USING DEEP LEARNING FOR FACT SOURCE DETECTION

by

A THESIS

Presented to the Department of Computer and Information Science

Department of Physics

and the Robert D. Clark Honors College

in partial fullfillment of the requirements for the degree of

Bachelor of Science

June 2018

An Abstract of the Thesis of

_____for the degree of Bachelor of Science

in the Department of Computer and Information Science

and Department of Physics to be taken June 2018.

Title: Using Deep Learning for FACT Source Detection

Approved: ______________________________

Dr. Joe Sventek

Cosmic rays bombard the Earth constantly, causing air showers that contain information about the original particle and potentially about that particle's source. Determining if an air shower is from a gamma-ray or a hadron is a difficult problem to solve. Current methods primarily use a machine learning technique called random forests to determine whether a given event is from a gamma-ray or hadron, as well as the initial energy and source position in the sky by using the image an air shower makes in a detector. Another type of machine learning algorithm called neural networks has been shown to work very well on tasks involving images, in some cases outperforming random forests. This project aims to improve three tasks: determining the particle's type, energy, and source location using data from the First G-APD Cherenkov Telescope (FACT).

Acknowledgments

I would like to thank the Stamps Family Foundation for financially supporting my entire undergraduate education. I am also grateful to the members of the FACT collaboration for their support and mentorship throughout this process. The support I received from Dr. Tim Cohen, Dr. Joe Sventek, and Dr. Vera Keller was invaluable in helping me complete this thesis.

This work benefited from access to the University of Oregon high performance computer, Talapas.

Contents

1 Introduction

Cosmic rays constantly bombard the Earth, generating air showers that create flashes of light that can be detected by telescopes. Those air showers can inform us about the properties of the source of that particle in the sky if they are from the right type of particle, such as a gamma-ray. The telescopes that detect the output of these showers, Imaging Air Cherenkov Telescopes (IACT), have amassed collections of millions of events with thousands more added during every observation run. The sheer number of events, coupled with both hadron and gamma-ray events creating nearly indistinguishable signals, complicates effective analysis and extraction of useful information, such as the original particle's type, energy, and source location.

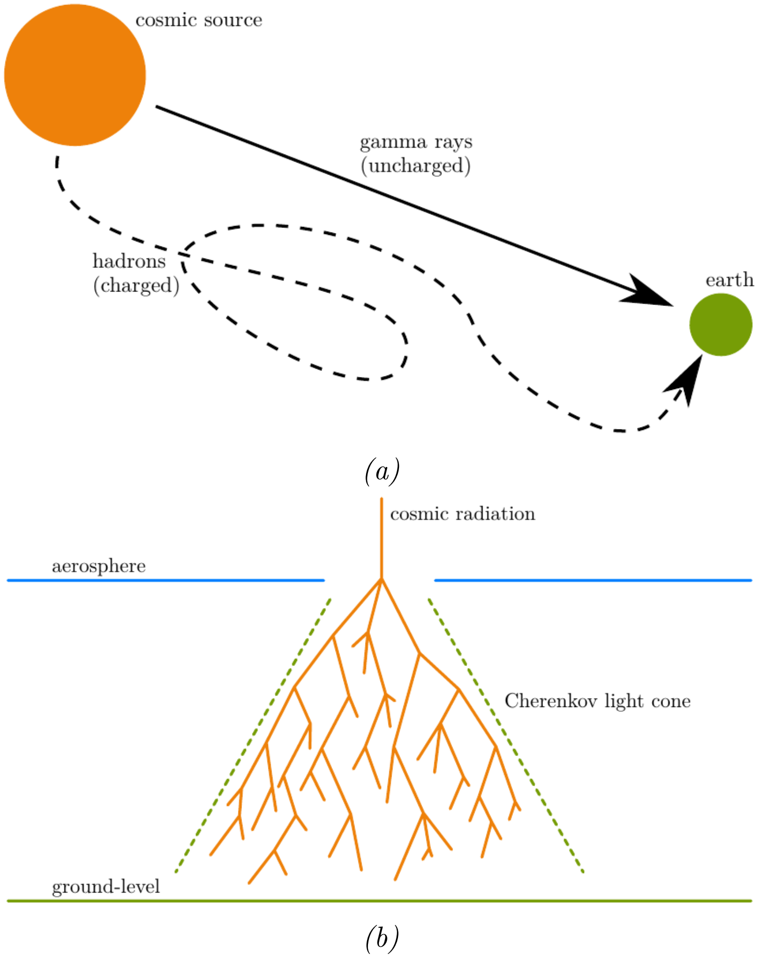
The massive amount of data makes it an ideal problem for using machine learning and, in fact, most IACTs use a type of machine learning called random forests to do so. The random forests classify the events and estimate the source and energy based on features that have been extracted from the images of the events. In this project another type of machine learning called neural networks is applied to this problem to explore whether neural networks can outperform the current approach using the raw images. Convolutional neural networks have been chosen for their success at image analysis. If the neural networks can outperform the random forests, then the turnaround time from the event being detected to the final analysis output would be much shorter because there would be no need to extract the features from the images first. This would be beneficial for detecting transient events and quicker followups on interesting events.

