

Deep Learning for Event Classification with CTA

Ari Brill, Columbia University
Fermi Summer School
June 4, 2018



Cherenkov Telescope Array (CTA)

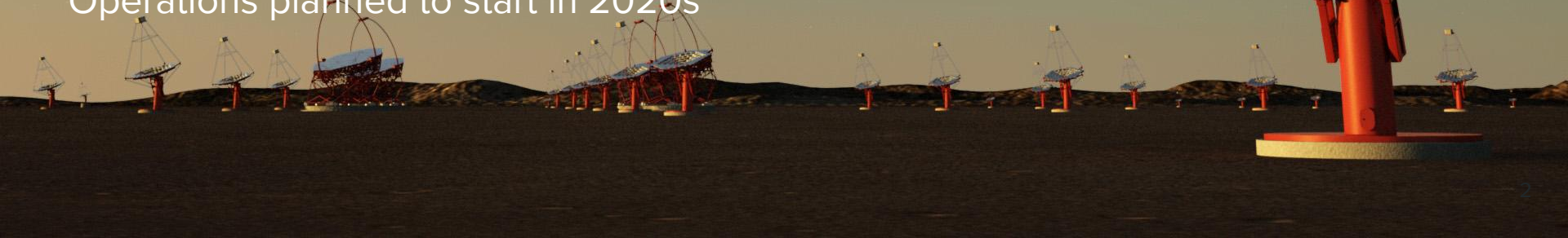
The next-generation Imaging Air Cherenkov Telescope (IACT) array

Small, Medium, and Large Size Telescopes

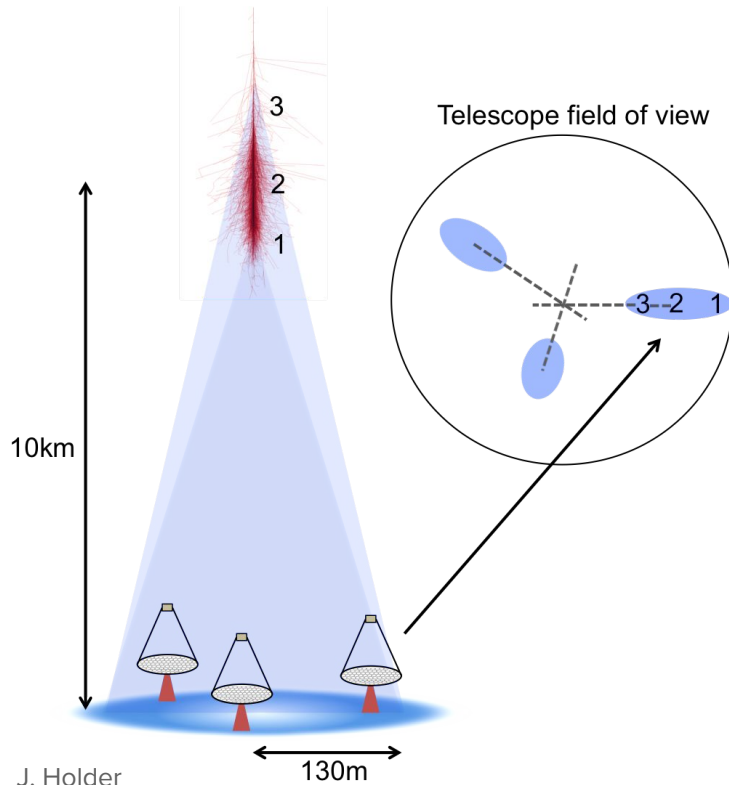
CTA North: La Palma, Canary Islands, Spain

