LABORATORIUM FÜR HOCHENERGIEPHYSIK
LHEP
UNIVERSITÄT BERN

UNIVERSITÄT BERN

AEC
ALBERT EINSTEIN CENTER
FOR FUNDAMENTAL PHYSICS

# ATMic: Fast $4\pi$ tracking based on GPUs and its framework

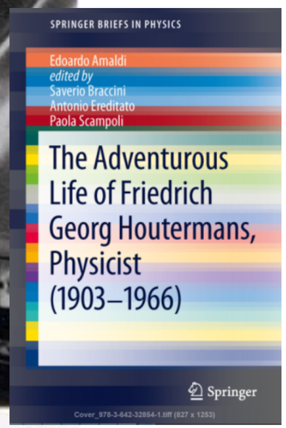

Akitaka Ariga (Senior staff)

Albert Einshtein Center for Fundamental Physics, Laboratory for High Energy Physics,

University of Bern

# Scanners in Bern some time ago

F.G. Houtermans im Kreise seiner Scannerinnen im Physikalischen Institut Bern 1955/56

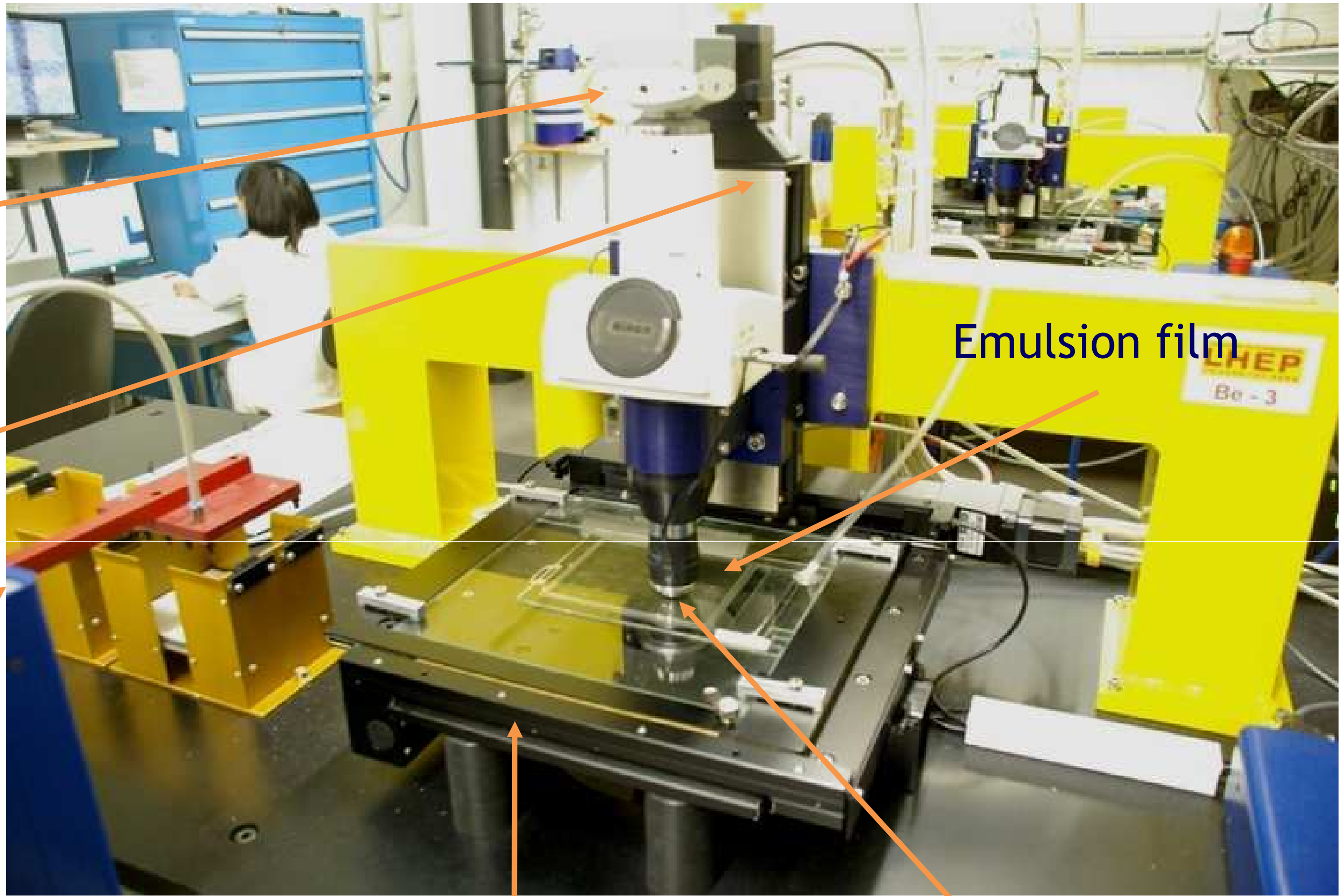


# Conventional system : OPERA Microscope

CMOS camera
1280×1024 pixel
256 gray levels
376 frames/sec
(Mikrotron MC1310)

Z stage (Micos)
0.05 μm nominal
precision

Automatic Plate Changer




XY stage (Micos)
0.1 μm nominal
precision

objective (Dry 50× NA 0.95)

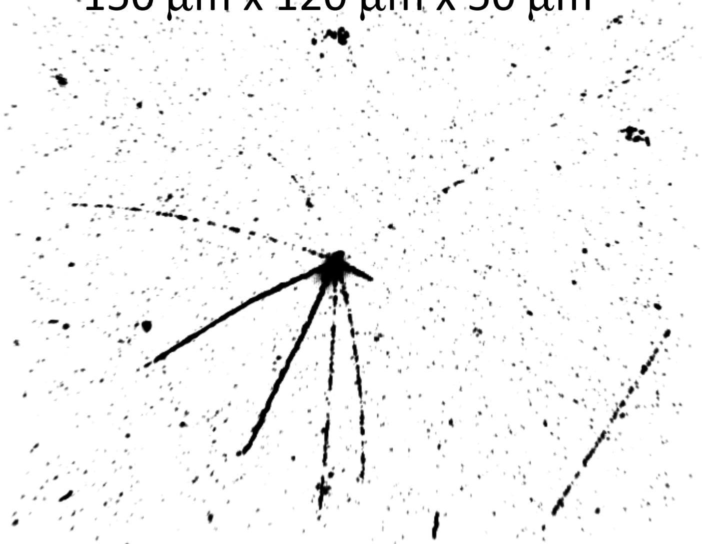
20 min / $cm^2$

# Limitation of conventional system

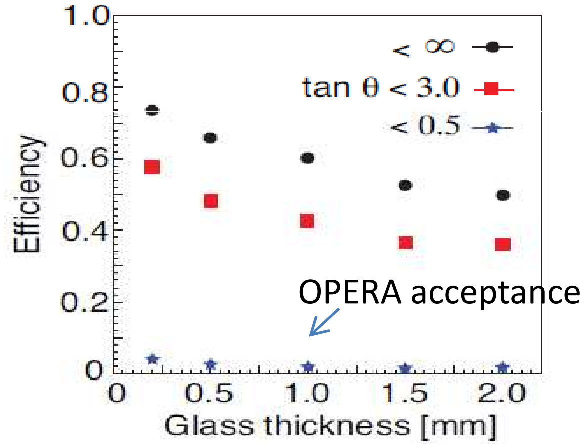
- Small angular acceptance
  - θ < 0.5
  - For AEgIS, we need 4π tracking!
- Tracking tuned for penetrating particles
  - Limited information can be extracted
  - Scattered track cannot be followed
  - Short tracks which start and stop in emulsion cannot be detected
- Less flexibility
  - Development beyond OPERA is difficult