In addition, this project serves to further explore the application of neural networks to single-telescope data of this type. Previous research into neural network applications using IACT image data has been limited to instances where an event appears in multiple telescopes that are arranged in an array, such as the HESS telescope in Namibia, or VERITAS in Arizona [1–3]. The amount of information that is available to the neural networks for this project is significantly limited in comparison because of the single view that is available for each event. If the neural networks can still discriminate between event types and estimate the source and energy correctly, then our discoveries would not be limited by the number of telescopes available.

2 Physics of Cosmic Rays

Cosmic rays are highly energetic particles that enter the atmosphere travelling near the speed of light in a vacuum. There are two types of cosmic rays that are detected by the FACT telescope. The events that are the most useful are gamma-ray events. Gamma-rays are high-energy photons with a frequency of greater than 3x10^{17} Hz and are energetic enough that they are treated as particles. Since gamma-rays are neutral they do not interact with magnetic fields and instead travel in a straight line from the source of the gamma-rays to Earth. This allows astronomers to use gamma-rays to determine properties of the source of those rays. For example, the Crab Nebula emits significant numbers of gamma-rays which are picked up by the FACT telescope and others. Those gamma-rays are then used to characterize the properties of the Crab Nebula.

The other type of cosmic rays that FACT detects are hadron events. These events are initiated by particles such as a proton. Since protons are electrically charged they are affected by magnetic fields from the Earth or other celestial bodies, and as such cannot be traced directly back to their source. In comparison to gamma-rays, hadron events happen an order of magnitude more often[4]. In fact even for a moderately strong gamma-ray source, proton-initiated air showers outnumber gamma-ray events by about 10^5 [5]. An illustration of the different paths taken by hadrons and gamma-rays is shown in Fig. 1a.



(a)

(b)

Figure 1: (a) Gamma-rays and Hadrons as they propagate through space. (b) Illustration of an air shower [6].

Gamma-rays and hadrons are detected by taking advantage of a process called Cherenkov radiation. Cherenkov radiation is light that is emitted by particles that are moving faster than the speed of light in a medium, such as the air or water. For Cherenkov radiation to happen, the particles have to be moving faster than the local speed of light, which is the speed of light divided by the index of refraction for that medium. As long as the particle moves faster than the local speed of light, Cherenkov radiation will be emitted in the direction the particle is moving, creating a narrow cone of light.

When gamma-rays and hadrons enter the atmosphere they not only are moving faster than the local speed of light, but also interact with air molecules. Those interactions produce more particles that produce more light cones. The end result is a shower of Cherenkov radiation that can be detected by instruments such as FACT. An illustration of the process is shown in Fig. 1b. The gamma-ray and proton events create slightly different signals in the detector, enabling the ability to discriminate between the two event types. Simulations of both a gamma-ray air shower and hadronic air shower are shown in Fig. 9. The processes that contribute to air shower development can be split into two rough classes, those that primarily change the momentum of particles in the shower and affect its shape, and those that produce the final photons observed by telescopes on the ground.

2.1 Momentum Processes

These processes spread out the momentum of the primary particle and are the primary drivers of the shape of the shower. Bhabha, Møller, and bremsstrahlung processes only apply to the charged particles in the shower, such as electrons and positrons, while Compton scattering and pair production also affect the momentum of uncharged particles, such as photons.