CTA South: Paranal, Atacama Desert, Chile

Operations planned to start in 2020s



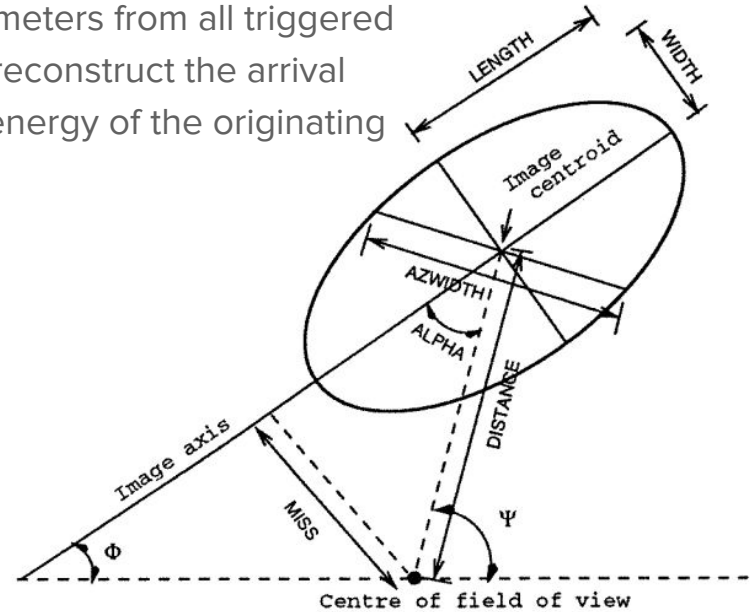
IACT Event Reconstruction



J. Holder

Model each shower image as an ellipse and parametrize by its moments

Combine parameters from all triggered telescopes to reconstruct the arrival direction and energy of the originating gamma ray



100 GeV
photon

100 GeV
proton

Cosmic-ray showers
are a massive
background! Reject
them using the
shower shape

camera
(FOV)

γ -ray

length

width

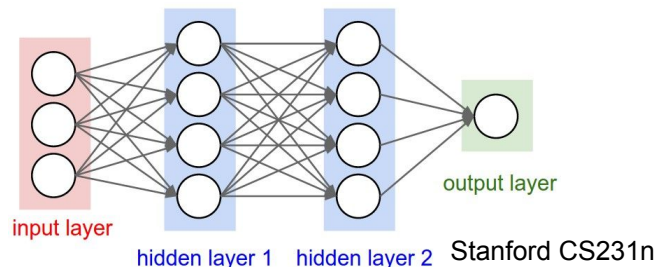
cosmic ray

Deep Learning

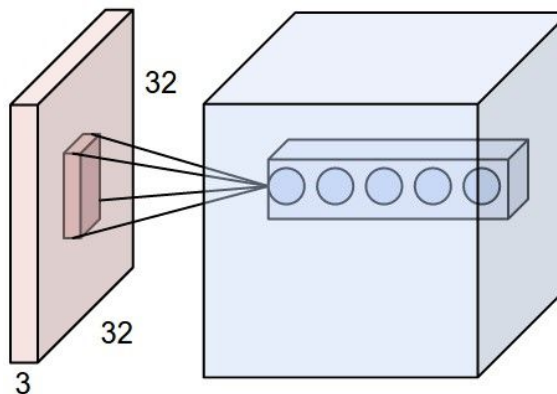
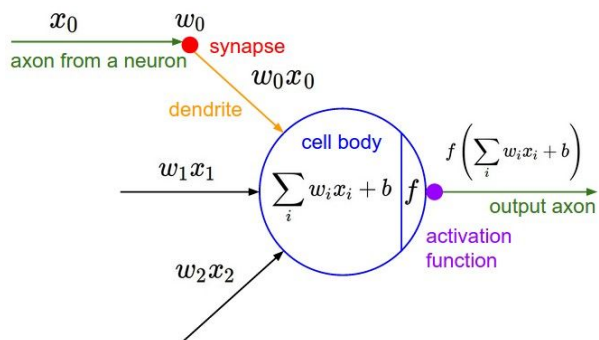
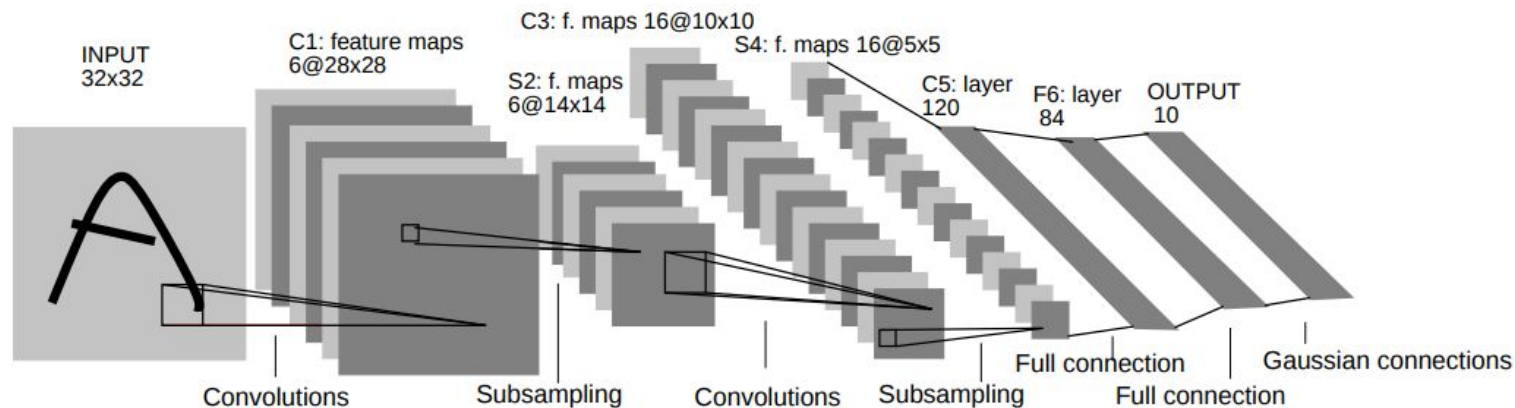
Convolutional Neural Network (CNN): machine learning technique that can directly classify camera images, with no explicit intermediate parameterization