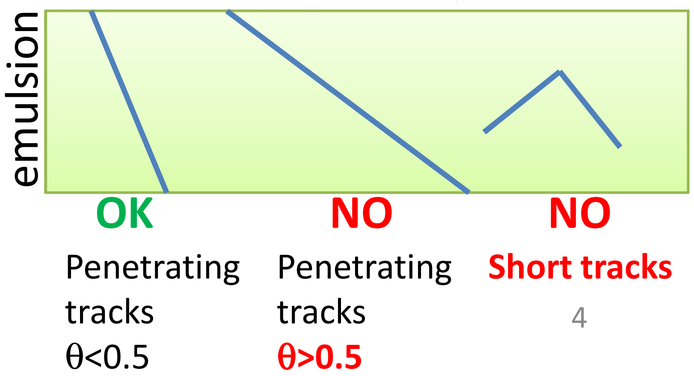
→New Framework : **ATMic**

150 μm x 120 μm x 50 μm



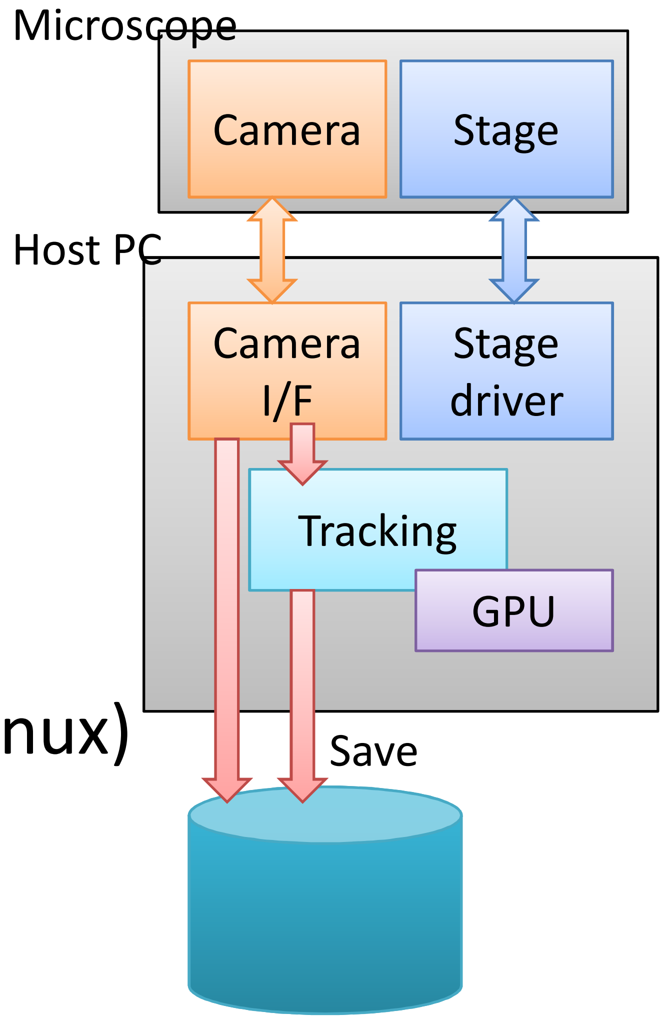
Antiproton detection efficiency



# ATMic

- A framework for general purpose scanning
  - Data classes
  - Stage driver
  - Camera interface
  - Tracking
  - Reconstruction
  - Display (Virtual microscope)
  - GPU interface
- ROOT-base program
- OS independent (Windows or Linux)