2.1.1 Bhabha, Møller, and Compton Scattering

There are three types of scattering that occur in air showers. The first, Bhabha scattering, is an electron-positron process, as shown in

$$e^+ + e^- \to e^+ + e^- \tag{1}$$

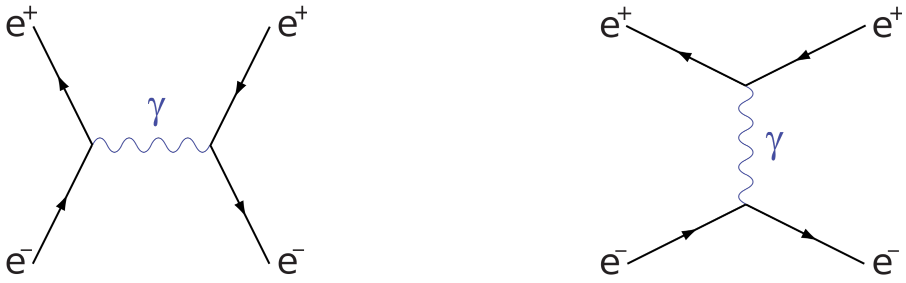


Figure 2: Feynman diagrams for Bhabha scattering [7].

Møller Scattering occurs for electron-electron interactions through

$$e^- + e^- \to e^- + e^- \tag{2}$$

The Feynman diagrams for this scattering are shown in Fig. 3

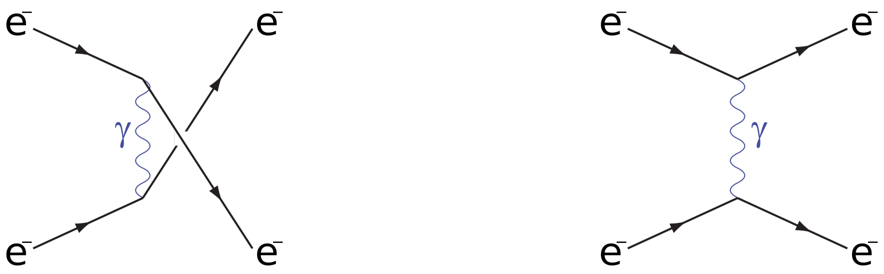


Figure 3: Feynman diagrams for Møller scattering [8].

These two types of scattering are similar except that Bhabha scattering is between a positron and an electron, while Møller scattering involves two electrons. The electrons and positrons in the shower primarily collide with atomic electrons in the atmosphere, causing their momentum to change and spreading out the shower. Both Møller and Bhabha scattering have the same cross section, resulting from the crossing symmetry of the diagrams.

The cross section for Bhabha and Møller scattering is [9]

$$\frac{d\sigma}{d\Omega} = \frac{e^4}{32\pi^2 E_{cm}^2}\left(\frac{1+cos^4\frac{\theta}{2}}{sin^4\frac{\theta}{2}} - \frac{2cos^4\frac{\theta}{2}}{sin^2\frac{\theta}{2}} + \frac{1+cos^2\theta}{2}\right) \tag{3}$$

where θ is the scattering angle, and E_{cm} is the total energy in the center-of-mass reference frame. As $\theta \to 0$, the cross section tends toward infinity. This implies that the cross section is maximized when the particles collide head on.

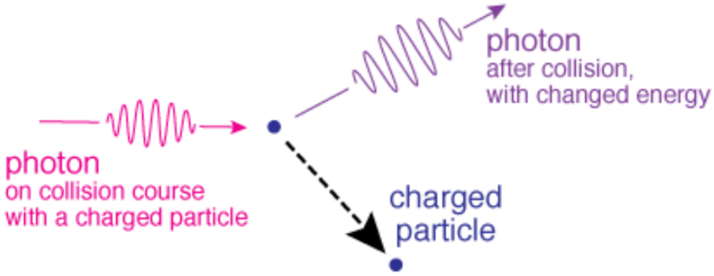


Figure 4: Diagram of Compton scattering [10].