Needs lots of training data and parallel processing using graphics cards (~10x speedup)

CNNs have been used to classify gamma-ray and muon images (HESS, VERITAS) and are currently being studied by many groups in CTA



Convolutional Neural Networks



Schwarzschild-Couder Telescope (SCT)

Prototype under construction at VERITAS site in Arizona

Dual-mirror optics and high-resolution camera improve sensitivity, angular resolution, and field of view compared to standard medium size telescopes

SCT images have high resolution and square pixels, making them well suited for analysis with neural networks



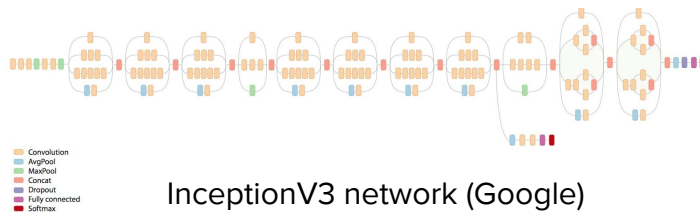
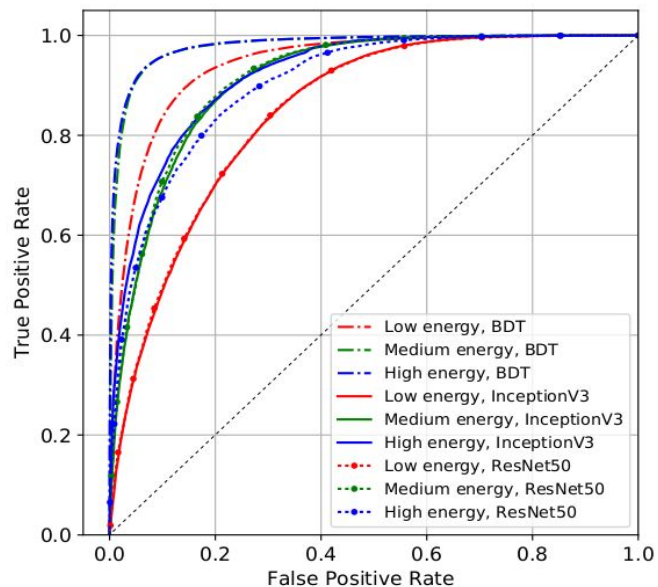
Proof of Concept: SCT Gamma/Hadron Classification