# ATMic classes

- All classes are ROOT classes
- Well documented
  - Transparent for users

- Capable to accommodate several tracking algorithms

ROOT

Quick Links: | ROOT Homepage | Class Index | Class Hierarchy

ATMic

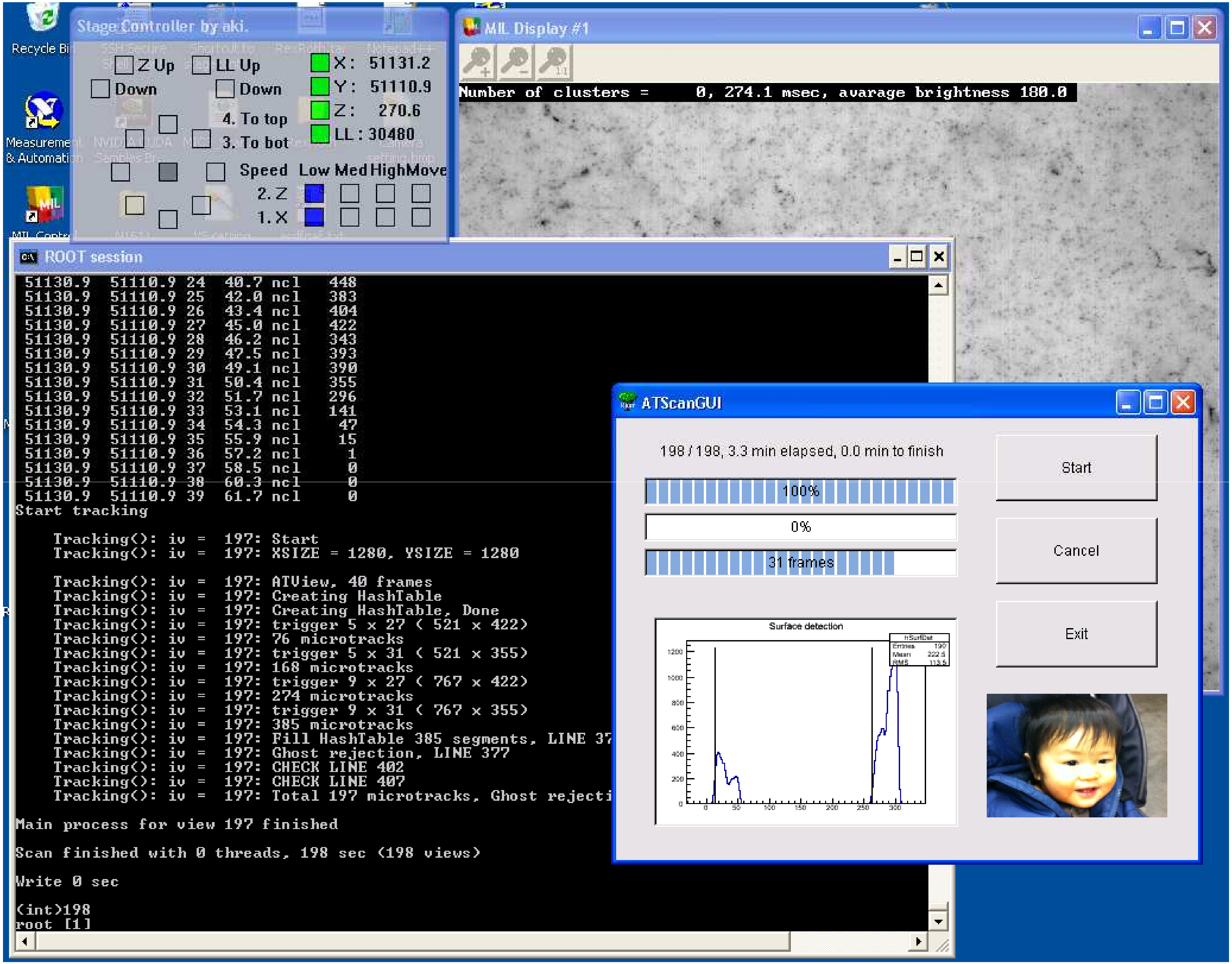
## Class Index

**Modules**

**Jump to**

A ATB ATBasetrackV ATC ATClu ATClusters ATCu ATF ATH
ATImageDisplayH ATImageDisplayO ATIn ATL ATLine3DC ATM
ATSegm ATSegmentC ATT ATTracki ATTrackingC
ATVer ATVi ATViewH ATViewP ATViews G

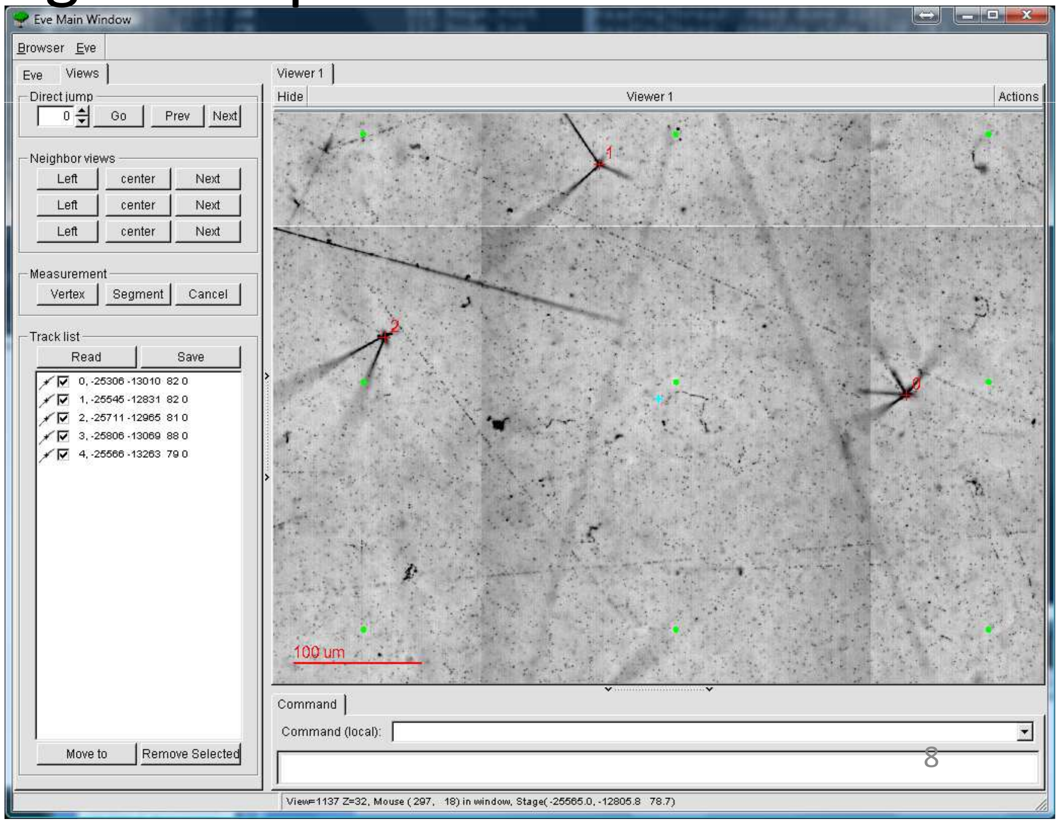
| | |
|---|---|
| ATAffine2D | 2D affine parameters |
| ATBasetrack | basic Segment class |
| ATBasetrackV | basic Segment class |
| ATClonesArray | Container of ATObject |
| ATCluster | Cluster class |
| ATClustersArray | Base class of data |
| ATCuda | |
| ATFrame | 8 bit image |
| ATHashTable | ATHashTable |
| ATHashTable2 | ATHashTable2 |
| ATImageDisplay | |
| ATImageDisplayHandler | |
| ATImageDisplayOverlay | a class for draw overlays, i.e. scale bars. |
| ATIntegerDialog | Input GUI for integer number. |
| ATLine3D | 3D line |
| ATLine3DC | 3D line |
| ATMic | |
| ATNoTracking | Tracking for G1 |
| ATObject | Base class of data |
| ATPred | prediction |
| ATSegV | basic Segment class |
| ATSegVC | basic Segment class |
| ATSegment | basic Segment class |
| ATSegmentC | Segment with clusters |
| ATTrackV | Track class for ATSegV |
| ATTracking | Tracking abstract |
| ATTrackingCluster | Tracking with clusters |
| ATTrackingCluster2 | Tracking with clusters |
| ATTrackingG1 | Tracking for G1 |
| ATVMDataManager | Virtual Microscope data manager |
| ATVector3 | 3D Vector class like TVector3 (specially made for CUDA |


Recycle Bi
Stage Controller by aki.
Z Up
LL Up
Down
Down
X : 51131.2
Y : 51110.9
Z : 270.6
LL : 30480
4. To top
3. To bot
Measureme
& Automati
Speed
Low
Med
High
Move
2. Z
1. X
MIL Display #1
Number of clusters = 0, 274.1 msec, avarage brightness 180.0
ROOT session
51130.9 51110.9 24 40.7 ncl 448
51130.9 51110.9 25 42.0 ncl 383
51130.9 51110.9 26 43.4 ncl 404
51130.9 51110.9 27 45.0 ncl 422
51130.9 51110.9 28 46.2 ncl 343
51130.9 51110.9 29 47.5 ncl 393
51130.9 51110.9 30 49.1 ncl 390
51130.9 51110.9 31 50.4 ncl 355
51130.9 51110.9 32 51.7 ncl 296
51130.9 51110.9 33 53.1 ncl 141
51130.9 51110.9 34 54.3 ncl 47
51130.9 51110.9 35 55.9 ncl 15
51130.9 51110.9 36 57.2 ncl 1
51130.9 51110.9 37 58.5 ncl 0
51130.9 51110.9 38 60.3 ncl 0
51130.9 51110.9 39 61.7 ncl 0
Start tracking
Tracking(): iv = 197: Start
Tracking(): iv = 197: XSIZE = 1280, YSIZE = 1280
Tracking(): iv = 197: ATView, 40 frames
Tracking(): iv = 197: Creating HashTable
Tracking(): iv = 197: Creating HashTable, Done
Tracking(): iv = 197: trigger 5 x 27 ( 521 x 422)
Tracking(): iv = 197: 76 microtracks
Tracking(): iv = 197: trigger 5 x 31 ( 521 x 355)
Tracking(): iv = 197: 168 microtracks
Tracking(): iv = 197: trigger 9 x 27 ( 767 x 422)
Tracking(): iv = 197: 274 microtracks
Tracking(): iv = 197: trigger 9 x 31 ( 767 x 355)
Tracking(): iv = 197: 385 microtracks
Tracking(): iv = 197: Fill HashTable 385 segments, LINE 37
Tracking(): iv = 197: Ghost rejection, LINE 377
Tracking(): iv = 197: CHECK LINE 402
Tracking(): iv = 197: CHECK LINE 407
Tracking(): iv = 197: Total 197 microtracks, Ghost rejecti
Main process for view 197 finished
Scan finished with 0 threads, 198 sec (198 views)
Write 0 sec
(int)198
root [1]
ATScanGUI
198 / 198, 3.3 min elapsed, 0.0 min to finish
100%
0%
31 frames
Start
Cancel
Exit
Surface detection