Finally, Compton scattering occurs when a photon scatters off an electron, through

$$\gamma + e^{-} \rightarrow \gamma + e^{-} \tag{4}$$

This results in a photon with lower energy while the rest of the energy is taken by the recoiling electron. In air showers, a gamma-ray undergoes multiple Compton scatterings, losing energy to an electron in each case. Every scattering results in an electron gaining some of the kinetic energy from the gamma-ray and the gamma-ray changing direction. The change in the gamma-ray energy after each scattering is given by [11]

$$E^{'} = \frac{E}{1 + E[1 - cos\theta]} \tag{5}$$

with θ is the angle of scatter for the gamma-ray. The maximum energy loss would result in the scattered energy being

$$E^{'} = \frac{E}{1 + 2E} \tag{6}$$

For large E, such as with the TeV gamma-rays that FACT observes, $E^{'}$ is close to 0.5 [11].

2.1.2 Bremsstrahlung

This phenomenon mostly applies to fast moving charged particles that have a velocity close to the speed of light. When these relativistic moving particles interact with a strong magnetic or electric field, single or multiple photons can be emitted from the particle, which deflects the particle and reduces its energy. In gamma-ray-initiated air showers, the electron and positron generated by pair production interact with the electric field of a nucleus in the atmosphere. The interaction creates more gamma-ray photons that then can undergo pair production with other nuclei.

For an electron of energy E to radiate a photon of energy $k = yE$, assuming the nuclear field is completely screened and the momentum transfer is small, the cross section for bremsstrahlung is approximately [12]

$$\frac{d\sigma_{e\to\gamma}}{dk} \approx \frac{A_{eff}}{X_0 N_A k}\left(\frac{4}{3} - \frac{4y}{3} + y^2\right) \tag{7}$$

where A_{eff} is the effective mass number of the air, X_0 is a constant, and N_A is Avogadro's number. This approximation fails when the screening is incomplete $y \to 1$, and as $y \to 0$, where the Landau-Pomeranchuk-Migdal Effect (LPM) dominates [12].

From these, we can calculate the probability for an electron to radiate a photon with energy in the range of $(k, k + dk)$ when travelling through $dt = dX/X_0$ of the atmosphere as

$$\frac{d\sigma_{e\to\gamma}}{dk}\frac{X_0 N_A}{A_{eff}} dk dt \approx \left(y + \frac{4}{3}\frac{1-y}{y}\right) dy dt \tag{8}$$

From Eq. 8, the total probability for bremsstrahlung is divergent. This divergence is eliminated though by the interference of bremsstrahlung amplitudes from multiple scattering centers, known as the LPM effect [12]. The LPM effect reduces the cross section for both bremsstrahlung and pair production, and effectively increases the mean free path for electrons and photons in the shower. This is described in more detail in section 2.1.4. An energy scale E_{LPM} can be defined where the LPM effect is significant, while below that threshold, the energy loss from bremsstrahlung is approximately

$$\frac{dE}{dX} \approx -\frac{1}{X_0}\int_0^1 yE\left(y + \frac{4}{3}\frac{1-y}{y}\right)dy = \frac{E}{X_0} \tag{9}$$

This gives a constant $X_0 \approx 36.7\frac{g}{cm^2}$ as the radiation length in air. This is the mean distance over which a high-energy electron loses $1/e$ of its energy or 7/9 of the mean free path for pair production by a high-energy photon [12, 13].

2.1.3 Pair Production

Pair production is the process by which a high energy photon near a nucleus produces an electron/positron pair. In this process, the photon is completely annihilated and creates an electron/positron pair through

$$\gamma + X \rightarrow e^+ + e^- + Y \tag{10}$$

Represented as a conservation of total energy, the photon has to have enough energy hf that [14]

$$hf = 2(m_{e_{rest}}) + K(e^-) + K(e^+) \tag{11}$$

where $m_{e_{rest}}$ is the rest energy of an electron, or 0.511 MeV, and $K(e^\pm)$ is the kinetic energy of the electron or positron. This means that pair production cannot occur if the photon involved has an energy less than 1.02 MeV. Pair production can also only happen near a nucleus as the momentum of the photon has to be absorbed. The momentum p of the photon is related to its wavelength λ and Planck's constant h by $p = h/\lambda$. For example, if the photon's energy is exactly equal to 1.02 MeV, then the positron and electron will be created at rest. The momentum will have to be absorbed by a nearby nucleus. If the photon has an energy higher than 1.02 MeV, then the excess energy is turned into the kinetic energy of the electron and positron [14].