Trained on single telescope SCT images of diffuse gammas and protons

Used slightly modified well-known CNN models, InceptionV3 (Google) and ResNet50 (Microsoft)

Neural networks work, but need stereo reconstruction to match existing analysis methods

Nieto, Brill, Kim, Humensky: [arXiv:1709.05889](https://arxiv.org/abs/1709.05889)

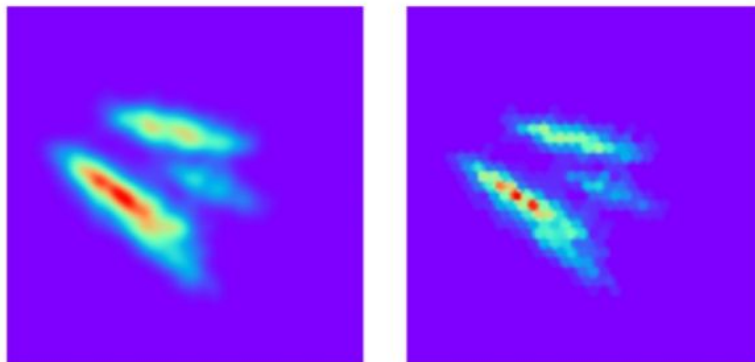
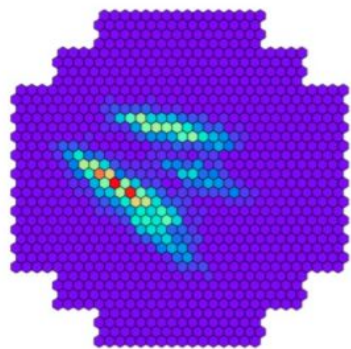


Model	Low E.	Med. E.	High E.
<i>ResNet50</i>	81.1%	90.1%	91.2%
<i>Inception V3</i>	81.4%	90.1%	91.6%

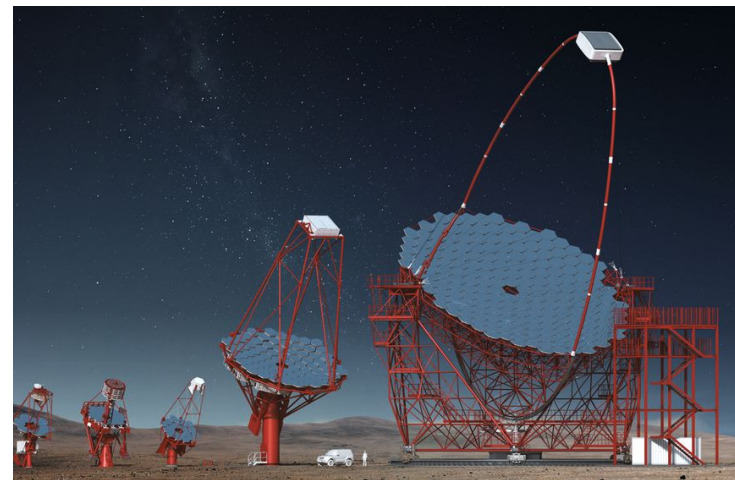
IACT Deep Learning Challenges

Combine images from a telescope array containing many different telescopes which may or may not have triggered → best results so far achieved with recurrent neural networks, but still preliminary

Process **hexagonally spaced pixels** into a square matrix → many methods under study, no consensus yet



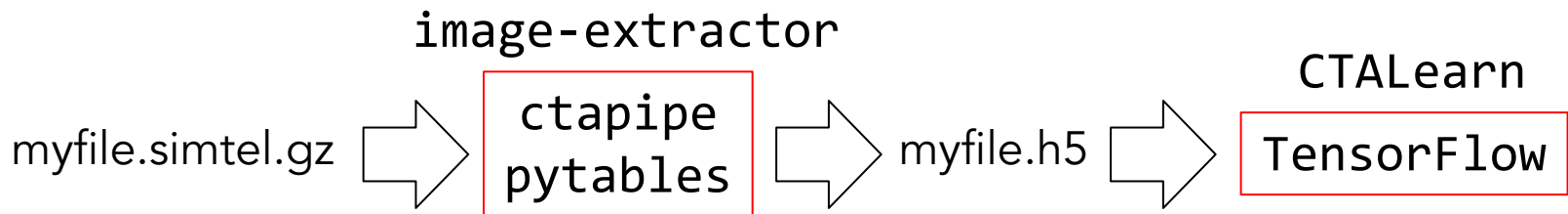
Shilon et al. 2018



Get lots of **labeled training data**:

