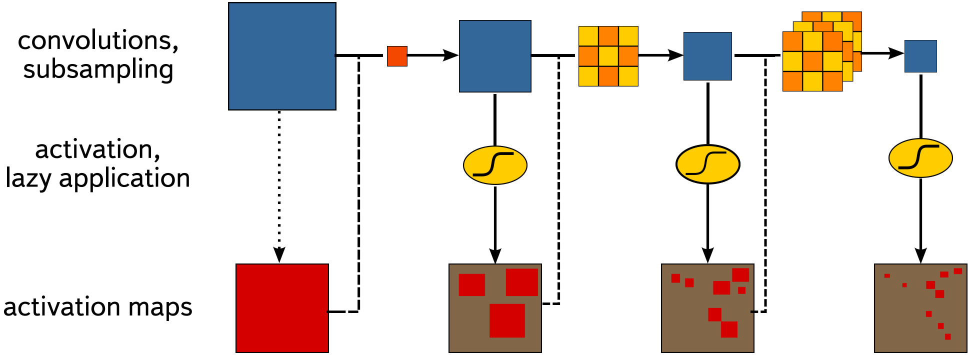
- globally optimized;
- high-level representation from one layer is passed to the subsequent one.

Single Cascade Schematics

- convolutions are computed only on activated areas ('lazy' application);
- resulting image is then subsampled;
- updated activation map is computed.

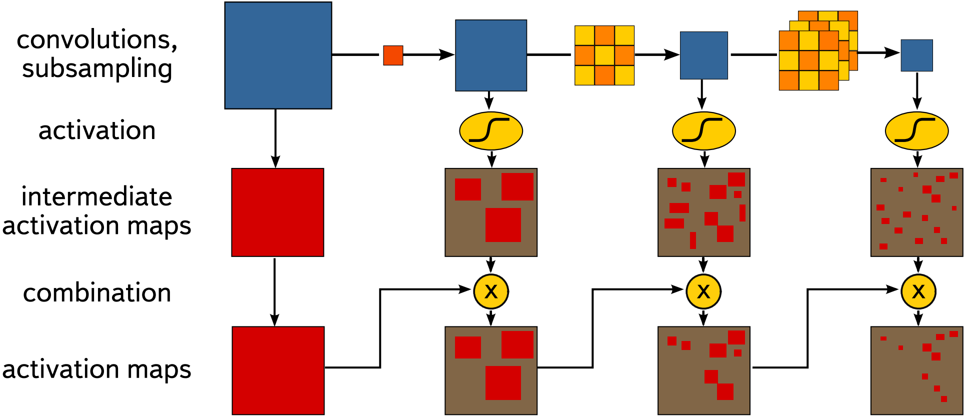


Cascade Network Schematics



Cascade Convolutional Neural Network schematic. The first row represents a conventional Convolutional Neural Network: a series of convolution layers, blue squares denote stages of image precessing. Red areas on the squares in the bottom row denote activated areas, dashed lines — 'lazy' applications (computations only on activated areas).

Cascade Network Training



Cascade Convolutional Neural Networks are trained in a similar way as conventional Convolutional Neural Networks. Instead of 'lazy' application convolution filters are applied to the whole image, nevertheless, activation maps are combined with the ones from the previous cascade.

Cascade Network Loss Function

$$\mathcal{L} = \sum_i \alpha^i \mathcal{L}^i_{\mathrm{cascade}} + \beta \sum_i \mathcal{L}^i_{\mathrm{complexity}}$$

- $\mathcal{L}^i_{\mathrm{cascade}}$ — i-th cascade's loss function, weighted cross-entropy;
- $\mathcal{L}^i_{\mathrm{complexity}}$ — i-th cascade's approximated computational cost;

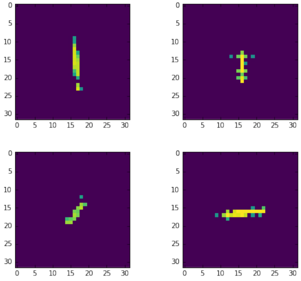
Cascade losses

- penalties for activation on noise gradually increase from cascade to cascade;
- penalty for missing target is constant and higher than for activation on noise.

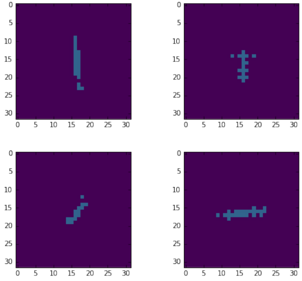
Proof of the concept

Data

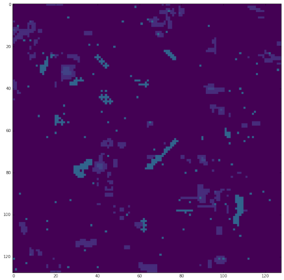
1. starting point: images from mobile phone exposed to a radioactive source;
2. particle traces are selected, brightness artificially lowered;
3. gloom traces are superimposed on background.