Similar to bremsstrahlung, the cross section for a photon undergoing pair production, in which one of the created particles has energy $E = xk$, can be approximated as [12]

$$\frac{d\sigma_{\gamma\rightarrow e^+e^-}}{dE} \approx \frac{A_{eff}}{X_0 N_A}\left(1 - \frac{4y}{3} + \frac{4x^2}{3}\right) \tag{12}$$

where A_{eff} is the effective mass number of the air, X_0 is the radiation length in air, and N_A is Avogadro's number.

The probability for a photon undergoing pair production with the electron energy in the range of $(E, E + dE)$ is then [12]

$$\frac{d\sigma_{\gamma \to e^+e^-}}{dE}\frac{X_0 N_A}{A_{eff}} dEdt \approx \left(1 - \frac{4x}{3} + \frac{4x^2}{3}\right) dxdt \tag{13}$$

From that, the total probability for pair production per unit of X_0 can be determined from integration,

$$\int \frac{d\sigma_{\gamma \to e^+e^-}}{dE}\frac{X_0 N_A}{A_{eff}} dE \approx \int \left(1 - \frac{4x}{3} + \frac{4x^2}{3}\right) dx = 7/9 \tag{14}$$

In air showers, once the photon undergoes pair production, the positron from the electron/positron pair generally annihilates with an electron to produce two gamma-rays, which can undergo further Compton scattering or photoelectric absorption.

2.1.4 Landau-Pomeranchuk-Migdal Effect

This effect lowers the cross sections of both pair production and bremsstrahlung once the particles are above an energy threshold E_{LPM}. The effect only becomes noticeable for photons and electrons with energies above $E_{LPM} \sim 10^{10}$ GeV [12]. This means that for hadronic showers, since much of the energy is initially contained in the hadronic interactions, the effect does not become important until the initial particle has an energy above 10^{12} GeV.

For the LPM effect to occur, it requires low momentum transfer between the nucleus and the electron (q):

$$q = \sqrt{E^2 - m^2} - \sqrt{(E-k)^2 - m^2} - k \sim \frac{m^2}{2E(E-K)} \sim \frac{k}{2\gamma^2} \tag{15}$$

Since in air showers γ is high and the emitted photon energy k is low, q is low. Because q is low, the interaction must occur over a large distance or formation length L_f, from the uncertainty principle. If the mean free path $\sim L_f$, then the emissions cannot be independent and instead they interact with each other. These interactions between emissions is what suppresses the cross sections for bremsstrahlung and pair production, as the probability for multiple interactions to occur together is necessarily lower than each interaction on its own[15].

For electromagnetic showers, there is also a competing effect, the geomagnetic field. For gamma-rays with energies above E_{LPM}, the shower will have already started in the geomagnetic field before reaching the atmosphere. Once the gamma-ray undergoes pair production, the e^+e^- pair emits synchrotron photons, resulting in a photon preshower that is peaked below 10^{10} GeV and so reduces the effect that LPM has on shower development [12]. These competing effects result in large fluctuations in the value of X_{max} for the air shower [12].

2.1.5 Pion Decay

Pions, or pi mesons, are subatomic particles made up of a quark and antiquark. They form an integral part of hadronic air shower development. In electromagnetic air showers, the pion component is miniscule, only appearing from gamma-air interactions that produce charged pions. In both cases charged pions are responsible for the creation of muons in the air showers and the continuation of the hadronic shower, while neutral pions, which do not live long enough to interact with other particles, redirect energy to electromagnetic subshowers in hadronic showers and away from the hadronic core [12].

Charged pions, unlike neutral ones, only have around a 10% chance of decaying before interacting with other particles as long as the pion's energy is above $E \approx 1$ TeV. Because of this, charged pions are the primary way in which a hadronic air shower continues. In a hadronic shower, the first few generations of pions interact with the atmosphere again, making a hadronic core in the shower that is surrounded by the electromagnetic and muonic parts [12].

