Optimizing fast convolutional neural networks for identifying long-lived particles in the CMS high-granularity calorimeter

Parker Watts,

Juliette Alimena, Yutaro Iiyama, and Jan Kieseler

September 22, 2021

The Setting

CMS

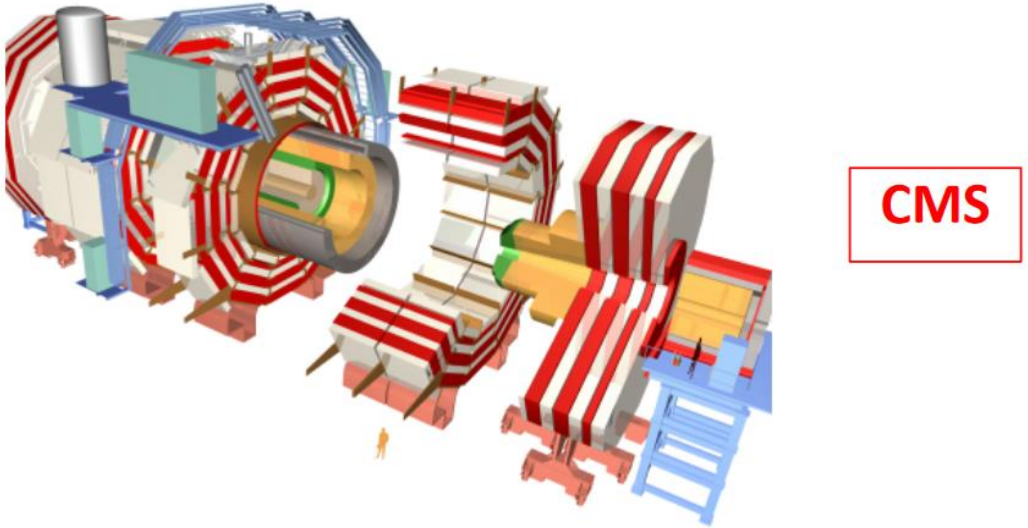
- The Compact Muon Solenoid(CMS) is a detector at the Large Hadron Collider (LHC)
- General purpose detector designed to observe new physics

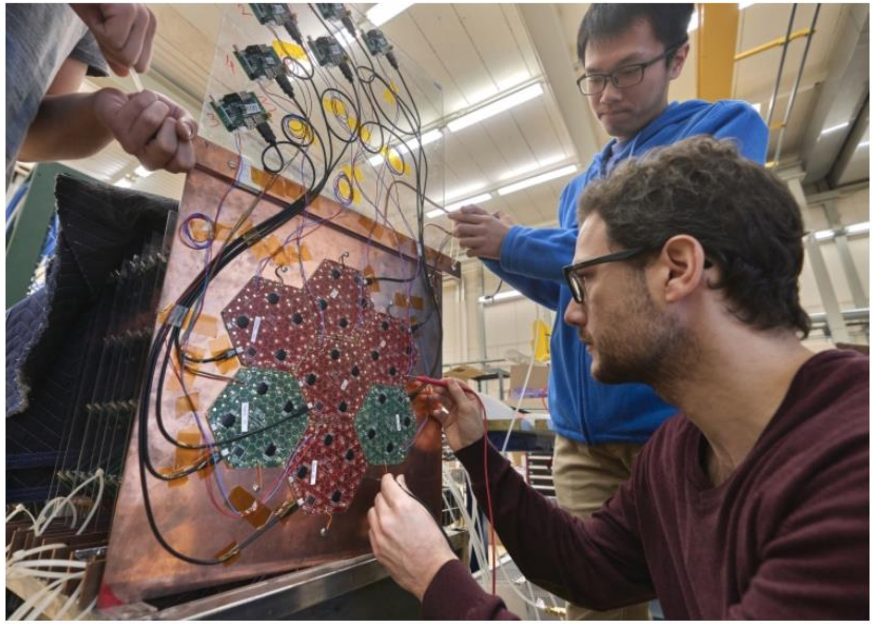
HGCal

- The high-granularity calorimeter is a planned upgrade of CMS
- It is designed to give good performance in the endcaps when LHC is upgraded

High-Luminosity Large Hadron Collider(HL-LHC)

- Increase luminosity by factor of 10 (proportional to number of collisions per time)
- Operational from the end of 2027



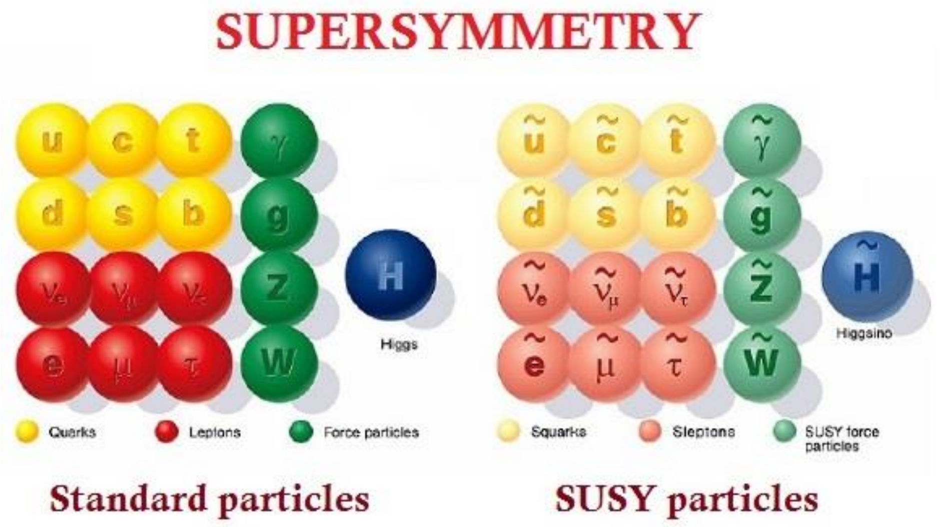


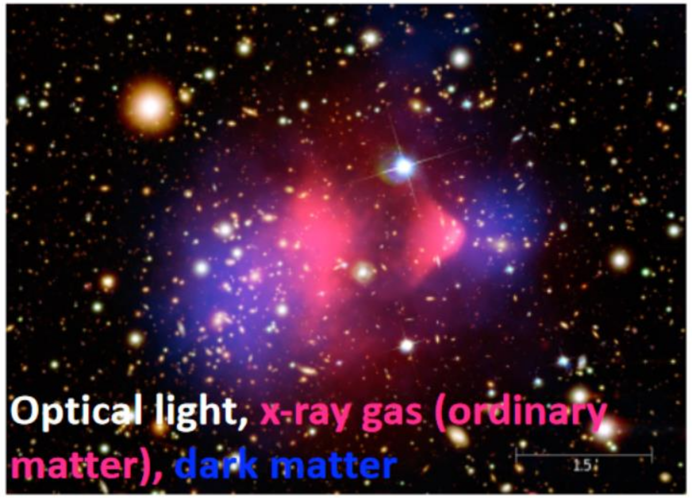
Prototype sensors for the HGCal detector plates

Long-Lived Particles (LLPs)

Long-Lived Particles

- LLPs can decay far from the proton-proton collision
- LLPs arise in many theories of new physics beyond the Standard Model
  - Supersymmetry
  - Elementary Particle nature of Dark Matter
- Triggers that assume decays close to the interaction point may miss LLPs!



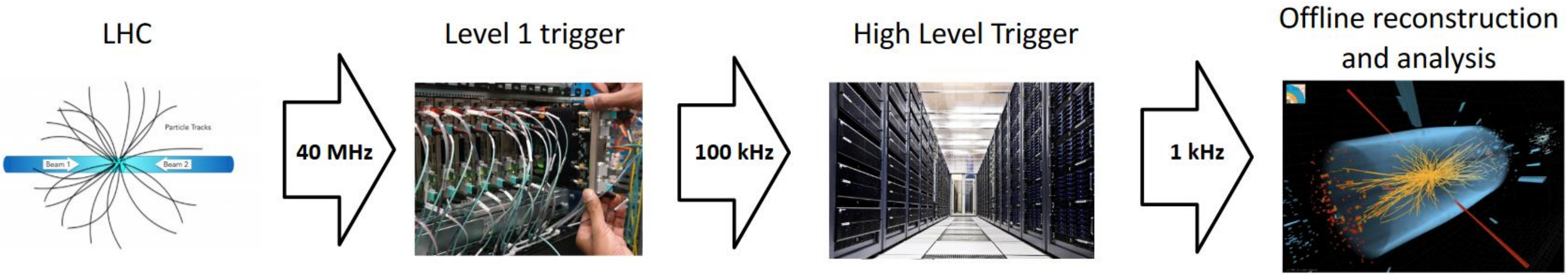


[Chandra X-ray Observatory]