# Virtual microscope

- ROOT & OpenGL base programming
- Dynamic data management
  - Necessary to deal Tera-byte scale data
  - Multithreads for display / reading data
- Nice also for students and general public

See image display

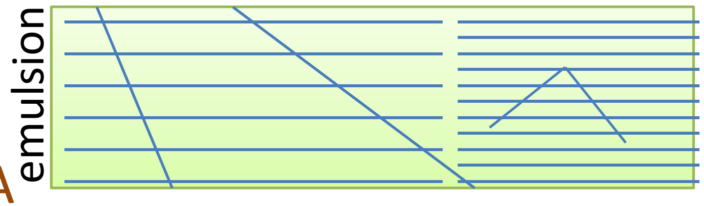


# Fast 4 $\pi$ tracking

- General purpose tracking
  - 8 times larger angular acceptance in solid angle than OPERA

| | AEgIS | OPERA |
|---|---|---|
| acceptance in $\theta$ (rad) | $\pi$ | 0.5 |
| Acceptance in steradian | 12.6 ($4\pi$) | 1.5 |

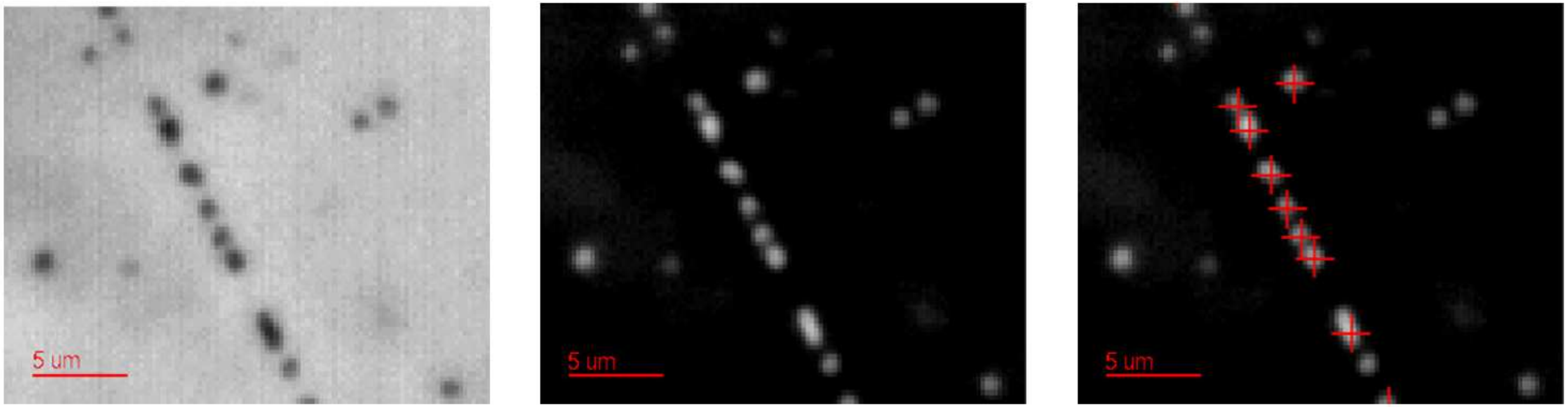
- Detection of short tracks
  - finer sampling (16->**40** frames/view)
  - ~~quick algorithm~~
  - start and stop points information
  - >10 times more computing than OPERA



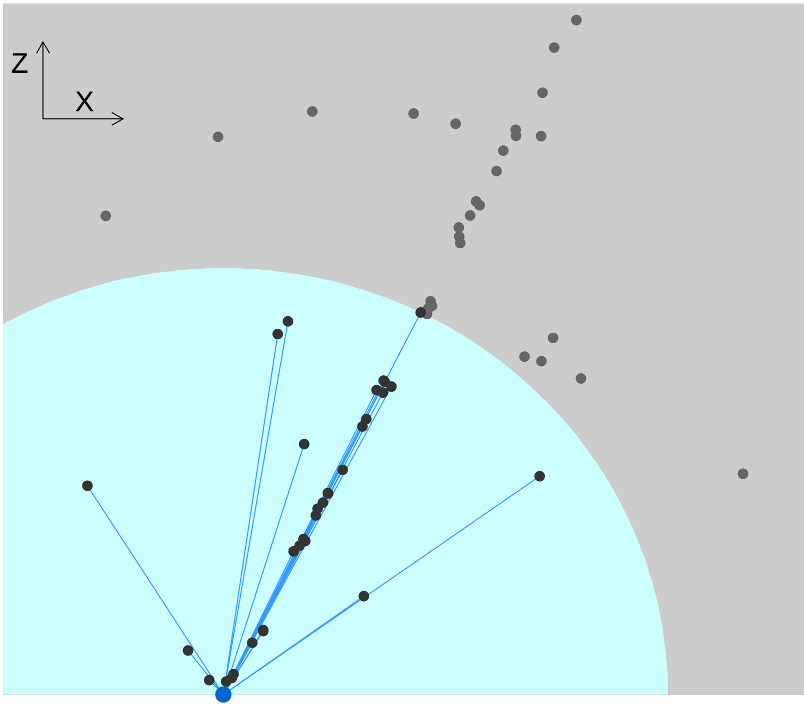

- → 2 order of magnitude larger computation is needed when it is compared OPERA algorithm