1. Reprocess MC simulations → ImageExtractor
2. Understand effect of subtle differences between sims and data → not there yet

Training Pipeline: ImageExtractor and CTALearn



The screenshot shows a PyTables database viewer with several tables open:

- Event Info Table of events:** A table with columns: alt, az, core_x, core_y, event number, first hit, mc energy, particle id, num. number, LST indices, MSTF indices, MSTN indices, MSTP indices, MSTS indices. It contains 56 rows of event data.
- Array Info Table of array data:** A table with columns: num. array, det. id, det. type, det. x, det. y, det. z. It contains 11 rows of array data.
- LST Table of LST images:** A table with columns: event_index, image_charge, image_peak. It contains 11 rows of LST image data.
- MSTF Table of MSTF images:** A table with columns: event_index, image_charge, image_peak. It contains 11 rows of MSTF image data.
- MSTN Table of MSTN images:** A table with columns: event_index, image_charge, image_peak. It contains 11 rows of MSTN image data.
- MSTP Table of MSTP images:** A table with columns: event_index, image_charge, image_peak. It contains 11 rows of MSTP image data.
- MSTS Table of MSTS images:** A table with columns: event_index, image_charge, image_peak. It contains 11 rows of MSTS image data.



<https://github.com/cta-observatory/image-extractor>

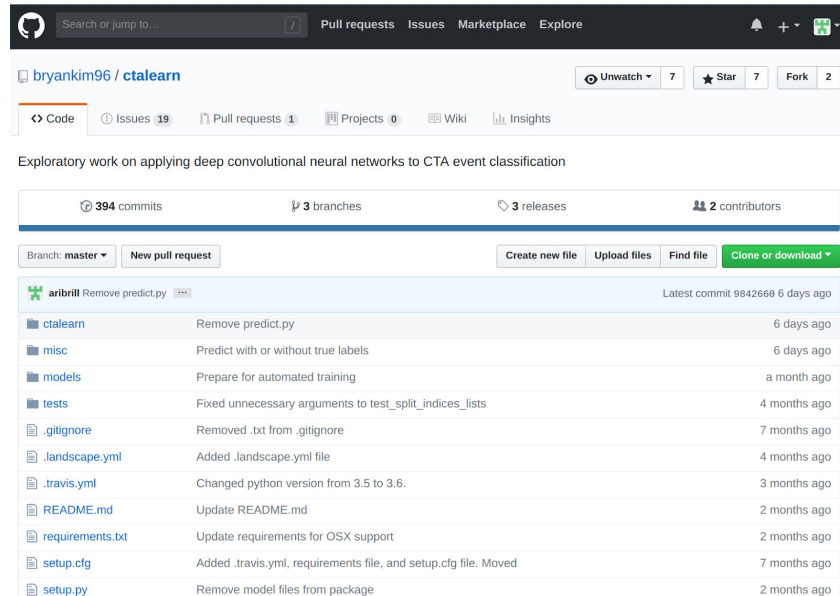
Daniel Nieto (UCM); Bryan Kim, Ari Brill (Columbia)

CTALearn

High-level Python library for doing deep learning with IACT image data

Includes modules for loading and manipulating ImageExtractor HDF5 data and for running machine learning models with TensorFlow

Configuration-file-driven workflow drives reproducible training and prediction



Search or jump to ... Pull requests Issues Marketplace Explore

bryankim96 / ctalearn Unwatch 7 Star 7 Fork 2

Code Issues 19 Pull requests 1 Projects 0 Wiki Insights

Exploratory work on applying deep convolutional neural networks to CTA event classification