Triggers

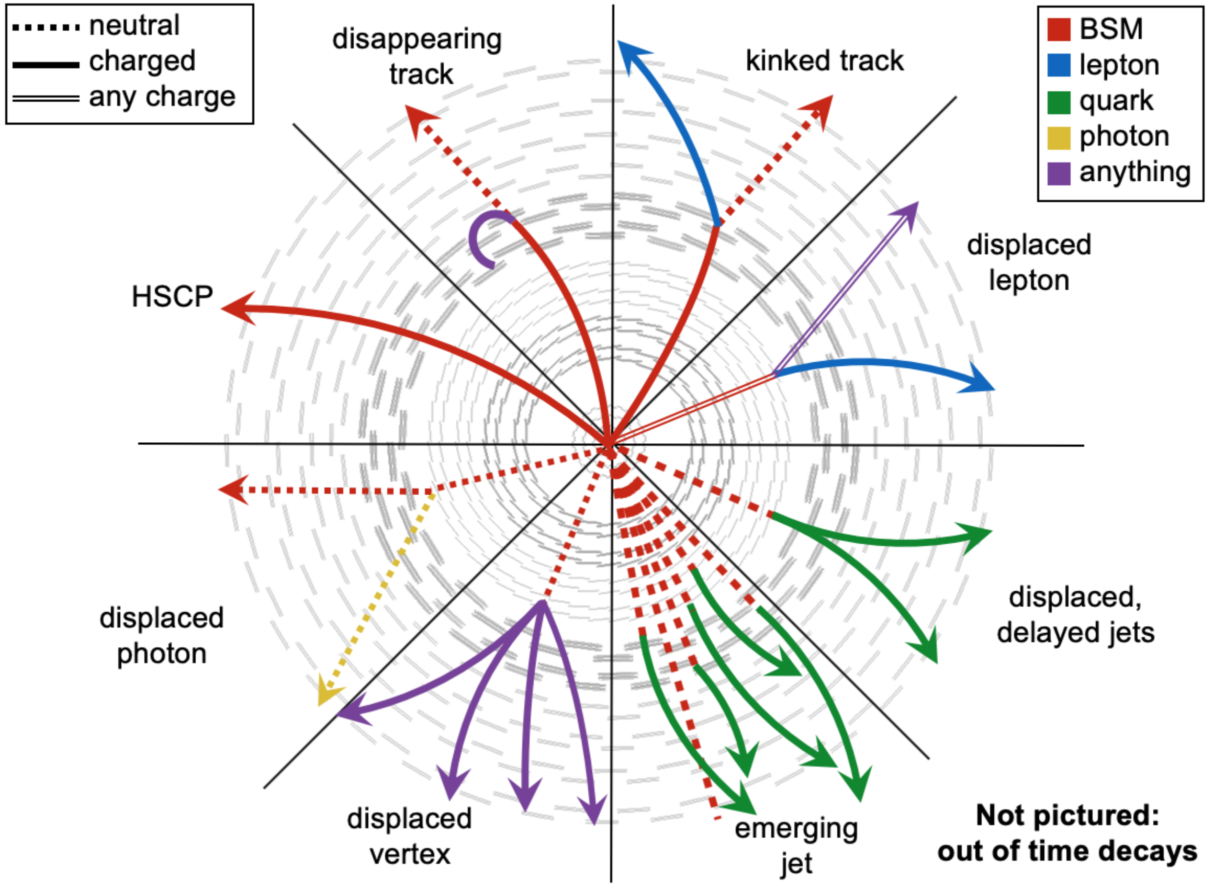
Triggers

- Not feasible to record every single event
- Most events are not interesting (low-momentum QCD)
- Triggers rapidly decide what information to keep
  - Kinematic constraints, energy requirements, fast calculations using FPGAs
- LHC event rate: 40 MHz
  - Level 1 Trigger at CMS reduces to 100 kHz



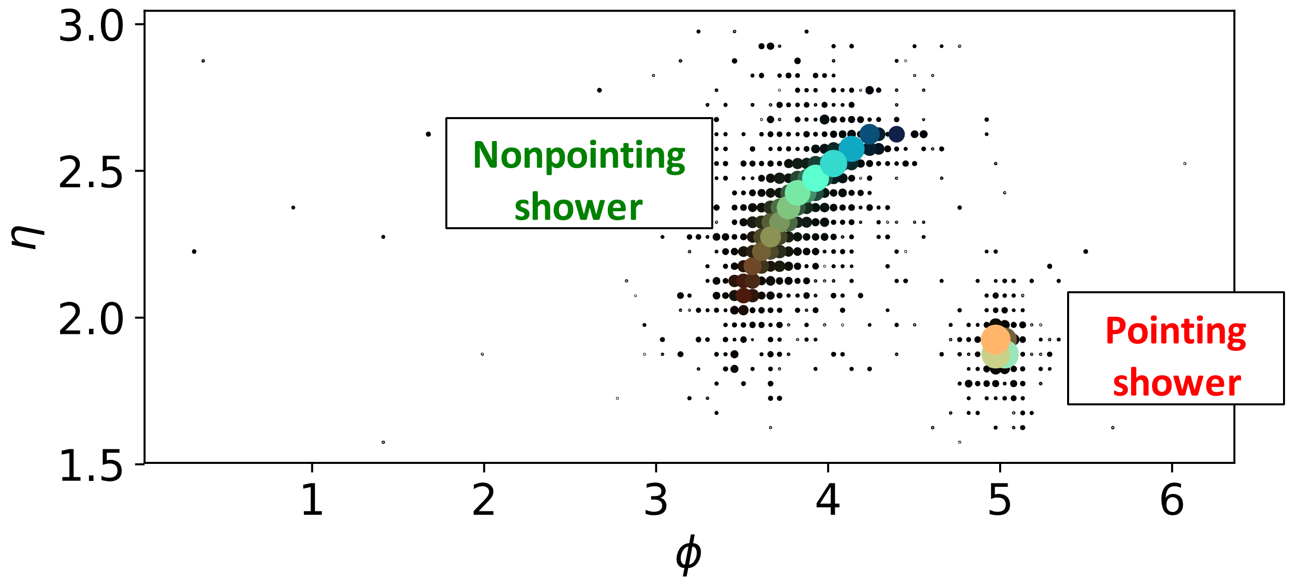
Triggers for Long-Lived Particles (LLPs)

- LLP searches span a wide variety of signatures, models, lifetimes, masses, decay locations, etc.
- The signatures are often **unusual** and **not covered** by "standard" reconstruction or triggers
- If your data is not triggered, it's lost!
- **Dedicated triggers for LLPs are crucial!**



CNN Trigger for LLP Decays in HGCal

- Developed a fast convolutional neural network (CNN) to find nonpointing showers in a high-granularity calorimeter (HGCal)
- Computer vision image recognition can easily differentiate between nonpointing and pointing showers
- Proof of concept paper (https://arxiv.org/abs/2004.10744) published in JINST

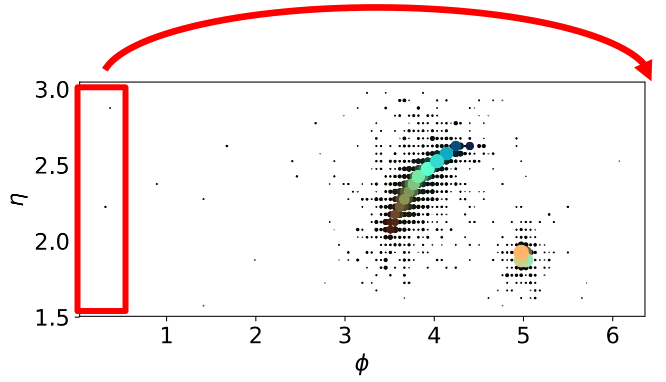


Colors indicate calorimeter layer number

Marker size indicates deposited energy

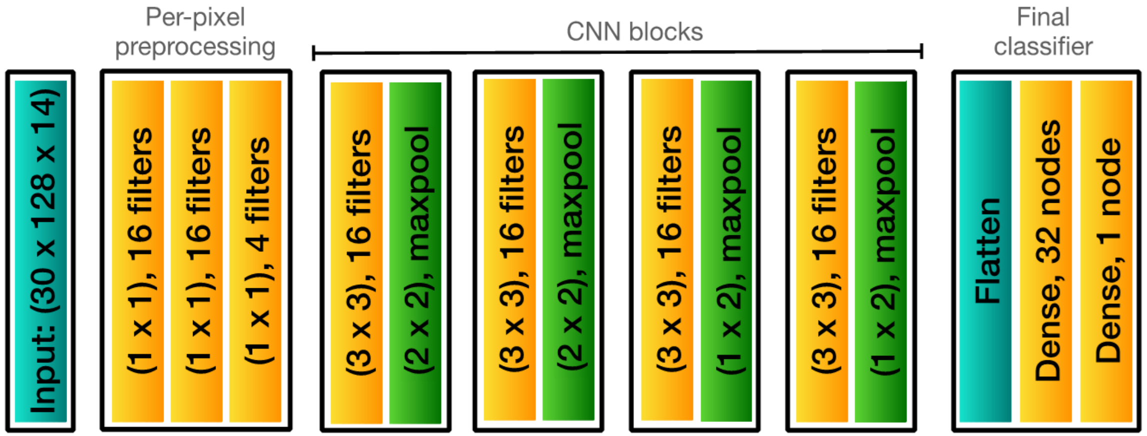
Neural Network and Training

- **Simple CNN designed to provide compromise between performance and resource requirements**
- Pixels from ϕ=0 to ϕ=0.4 are repeated at ϕ=2π to account for particles that enter the calorimeter at ϕ~ 0
  - Therefore 120 pixels in ϕ become 128



- **We consider the full HGCAL in one phi slice**

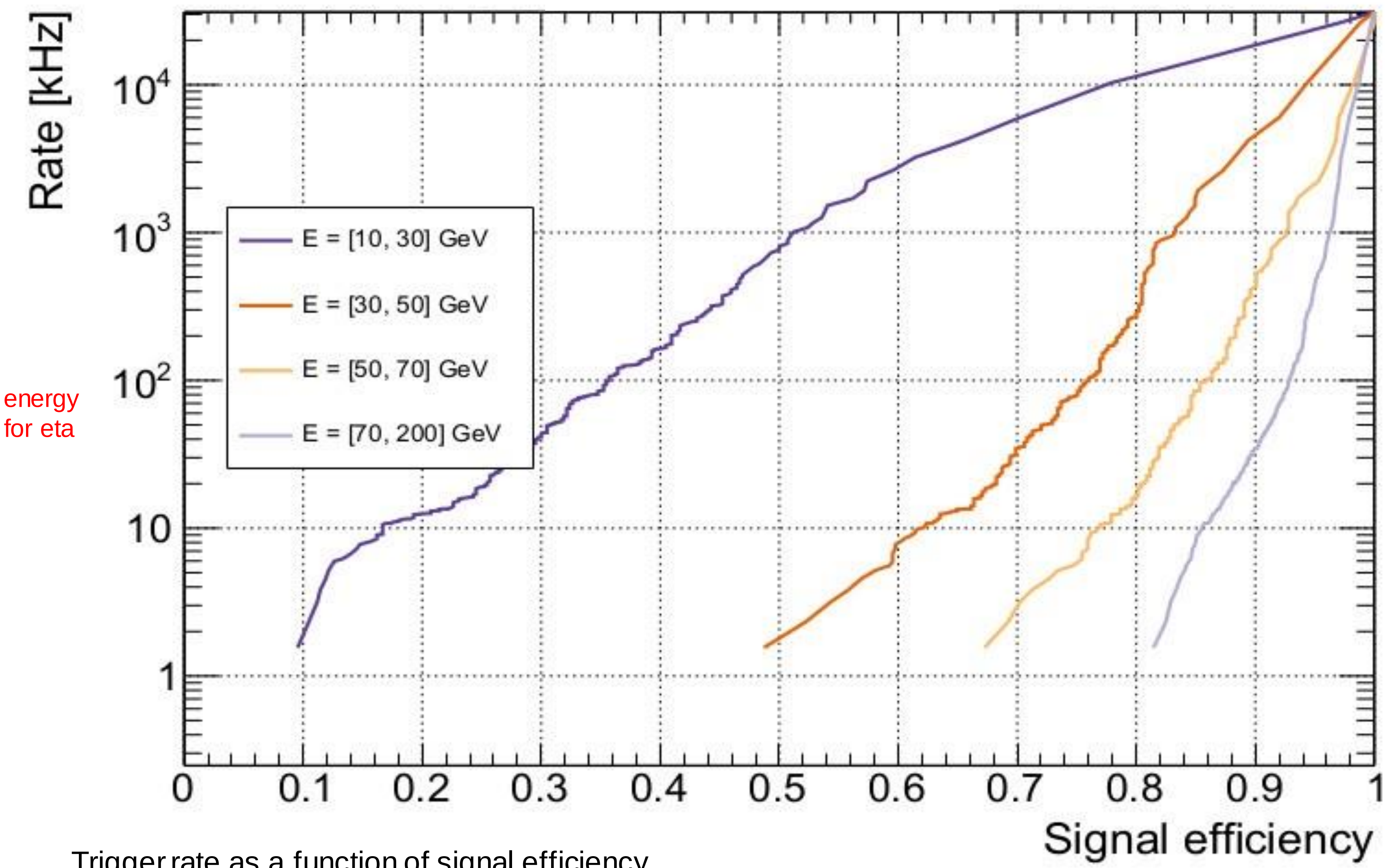
- Per-pixel preprocessing: 14 layers and 4 "colors", to make input to CNN blocks small
- Simple CNN + max pool blocks
- Final classifier with low parameter count
- 1 signal event : 70 minbias events for testing