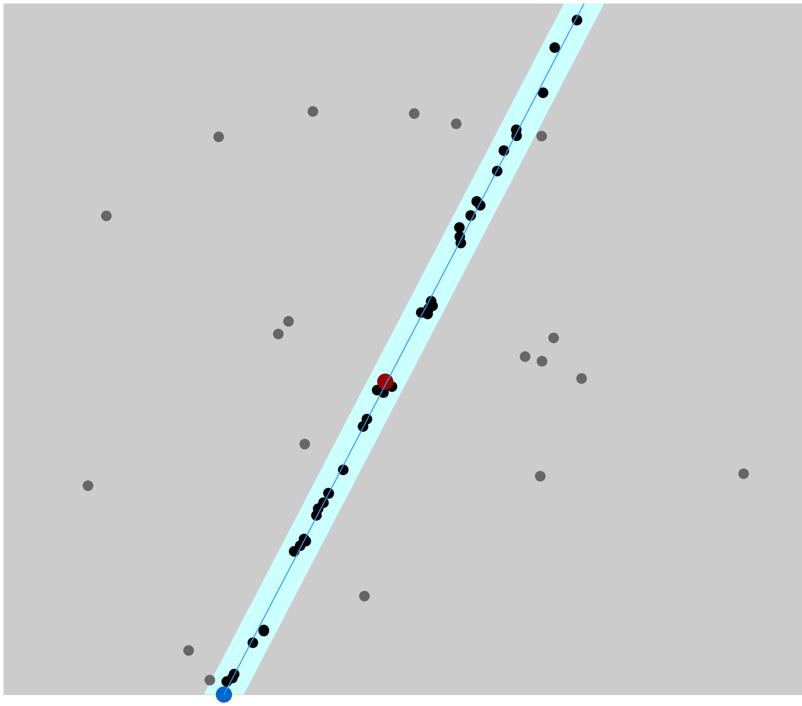
# 3D Filtering

- Pixel by pixel gain control
- BG subtraction
- 3D grain recognition



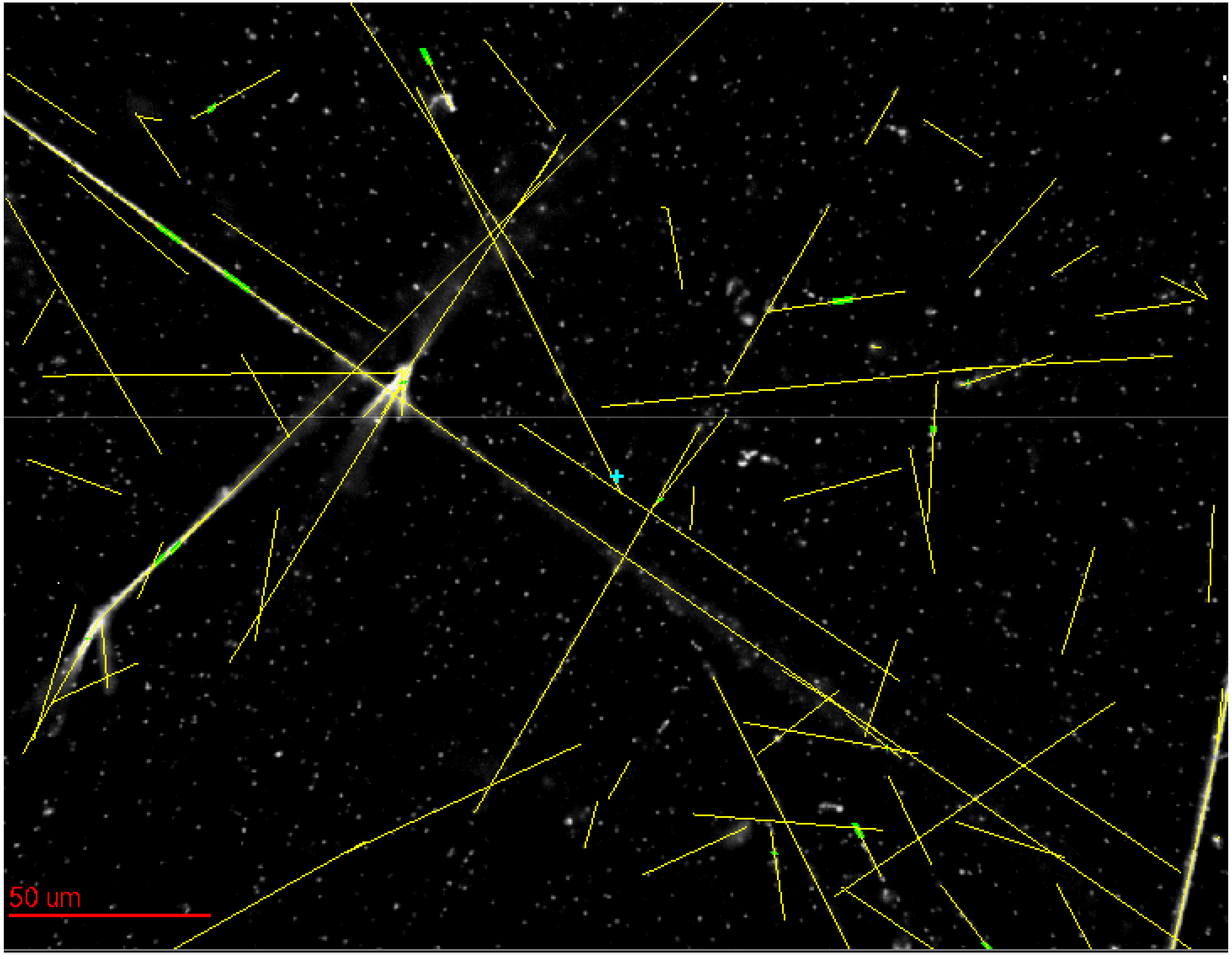
# Tracking algorithm



- Find any 2 grains combination within a certain distance, forming seeds (lines).
- Count number of clusters along the lines. if number of clusters is bigger than a threshold, classify as a track.

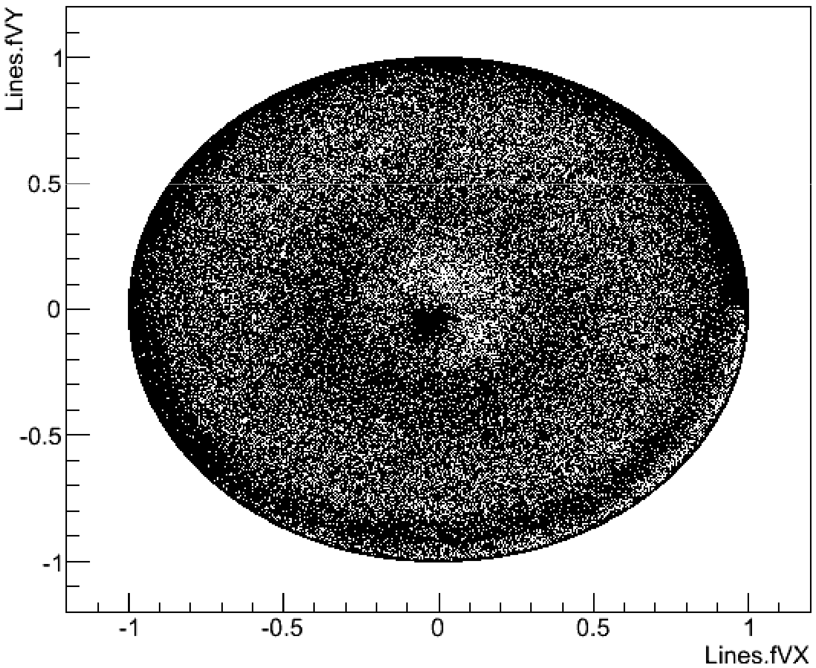
# Example of a view (SEE DISPLAY)