394 commits 3 branches 3 releases 2 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

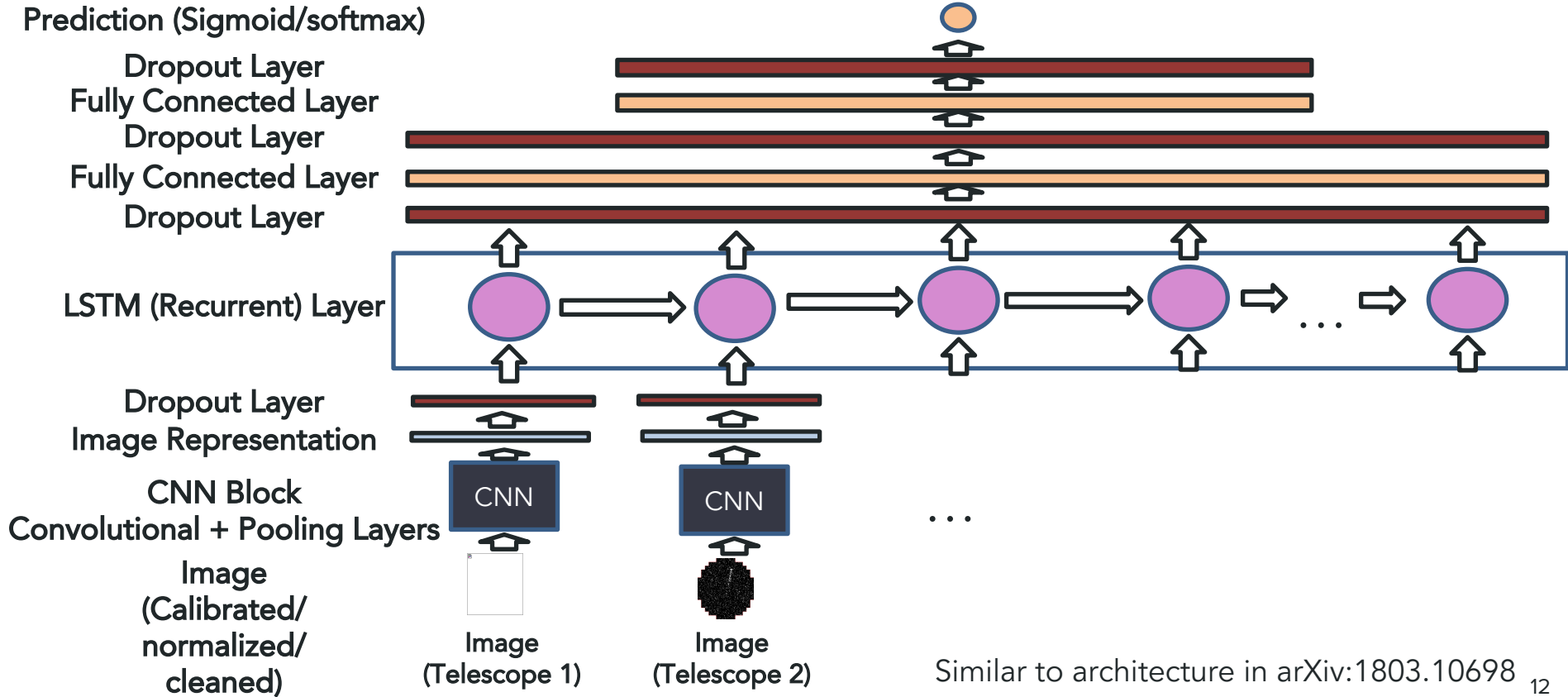
File	Commit Message	Time Ago
ctalearn	Remove predict.py	6 days ago
misc	Predict with or without true labels	6 days ago
models	Prepare for automated training	a month ago
tests	Fixed unnecessary arguments to test_split_indices_lists	4 months ago
.gitignore	Removed .txt from .gitignore	7 months ago
.landscape.yml	Added .landscape.yml file	4 months ago
.travis.yml	Changed python version from 3.5 to 3.6.	3 months ago
README.md	Update README.md	2 months ago
requirements.txt	Update requirements for OSX support	2 months ago
setup.cfg	Added .travis.yml, requirements file, and setup.cfg file. Moved	7 months ago
setup.py	Remove model files from package	2 months ago