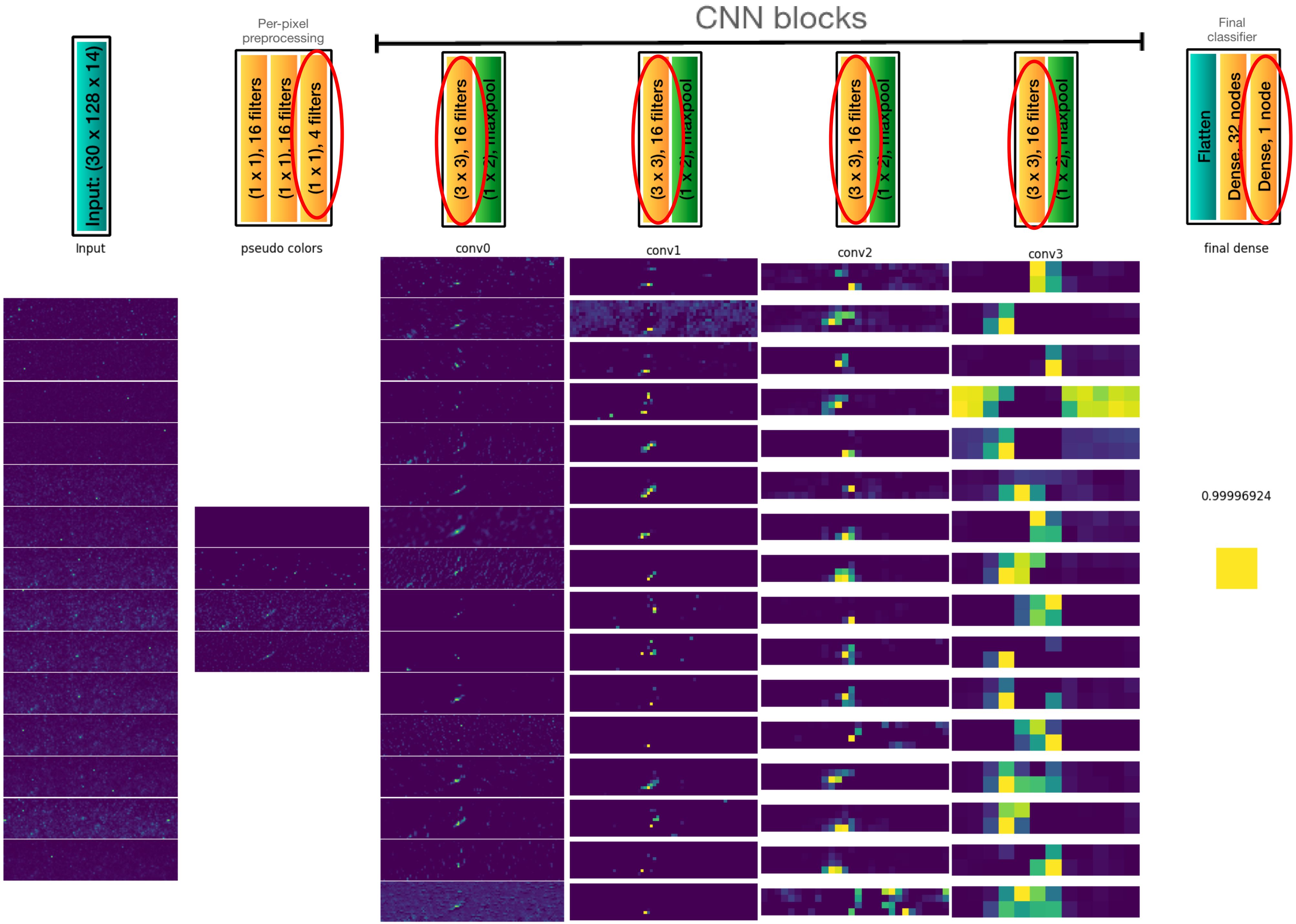
Starting Point: Paper Results

Energy, not pt

The conversion from energy to pt is factor of 2-10 for eta between 1.5 and 3

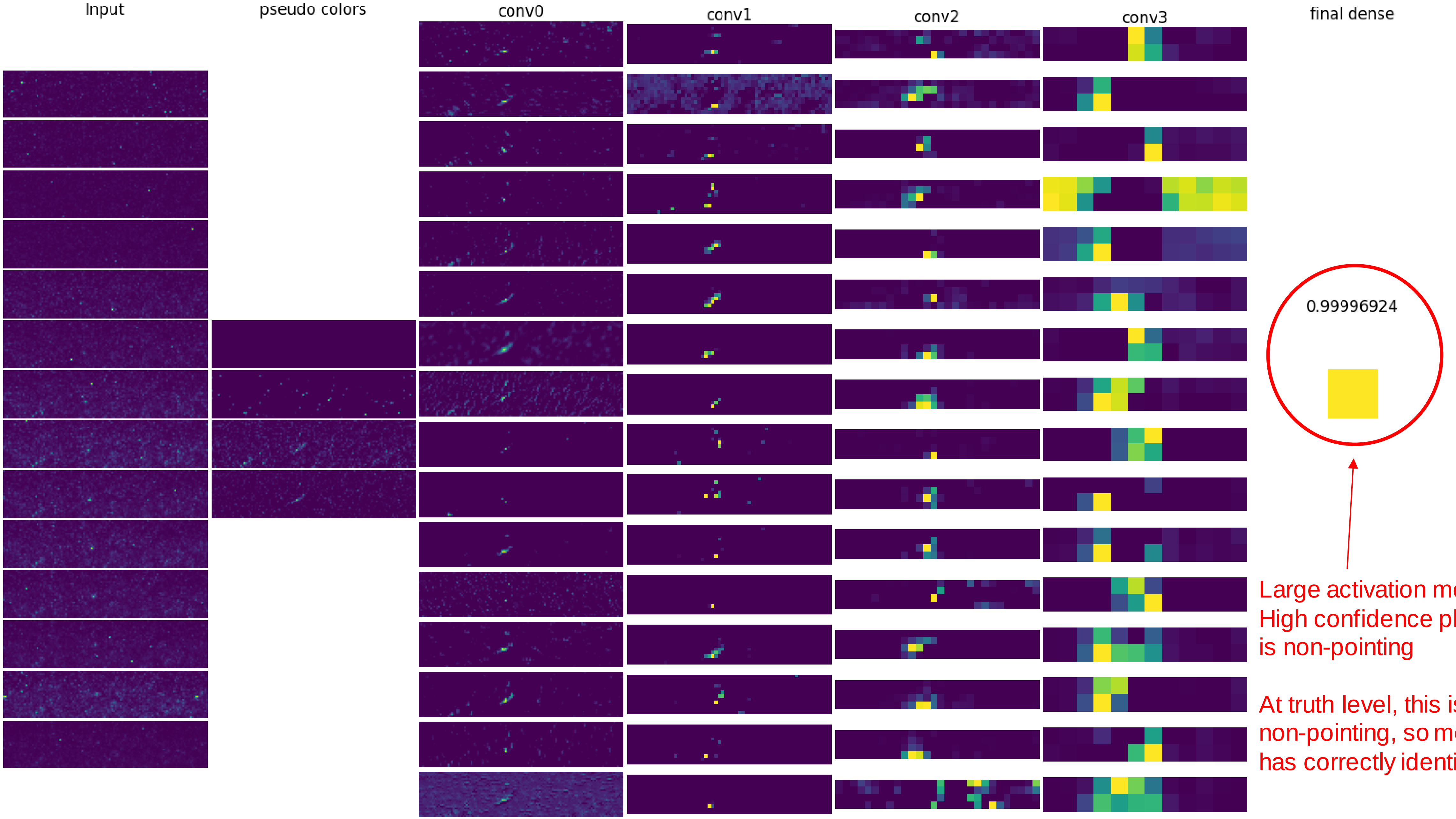


Trigger rate as a function of signal efficiency.