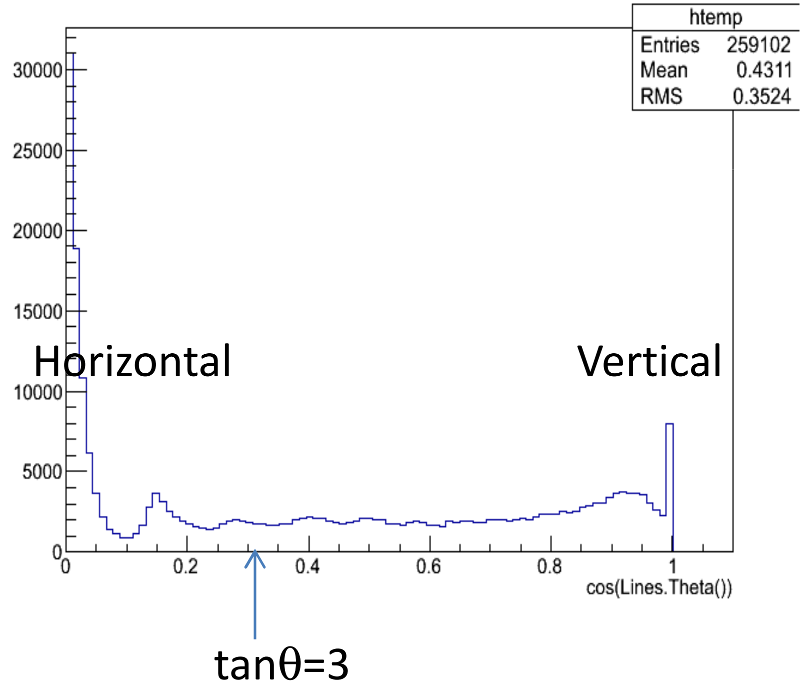

all tracks recognized by Aki are reconstructed as well by the algo.

# Reconstructed angular distributions AEgIS film exposed to antiprotons

**Angle as 3D unit vector**
**VX-VY**



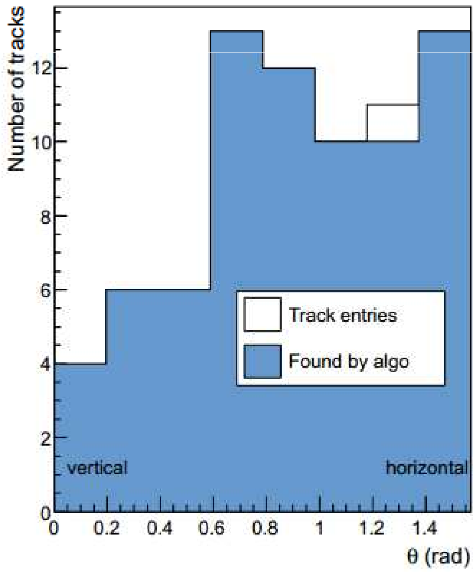

**Cosθ distribution**
**θ : azimuthal angle**

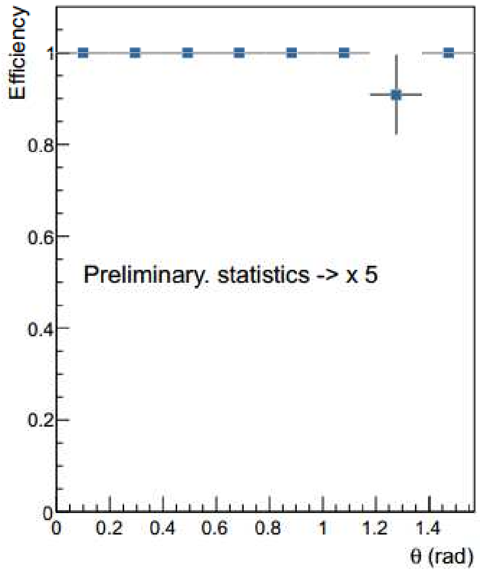



The tracking covers 4 π solid angle

# Efficiency of tracking

- Efficiency has been checked by comparing unbiased manual measurement
  - Tracks longer than 10 microns.
  - Half of tracks are heavily ionizing particles

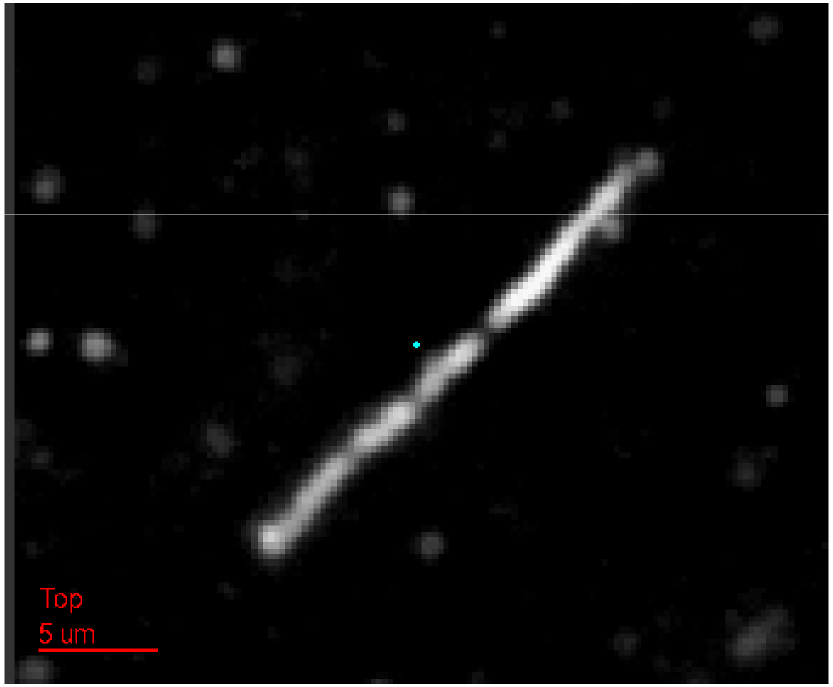


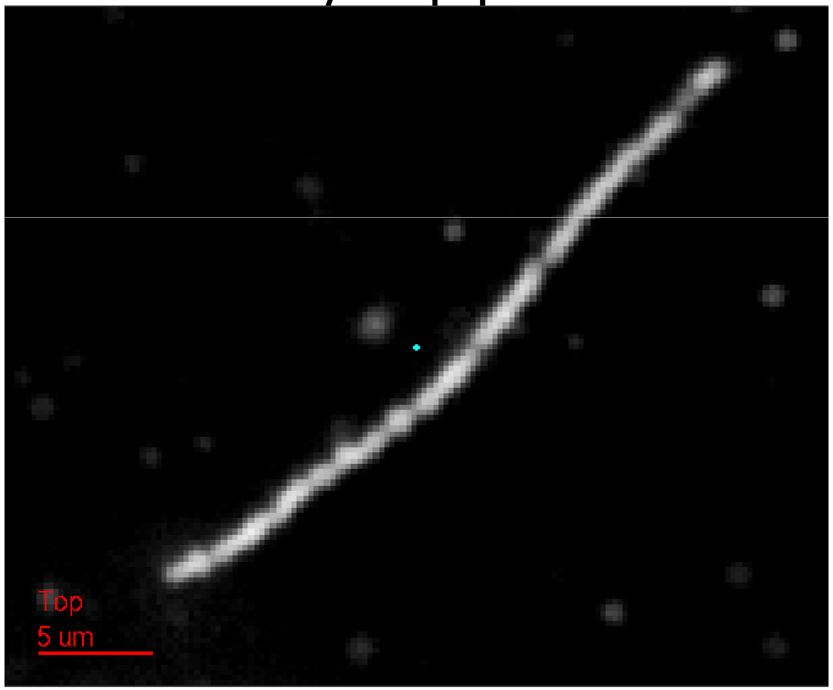

- High tracking efficiency is achieved.