1. original traces



2. low-brightness traces



3. superimposed on background

Baseline

- simple strategy (brightness cut);
  - signal efficiency: 1.0;
  - background rejection: 0.01;
- no signal efficiency-background rejection trade-off;
- extremely fast (1 operation per pixel);

Results

- 4 cascades:

$$1 \text{ filter } 1 \times 1 \longrightarrow 1 \text{ filter } 3 \times 3 \longrightarrow 3 \text{ filters } 3 \times 3 \longrightarrow 6 \text{ filters } 3 \times 3$$

- total complexity: $< 0.04 \times$ complexity of the full CNN.

complexity relative to the simple strategy	1.4			2.0		
signal efficiency	0.90	0.95	0.99	0.90	0.95	0.99
background rejection	0.60	0.39	0.12	0.65	0.44	0.15

Summary

Summary

CRAYFIS experiment:

› aimed for Ultra-High Energy Cosmic Rays;

› efficient trigger on-device strategy plays crucial role.

Cascade Convolutional Neural Networks:

› cascade architecture allows to discard noise on early stages;

› considerable decrease in total computational cost relative to conventional CNNs.

Proof of the concept:

› total complexity: $1.4 \times$ complexity of the simple strategy, 0.02 of CNN;

› signal efficiency: 0.9; background rejection: 0.6.

References I

LeCun, Y., Jackel, L., Bottou, L., Cortes, C., Denker, J. S., Drucker, H., Guyon, I., Muller, U., Sackinger, E., Simard, P., et al. (1995).
Learning algorithms for classification: A comparison on handwritten digit recognition.
Neural networks: the statistical mechanics perspective, 261:276.

Viola, P. and Jones, M. J. (2004).
Robust real-time face detection.
International journal of computer vision, 57(2):137--154.

Whiteson, D., Mulhearn, M., Shimmin, C., Cranmer, K., Brodie, K., and Burns, D. (2016).
Searching for ultra-high energy cosmic rays with smartphones.
Astroparticle Physics, 79:1--9.

CRAYFIS collaboration

University of California Irvine:

- Daniel Whiteson;
- Chase Shimmin;
- Homer Strong;
- Jay Karimi;
- Kyle Brodie;
- Rob Porter;
- Eric Albin;
- Emma He;
- Zichao Ziliver Yuan;
- Jeff Swaney;

University of California Davis:

- Michael Mulhearn;
- Dustin Burns;
- Brandon Buonacarsi;

New-York University:

- Kyle Cranmer;

Yandex School of Data Analysis:

- Andrey Ustyuzhanin;
- Maxim Borisyak;
- Mikhail Usvyatsov;

National Astronomical Observatory of China:

- Jianrong Deng;

Unaffiliated:

- Danielle Cranmer;
- Jodi Goddard.

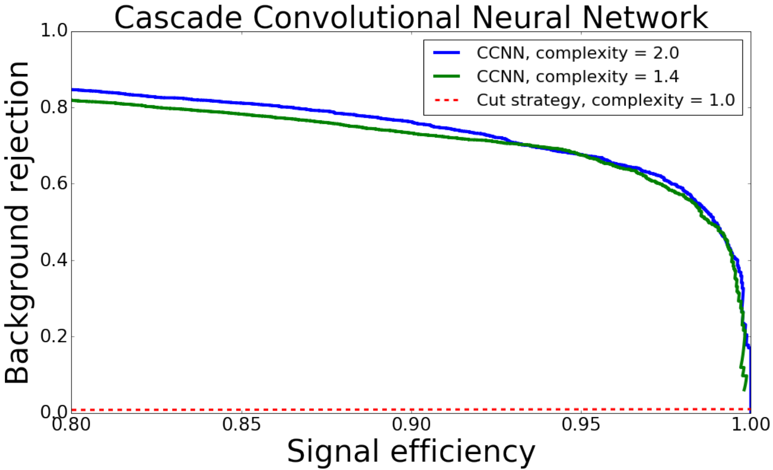
Backup

Cascade Network Loss Function

$$
\begin{aligned}
\mathcal{L} &= \sum_i \alpha^i \mathcal{L}^i_{\text{cascade}} + \beta \sum_i \mathcal{L}^i_{\text{complexity}};\\
\mathcal{L}^i_{\text{cascade}} &= \sum_j \gamma^i (1 - y_j) \log(1 - p^i_j) + y_j \log p^i_j;\\
\mathcal{L}^i_{\text{complexity}} &= \sum_j F^i (1 - y_j) p^i_j;
\end{aligned}
$$

- γ^i — penalty for passing noise;
- p^i_j — i-th cascade activation on j-th window;
- y_j — presence of particle trace in j-th window (target);
- F^i — total size of filters in i-th cascade (computational cost).

Experiment



Cascade Convolutional Neural Network
CCNN, complexity = 2.0
CCNN, complexity = 1.4
Cut strategy, complexity = 1.0
Background rejection
Signal efficiency
1.0
0.8
0.6
0.4
0.2
0.0
0.80
0.85
0.90
0.95
1.00