CNN blocks
Input: (30 x 128 x 14)
Input
Per-pixel preprocessing
(1 x 1), 16 filters
(1 x 1), 16 filters
(1 x 1), 4 filters
pseudo colors
(3 x 3), 16 filters
(1 x 2), maxpool
conv0
conv1
conv2
conv3
Final classifier
Flatten
Dense, 32 nodes
Dense, 1 node
final dense
0.99996924

Non-pointing Photon (Signal)

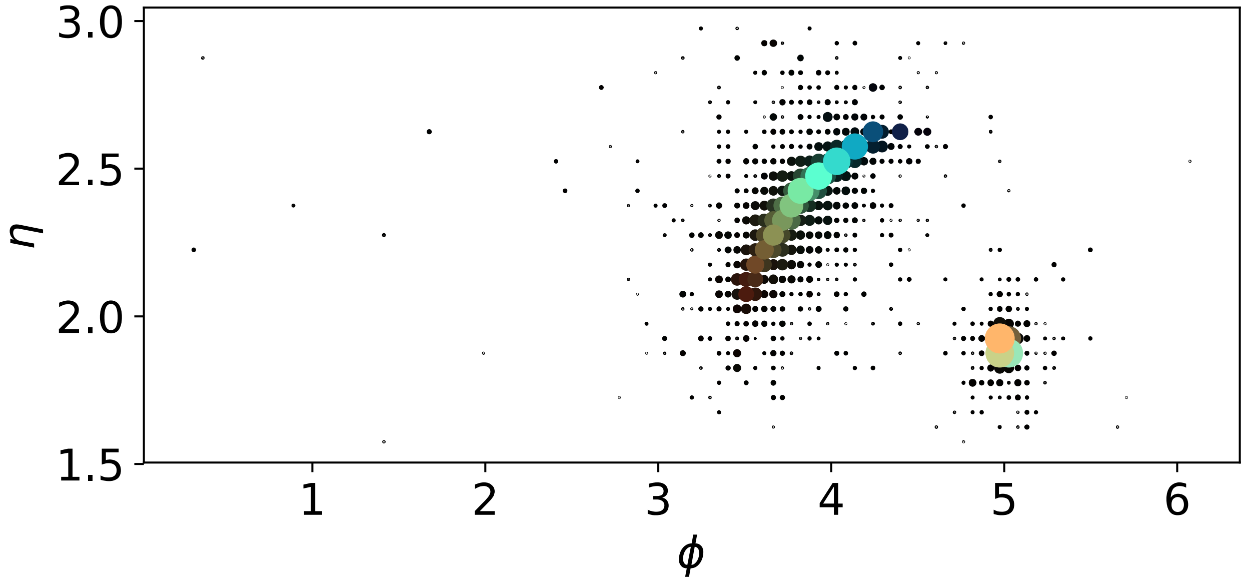


Large activation means
High confidence photon
is non-pointing

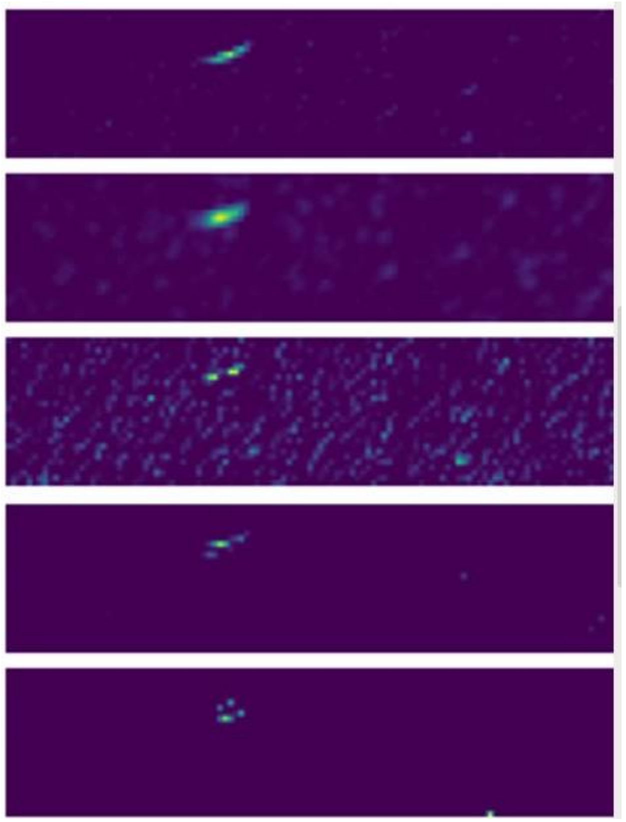
At truth level, this is
non-pointing, so model
has correctly identified.

Non-pointing Photon
(Signal)

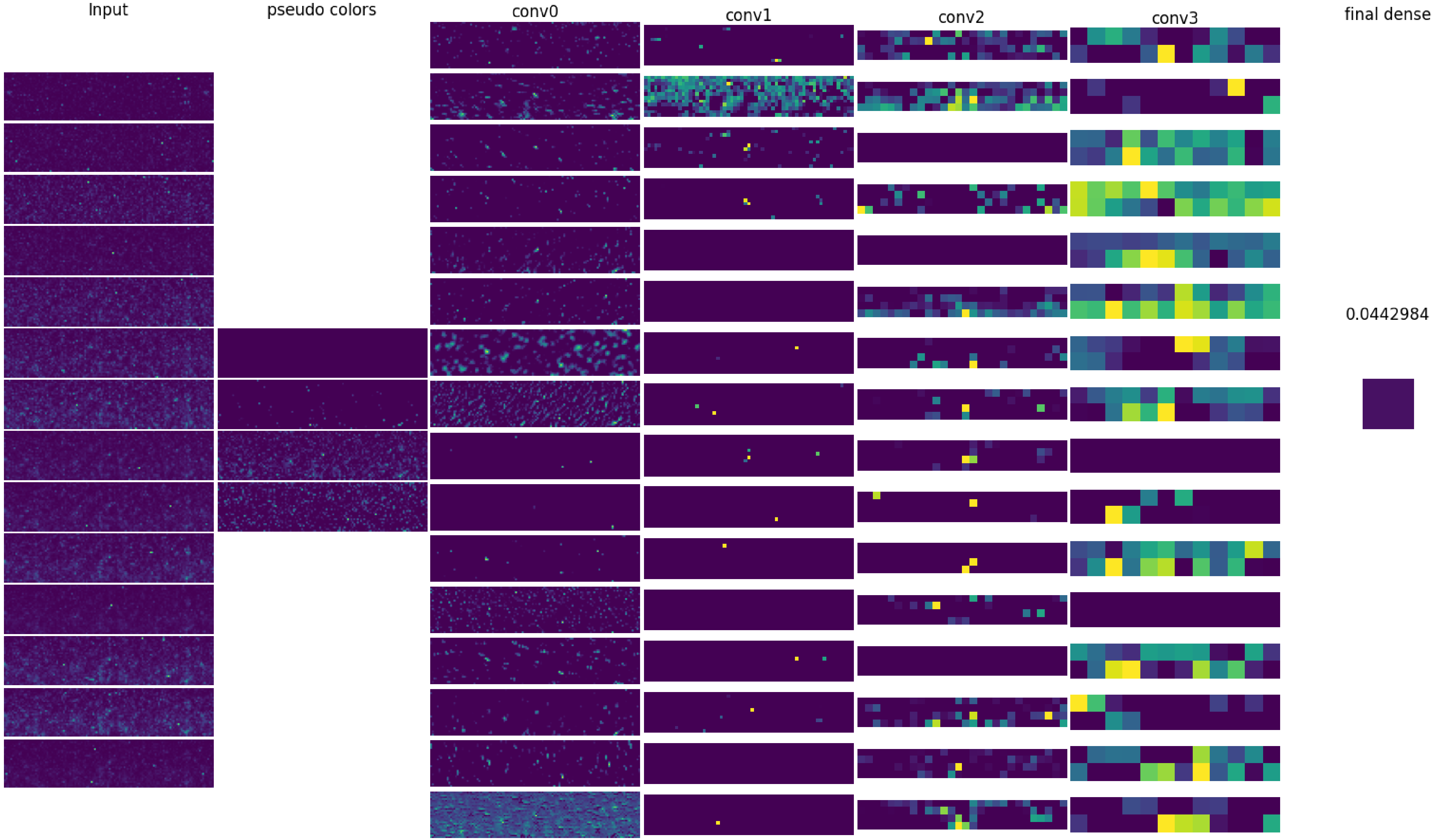
Non pointing Signature



Zoomed in Non-pointing Photon; Conv 0 Filters



Pointing Photon (Background)