# Example of proton tracks recoiled by neutrons

- Proton tracks by 2.5 MeV neutrons.
- (horizontal tracks are chosen for demo)
- Short tracks are reconstructed with start/stop point




A 30 micron short track is reconstructed




A scattered track is reconstructed as 2 segments

# Reconstruction of emulsion data

- Goal of processing time = 0.2 sec/view (→ ~10 $cm^2$/microscope)
  - Typical data unit “view” = 1280 x 1024 pixels x 40 frames = 52 Mbyte
  - FOV = 300x250 microns
  - image filtering, 3D grain recognition, 3D track reconstruction
- With single thread programming

  ~**10 sec/view** with high spec CPU.
- need factor 50 times faster processing
- → Track reconstruction based on

  **GPU** technology
  - GPU = Graphic Processing Unit
  - Advantage : Parallel processing with

    thousands of processing unit

# Use of GPU, as far as I concerned

- In 2003, use of GPU shaders for tracking
  - Use of alpha blending for tracking
  - Not enough speed… gave up
- In 2006, CUDA library has been released by NVIDIA
  - General purpose library
- In 2008, restarting to use GPUs
  - Suggestion to use GPU to the OPERA collaboration → a few successful implementation. See T. Fukuda, V. Tioukov
- After several interruptions, achieved a reasonable result recently

# Processing machine equipped with 3 GPUs

Fast memory
DDR3 2400, 16GB

Water cooling

Geforece GTX
TITAN x 3
2688 cores, 6GB
memory, 4.5
TFLOPs in each

Strong power supply
1250 W

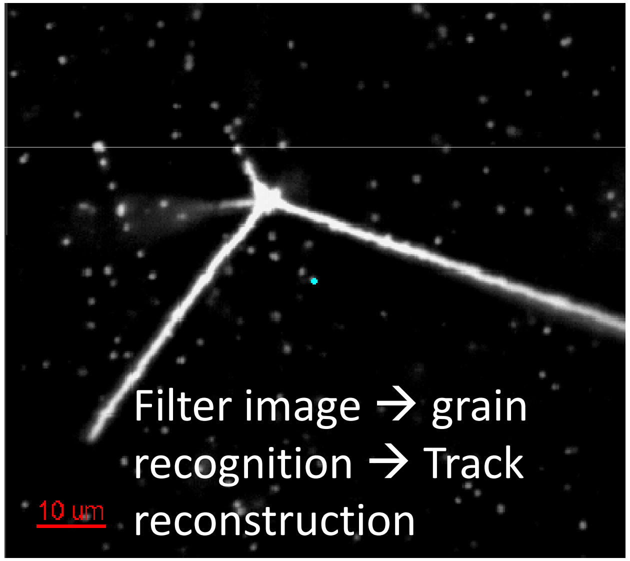
i7- 3930K
6 cores
12 threads
3.2 GHz


# Processing time by single thread

- Goal: 0.2 sec/view
- Same algorithm was implemented on CPU and GPU program.

| | 1 CPU (sec/view) | using 1 GPU (sec/view) | Gain |
|---|---|---|---|
| Image filtering | 0.55 | 0.022 | x25 |
| 3D grain reco | 0.20 | 0.025 | x8 |
| 3D tracking | 5.90 | 0.373 | x16 |

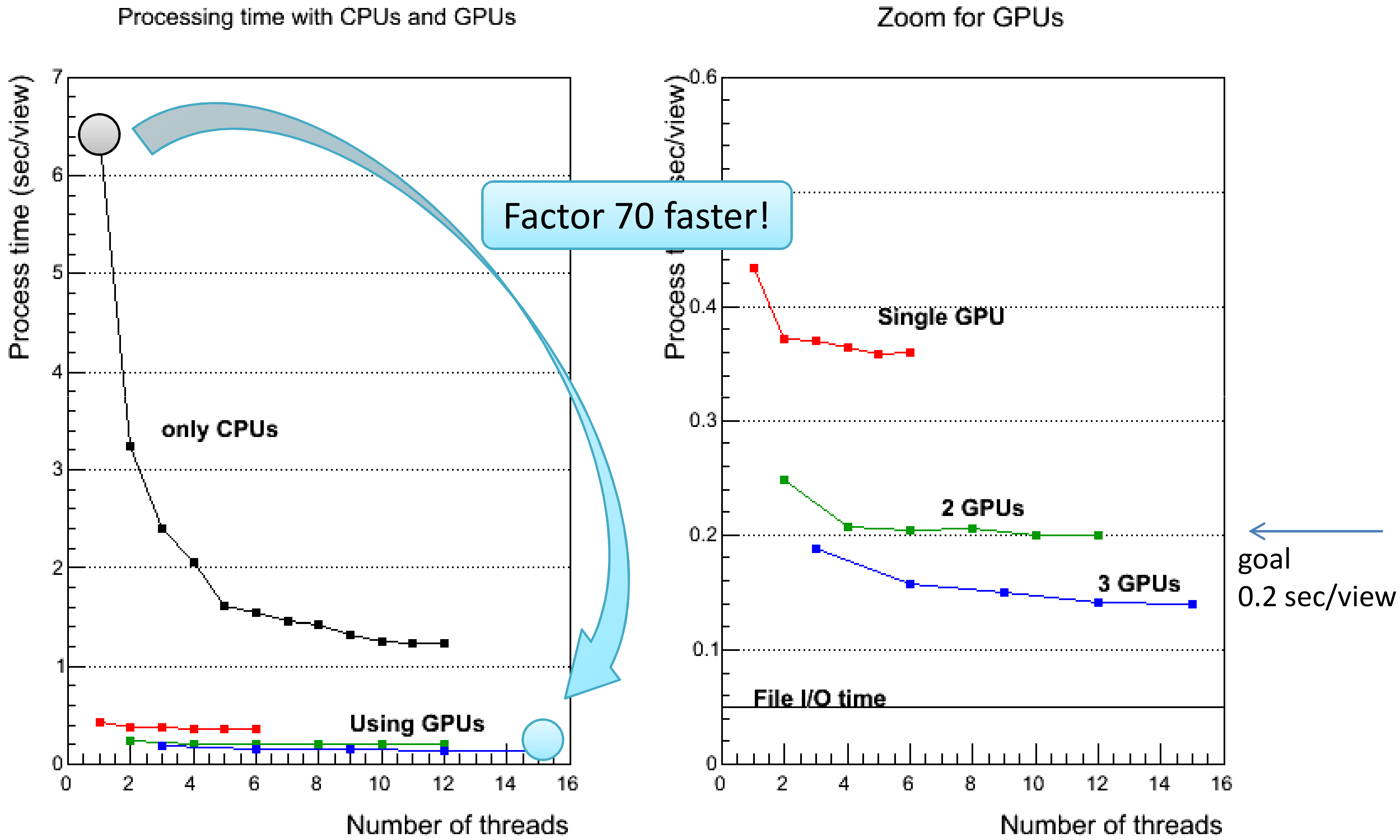
i7- 3930K 6 cores, 12 threads, 3.2 GHz
Geforce GTX TITAN, 2688 cuda cores




Single-CPU-thread and single-GPU is not enough to achieve the goal.
→ **Multi-CPU-thread and multi-GPU programing**

# Processing time with "multithreading" and "multi-GPUs "

(mean of 60 views)




- Multithreading x multi-GPUs allow to reduce tracking time factor 70.