Current development version: v0.1.2

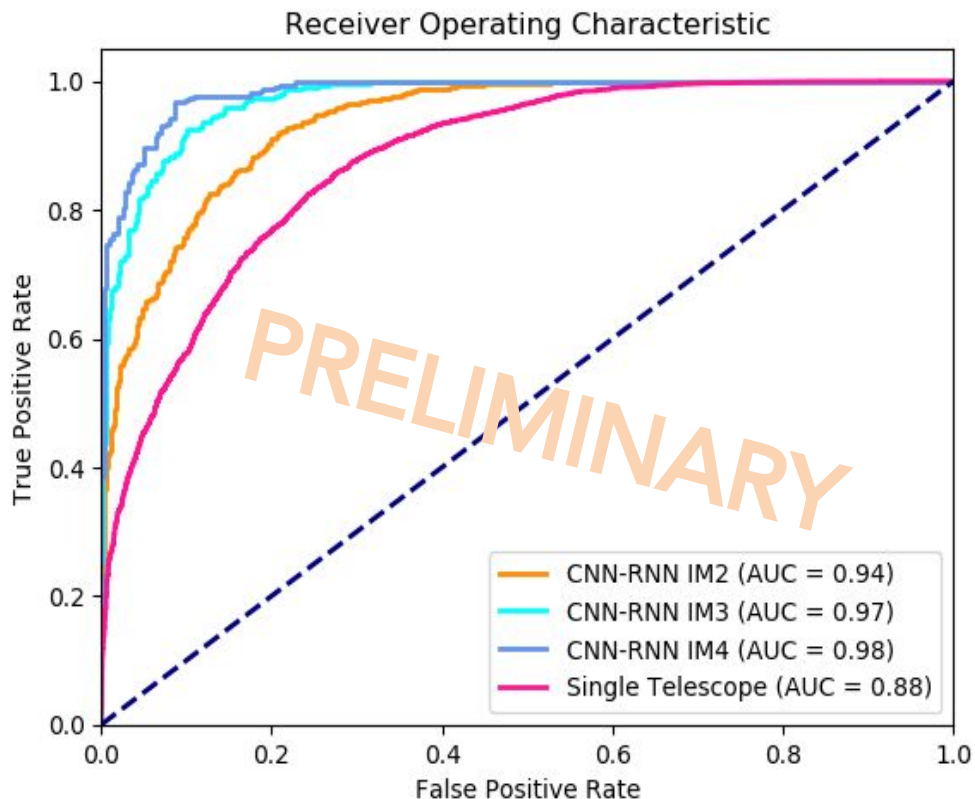
Primary developers (join us!):
Ari Brill, Bryan Kim, Qi Feng (Columbia)
Daniel Nieto (UCM)

<https://github.com/bryankim96/ctalearn>

CNN-RNN



CNN-RNN



- Trained on 200k events each of protons and diffuse gammas from ImageExtractor “ProtoML” dataset
- No quality cuts, no pre-selection of data (except for multiplicity)
- SCT images only
- Preliminary results look promising albeit no hyperparameter optimization

PRELIMINARY

Multiplicity	AUC (test)
1 (basic CNN)	88%
2 (CNN-RNN)	94%
3 (CNN-RNN)	97%
4 (CNN-RNN)	98%