Strategies to Improve Neural Network

Add More Maxpooling Layers

- Decrease parameters
- Decrease resolution

conv 1

conv 1 after max pooling

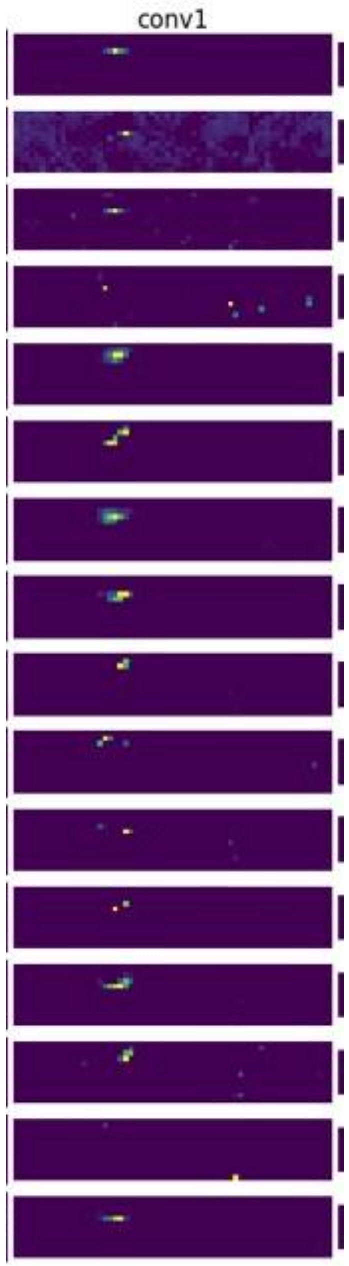
Strategies to Improve Neural Network

Add More Maxpooling Layers

- Decrease parameters
- Decrease resolution

Increase Number of Filters

- Increase number of parameters
- Increase complexity



Strategies to Improve Neural Network

Add More Maxpooling Layers

- Decrease parameters
- Decrease resolution

Increase Number of Filters

- Increase number of parameters
- Increase complexity

Remove/Add CNN Blocks

- Affects number of parameters

Strategies to Improve Neural Network

Add More Maxpooling Layers
- Decrease parameters
- Decrease resolution

Increase Number of Filters
- Increase number of parameters
- Increase complexity

Remove/Add CNN Blocks
- Affects number of parameters

Remove Preprocessing Layers
- Changes input size to first CNN Block

Strategies to Improve Neural Network