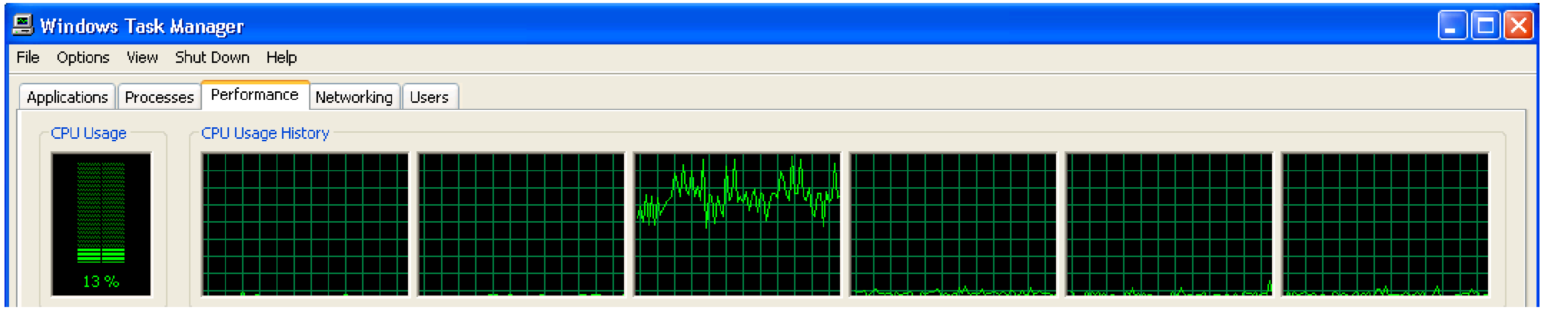
CPU :i7- 3930K 6 cores, 12 threads, 3.2 GHz, GPU : Geforce GTX TITAN, 2688 cuda cores

# Summary

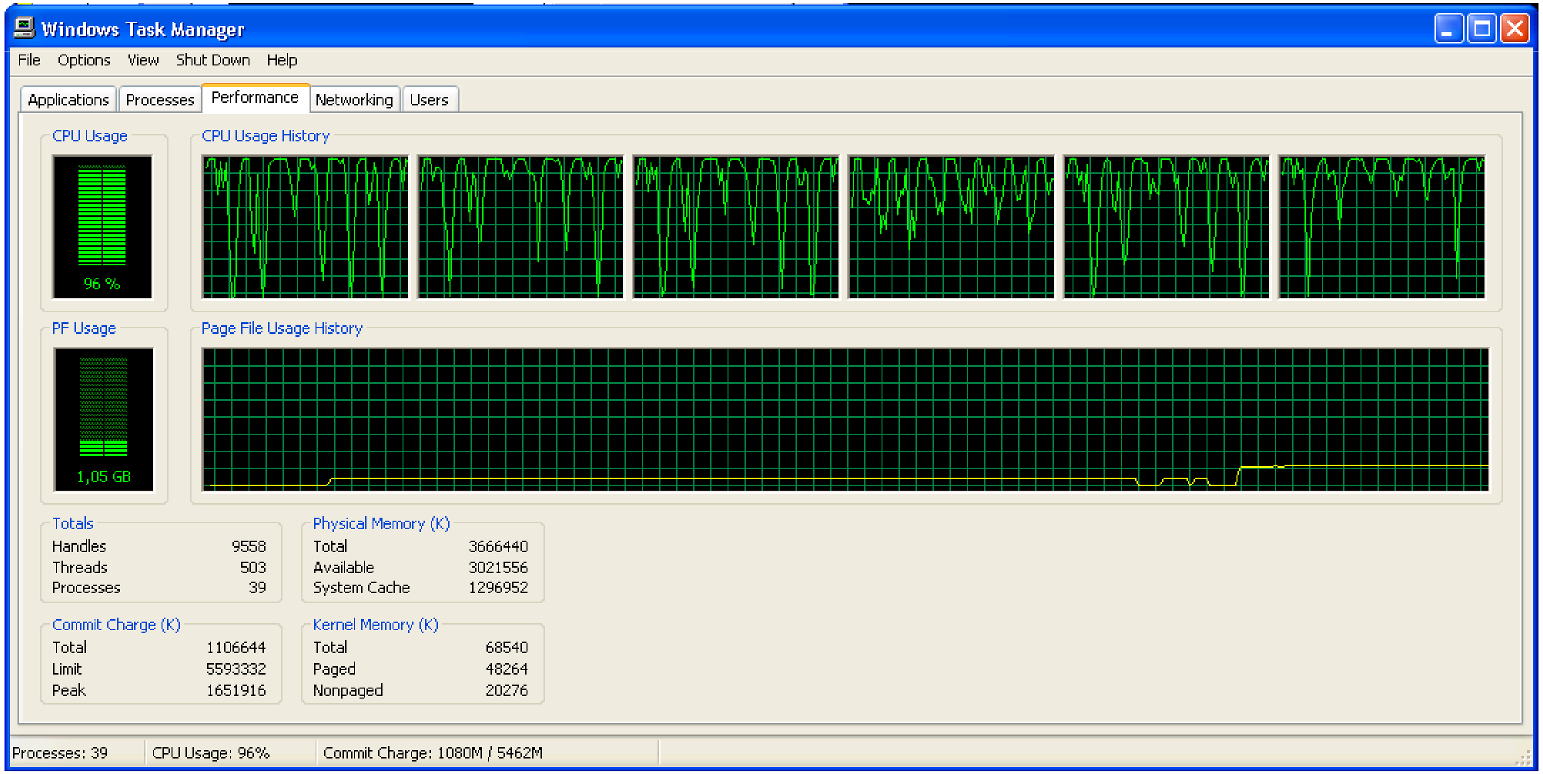
- A new framework for general purpose emulsion scanning/reconstruction is being developed in LHEP Bern
  - From data taking to display
- Fast $4\pi$ tracking is realized with GPU with a reasonable processing time
  - 0.14 sec/view, even with a factor 100 more computation than the OPERA algo.
  - $>10 cm^2/h$
- The system is being applied to AEgIS, neutron projects and the other applications
- Further improvement in processing speed and tracking performance are expected.

# Multi-thread processing

- Single process




- Multi-thread



# Antiproton annihilation in the emulsion
# AEgIS commissioning run (2012)