Looking Ahead

Neural networks work well for IACT data but have many challenges remaining

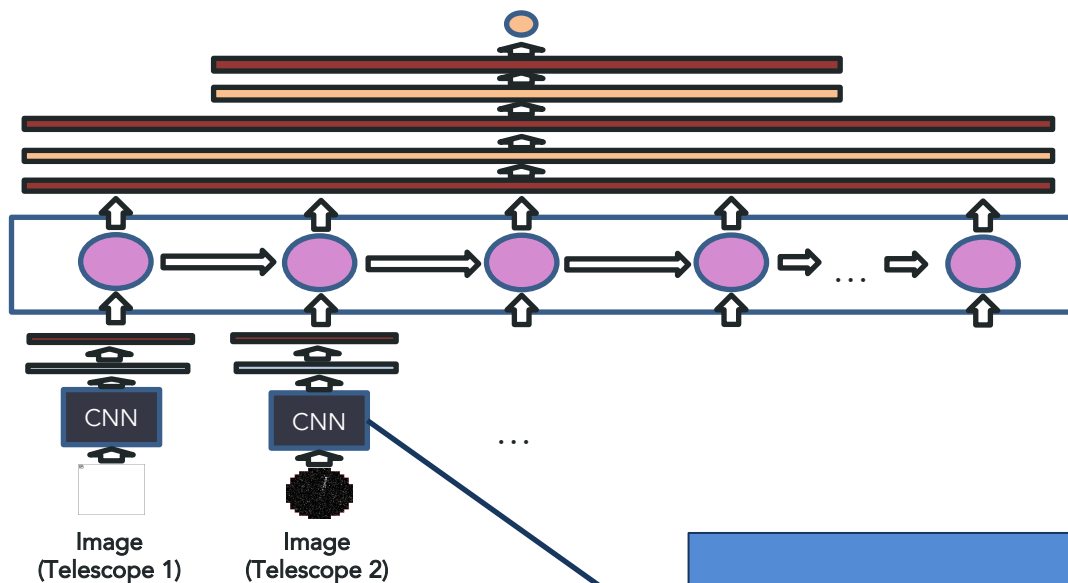
CNN-RNN architecture gives promising preliminary results on gamma/hadron classification with many improvements possible: hyperparameter optimization, inclusion of more telescope types, application to energy/direction reconstruction

CTALearn v0.2 release expected in July:

- Configure user-provided machine learning models
- Load and train with data from any CTA telescope/camera (+ VERITAS)
- Major code upgrades aimed at future use with CTA low-level analysis (ctapipe)
- Code available at github.com/bryankim96/ctalearn

Backup

CNN Block



Basic CNN

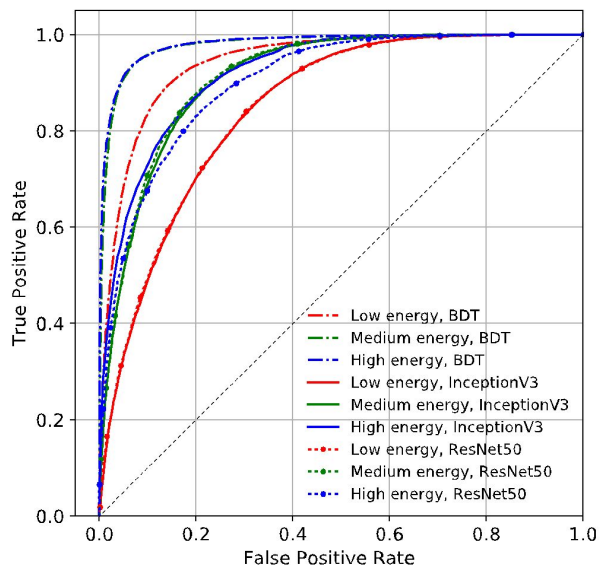
48x48x1 INPUT →
→ 3x3x32 CONV → 2x2 stride 2 MAX_POOL →
→ 3x3x64 CONV → 2x2 stride 2 MAX_POOL →
→ 3x3x128 CONV → 2x2 stride 2 MAX_POOL →

[arXiv:1709.05889](https://arxiv.org/abs/1709.05889)

Accuracy

Model	Low E.	Med. E.	High E.
<i>ResNet50</i>	81.1%	90.1%	91.2%
<i>Inception V3</i>	81.4%	90.1%	91.6%

ROC



06/2017 04/2018

PRELIMINARY

Multiplicity	AUC (test)
1 (basic CNN)	88%
2 (CNN-RNN)	94%
3 (CNN-RNN)	97%
4 (CNN-RNN)	98%

Receiver Operating Characteristic