Add More Maxpooling Layers

- Decrease parameters
- Decrease resolution

Increase Number of Filters

- Increase number of parameters
- Increase complexity

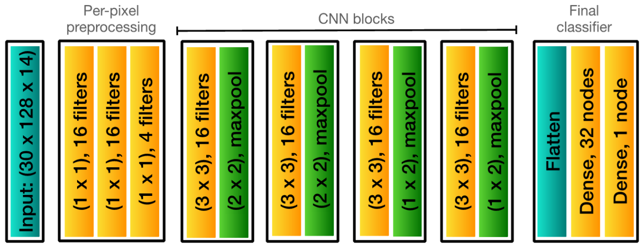
Remove/Add CNN Blocks

- Affects number of parameters

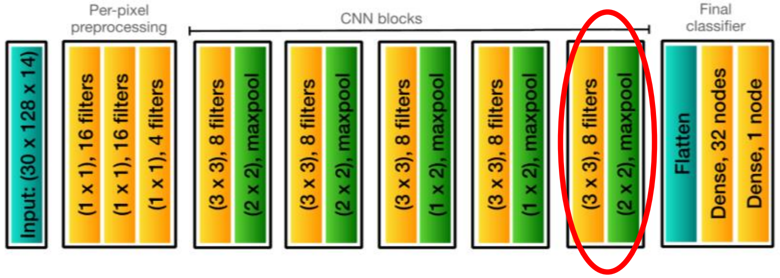
Remove Preprocessing Layers

- Changes input size to first CNN Block

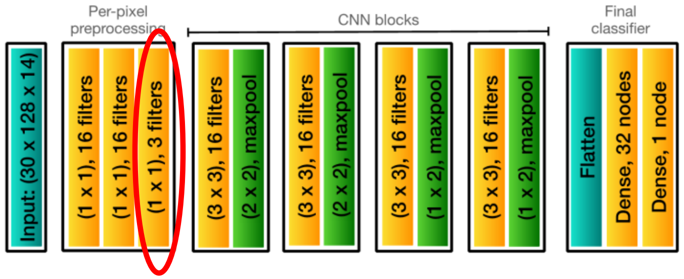
Original



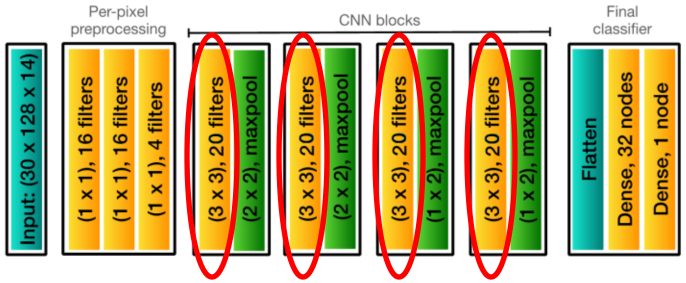
Add block 8 filters



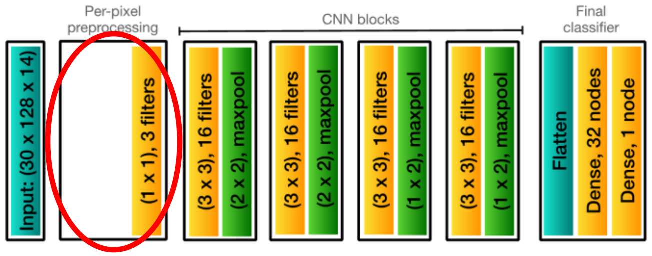
3 Colors



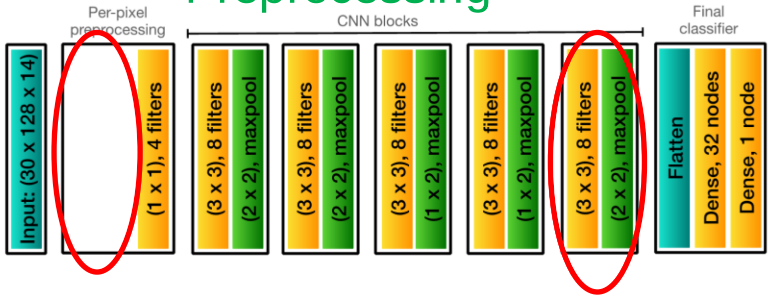
Conv 16 to 20



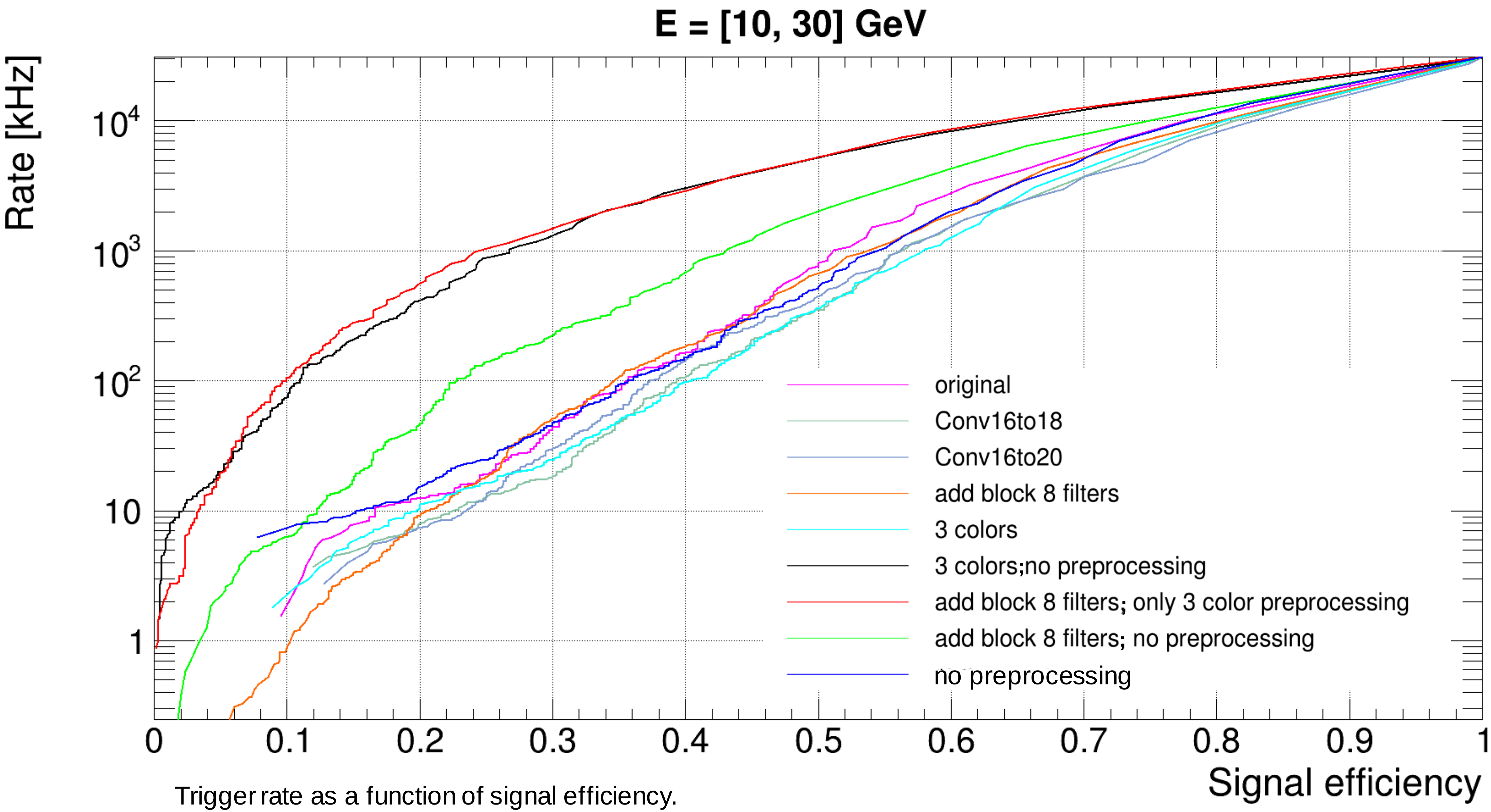
3 Colors; no Preprocessing



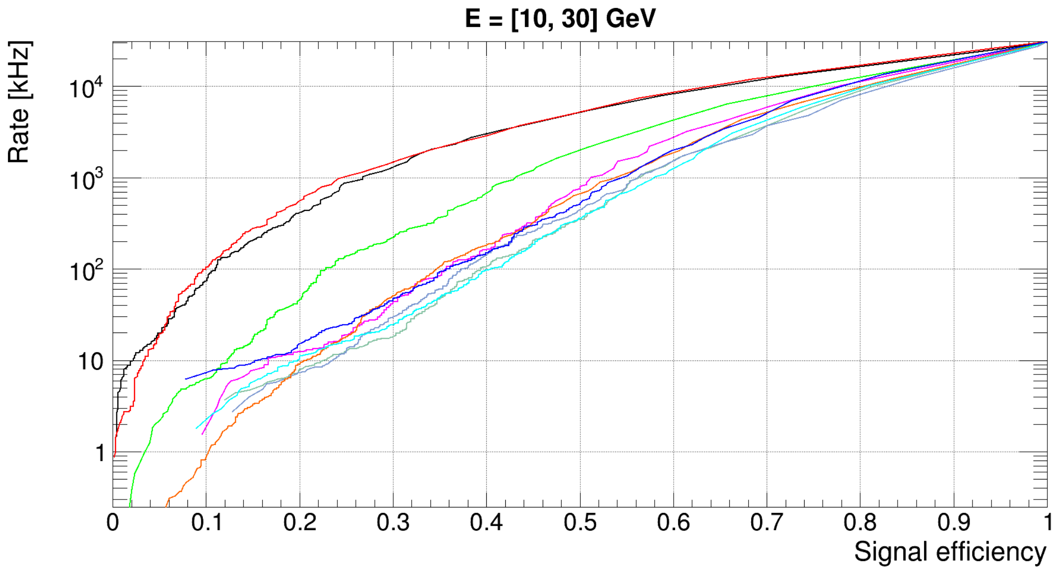
Add Block 8 filters; no Preprocessing

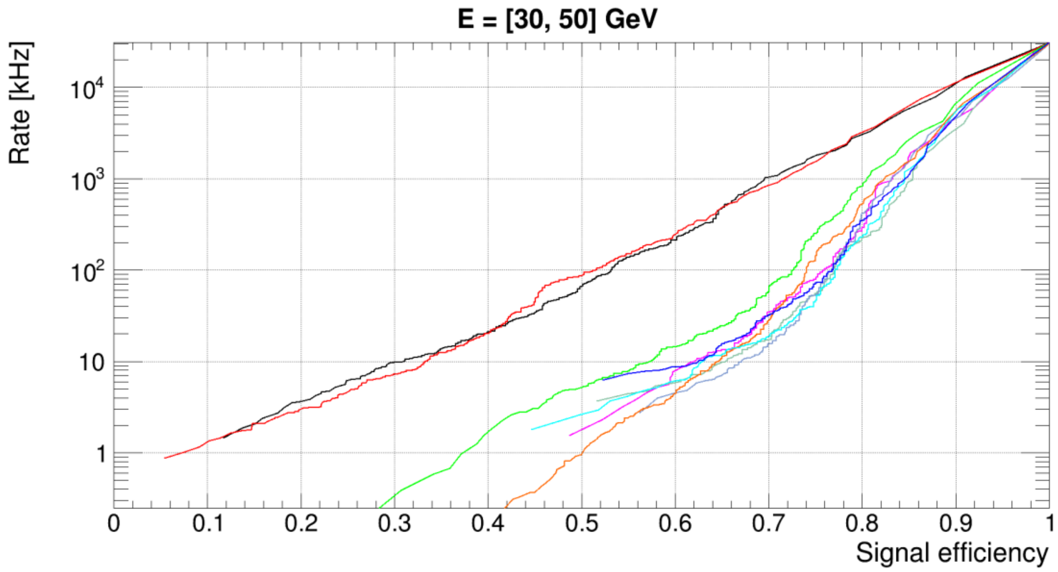


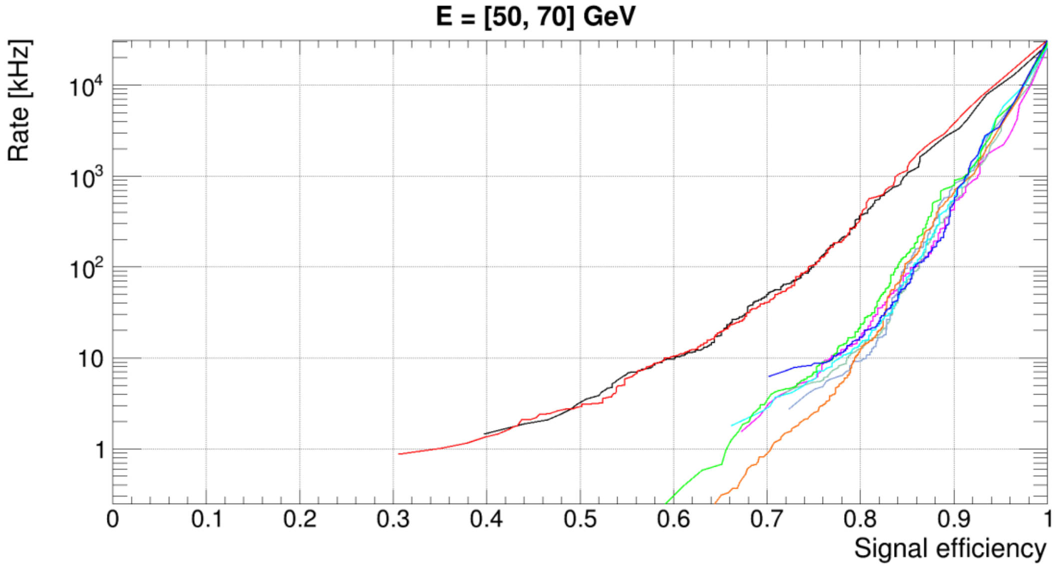
Efficiencies of All Tested Models

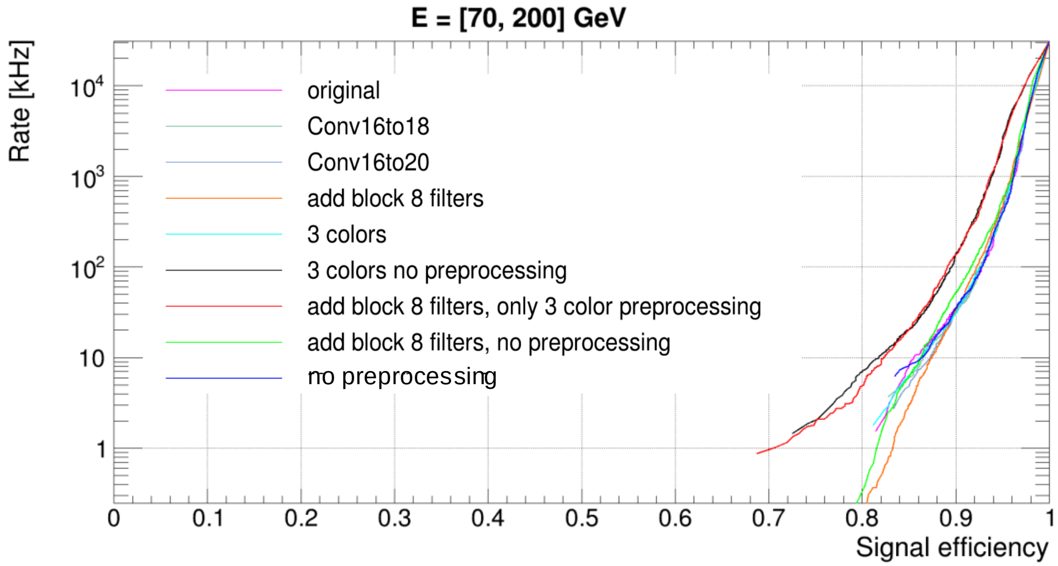


Trigger rate as a function of signal efficiency.

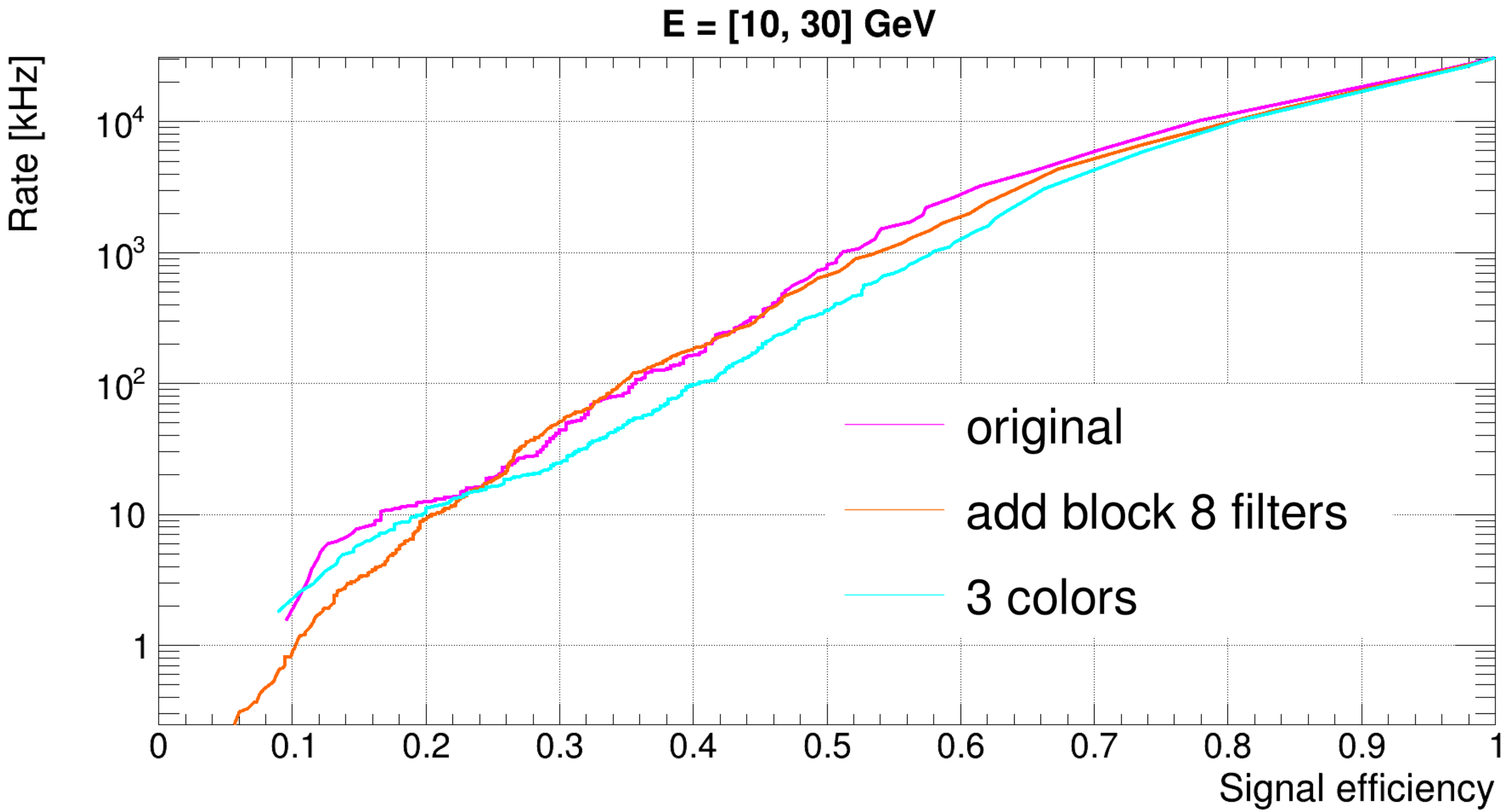

E = [10, 30] GeV
Rate [kHz]
Signal efficiency

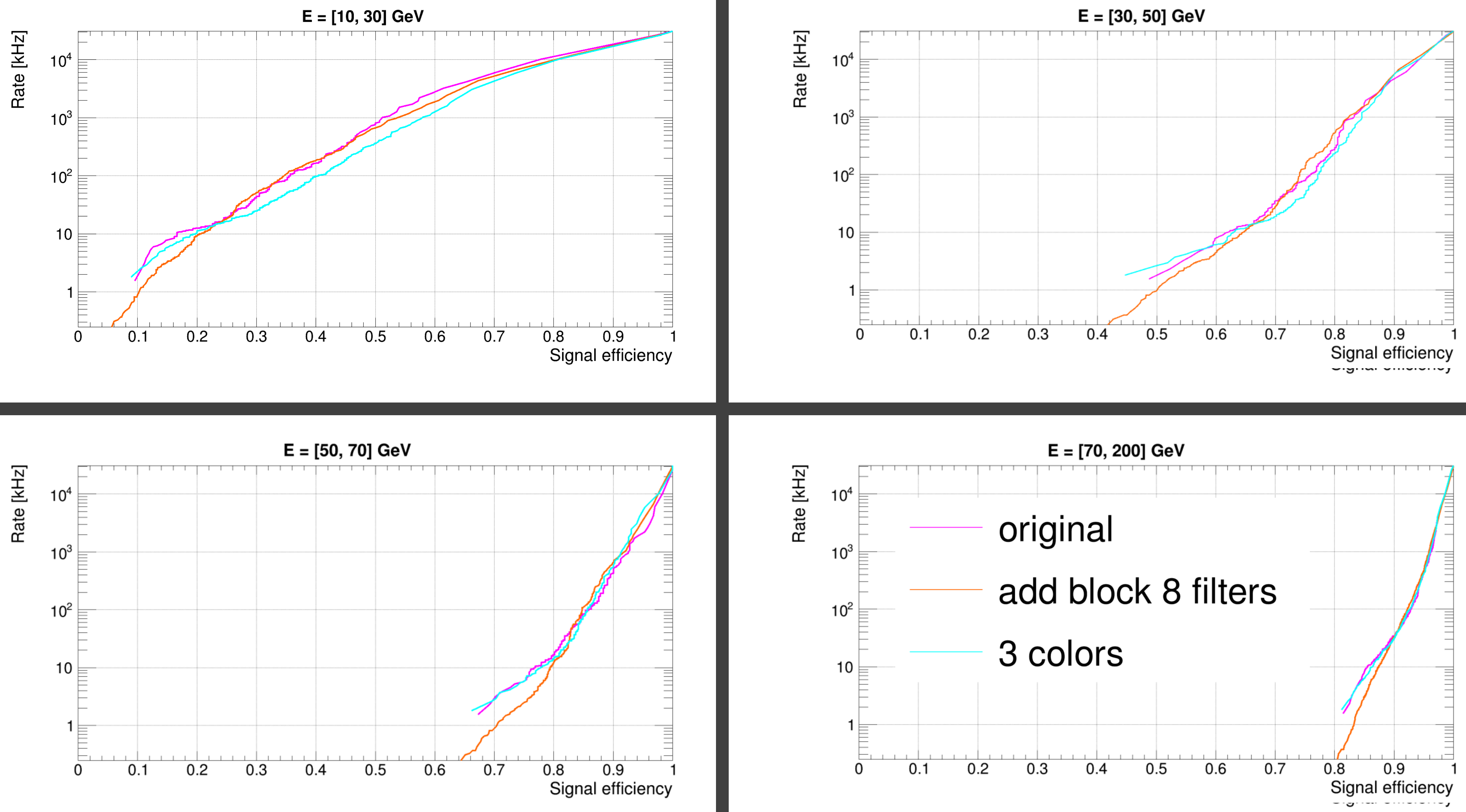

E = [30, 50] GeV
Rate [kHz]
Signal efficiency


E = [50, 70] GeV
Rate [kHz]
Signal efficiency


E = [70, 200] GeV
Rate [kHz]
Signal efficiency
original
Conv16to18
Conv16to20
add block 8 filters
3 colors
3 colors no preprocessing
add block 8 filters, only 3 color preprocessing
add block 8 filters, no preprocessing
no preprocessing

Efficiencies of Best Models

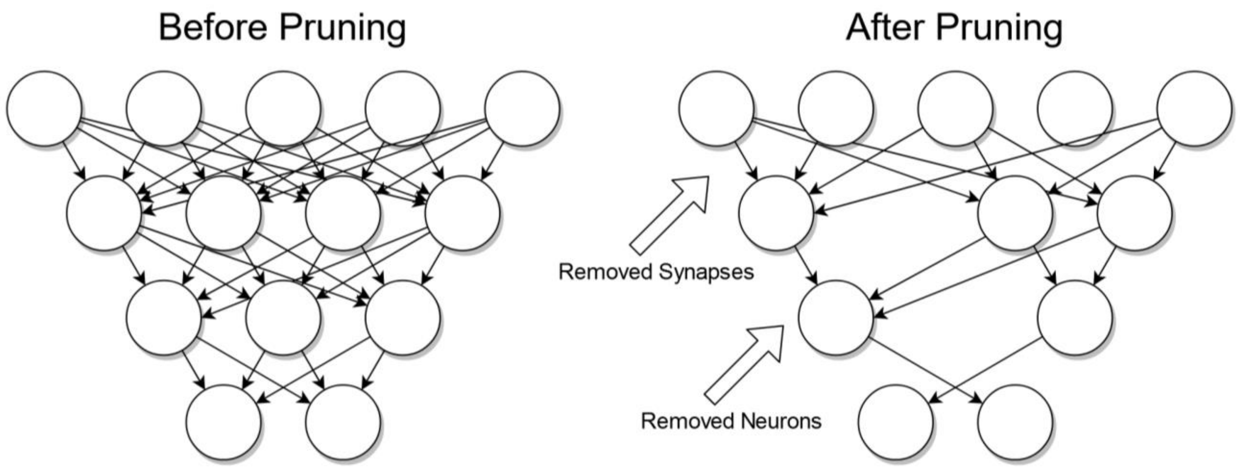



E = [10, 30] GeV
E = [30, 50] GeV
E = [50, 70] GeV
E = [70, 200] GeV
Rate [kHz]
Signal efficiency
original
add block 8 filters
3 colors

Minimizing Resources

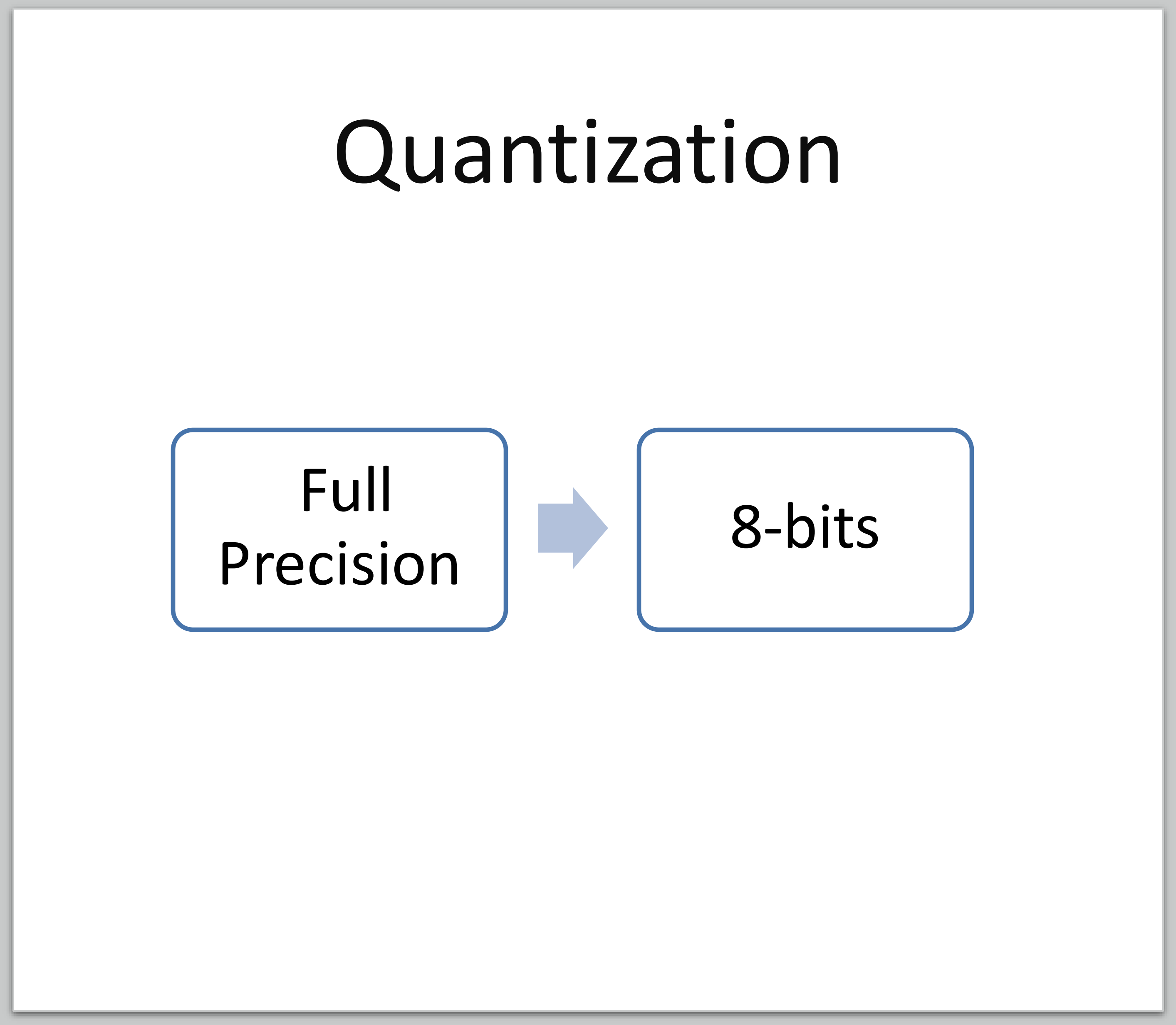
- Pruning **removes unused parameters**
- **Lowers resource requirements** by eliminating unnecessary activations

Pruning

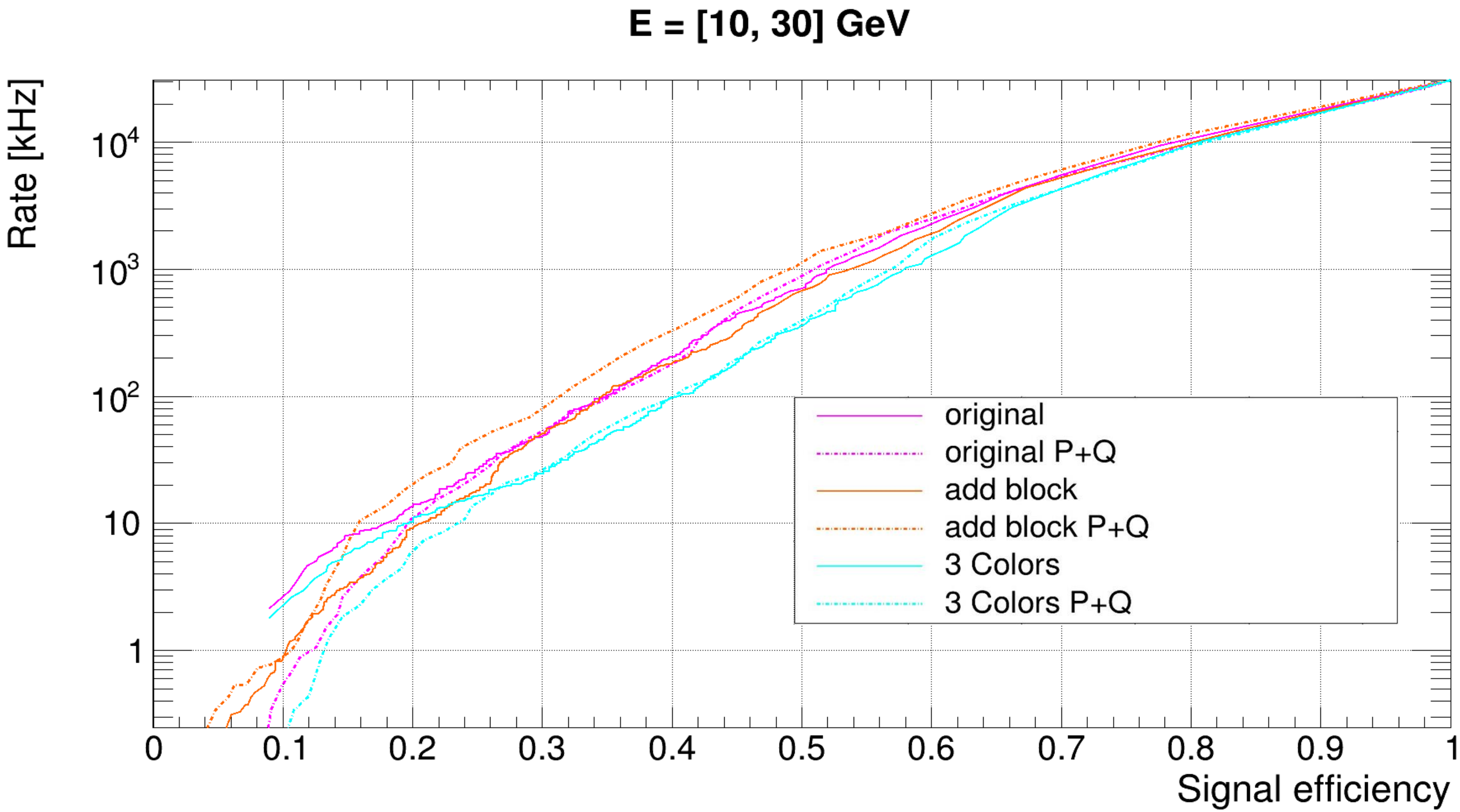


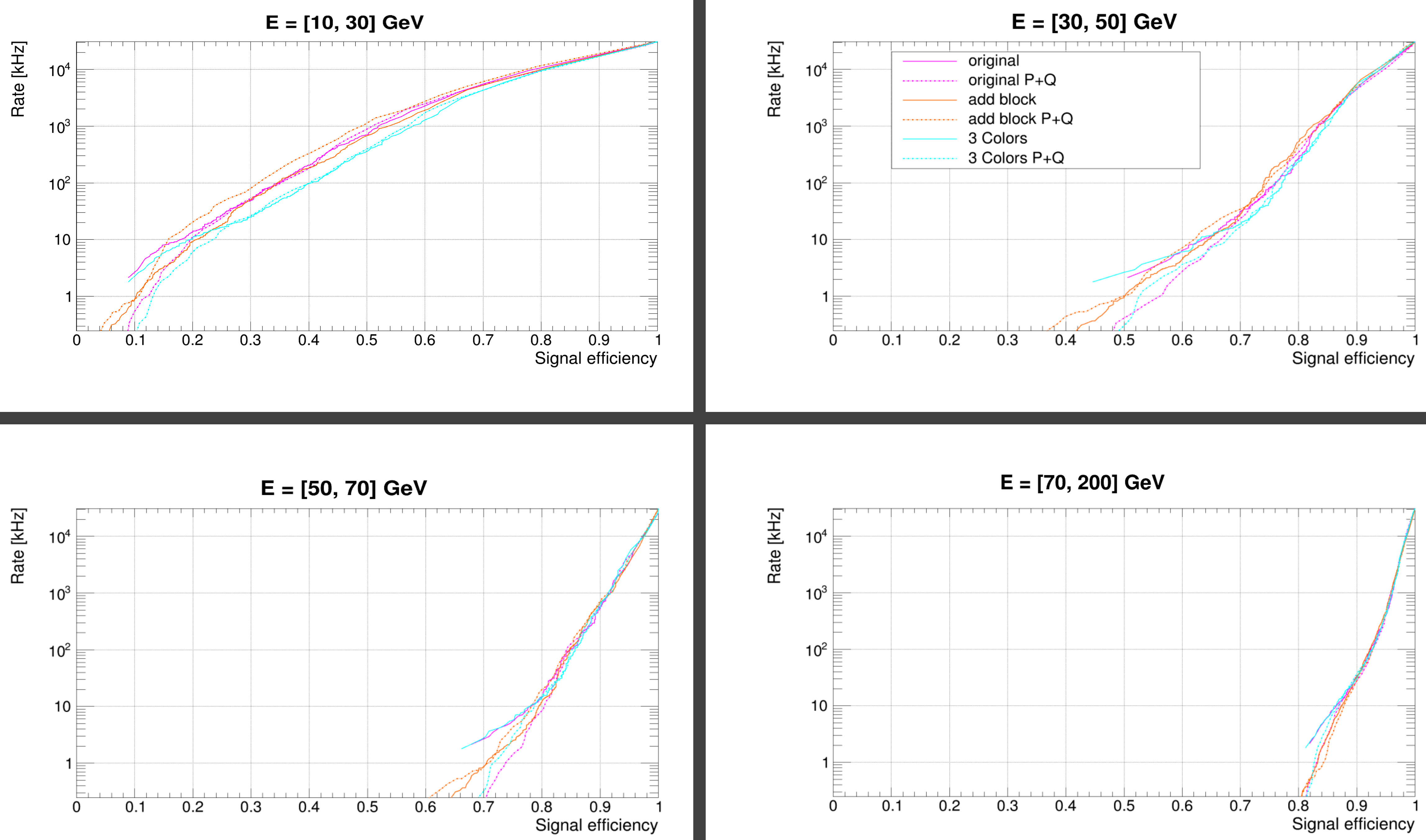
Minimizing Resources

- Quantization **reduces precision of numbers that model must store**

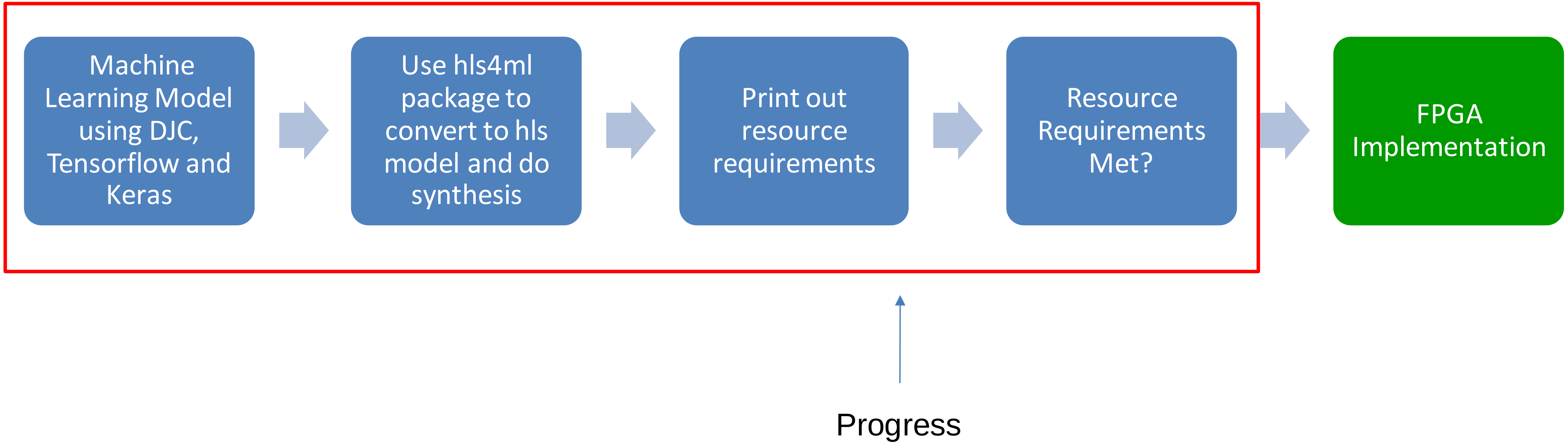


Efficiencies of P+Q Models




E = [10, 30] GeV
E = [30, 50] GeV
E = [50, 70] GeV
E = [70, 200] GeV
Rate [kHz]
Signal efficiency
original
original P+Q
add block
add block P+Q
3 Colors
3 Colors P+Q

Resource Requirements with **hls4ml** package



Reminder: We consider the full HGCAL in one phi slice

Resource Requirements

MODE	Model	Latency (us)
RESOURCE	Original	115.945
RESOURCE	3 Colors Pruned and Quantized	154.305
LATENCY	Add Block Pruned and Quantized	43.24
LATENCY	3 Colors Pruned and Quantized, reduced final CNN complexity	**19.725**

Summary

- Continued work beyond the proof of concept paper by optimizing neural network models to identify non pointing showers

- Showed that required latency is likely feasible when implementing the CNN onto an FPGA