2.2 Energy Loss Processes

There are three main processes by which air showers convert their energy into the photons that FACT sees. The majority is from Cherenkov radiation from the relativistic particles in the shower and the ionization of the atmosphere, although muon decay also contributes.

2.2.1 Cherenkov Radiation

Cherenkov radiation is a phenomenon that occurs when a charged particle is moving faster than the speed of light in a medium. The speed necessary for Cherenkov radiation to occur is determined by the index of refraction of the medium. The index of refraction for air at standard temperature and pressure is roughly 1.0003, resulting in the speed of light in air being roughly $c_{air} = c/n_{air} = c/1.0003 = 0.9997c$. Therefore, any particle that is moving faster than 0.9997c in the atmosphere will start to emit Cherenkov light. The light is emitted in a small cone that moves in the direction of the particle. Specifically, the opening angle Θ of the cone is determined by

$$\cos\Theta = \frac{1}{\beta n} \tag{16}$$

where n is the refractive index of the material, in this case 1.0003, and β is $\frac{v}{c}$ where v is the velocity of the particle and c is the speed of light in the medium. For cosmic rays, the opening angle is always less than about 1.4°[16]. With roughly $10^8 - 10^9$ charged particles in a given air shower at its maximum extent, the showers produce large amounts of ultraviolet Cherenkov radiation in overlapping cones of light that travel towards the ground. This light is what is detected by imaging air Cherenkov telescopes such as FACT. Depending on where the first interaction occurs in the atmosphere and the type of air shower, the Cherenkov radiation can make a ring of light on the detector or be more localized in a clump [16, 17].

2.2.2 Ionization

Air showers ionize parts of the atmosphere that the shower travels through from a combination of the secondary electrons, positrons, and gamma-rays undergoing Møller, Bhabha, and Compton processes, as well as electron-positron annihilation. Electron-positron annihilation occurs through

$$e^+ + e^- \rightarrow \gamma + \gamma \tag{17}$$

In air showers, even though all four processes conserve the total charge, many of the electrons in the air shower scatter off of electrons in atmospheric atoms [18].

Bhabha, Møller, and Compton scatterings accelerates those atomic electrons into the electromagnetic shower, while electron-positron annihilation reduces the absolute number of positrons in the shower at the same time. This results in the shower ionizing the atmosphere and the air shower having an asymmetry in charge [18]. In total, around 80-95% of the primary energy is converted into ionization energy by the time the shower is finished [13].

2.2.3 Muon Decay

Muons are an elementary particle with a charge of -1 and a mass 207 times that of the electron. They are essentially heavier versions of electrons, but with a mean lifetime of 2.2 μs before they decay [19].

In air showers muons are mostly generated through the decay of charged pions and so depend strongly on the initial baryonic content of the original particle [12]. They can also be generated directly from photons through muon pair production. This direct generation is much less likely to occur than electron pair production because electron pair production has a cross section $\approx$ 43000 larger than muon pair production. Once created, muons also have much smaller cross sections for pair production and bremsstrahlung than electrons due to their larger mass, leading to muons arriving earlier to the ground than most of the shower and suffering fewer scattering interactions [12].

High energy muons lose energy through muon-nucleus interactions, bremsstrahlung, knock-on electron production, and e^+e^- pair production [20]. Knock-on electron production occurs when a very energetic charged particle knocks orbiting electrons out of atoms. Those ejected electrons, if they are given enough energy by the muon, can further ionize the atmosphere along their path by interacting with subsequent atoms themselves [21]. Of these processes, knock-on electron production can be considered continuous because of its short mean free path, while the others are discrete from their large mean free path [12]. For bremsstrahlung, the cross section for muons is lowered by a factor of $(m_e/m_\mu)^2$ with respect to the electron bremsstrahlung cross section in Eq. 8. Muon energy loss is therefore mostly dominated by knock-on electron production and by pair production for muons above 1 GeV, with bremsstrahlung becoming more dominant than pair production below 1 GeV. Energy loss from muon-nucleus interactions is smaller than from bremsstrahlung and so does not contribute much to the total energy loss [12]. The relative mean free paths of the three dominant processes is shown in Fig. 5.

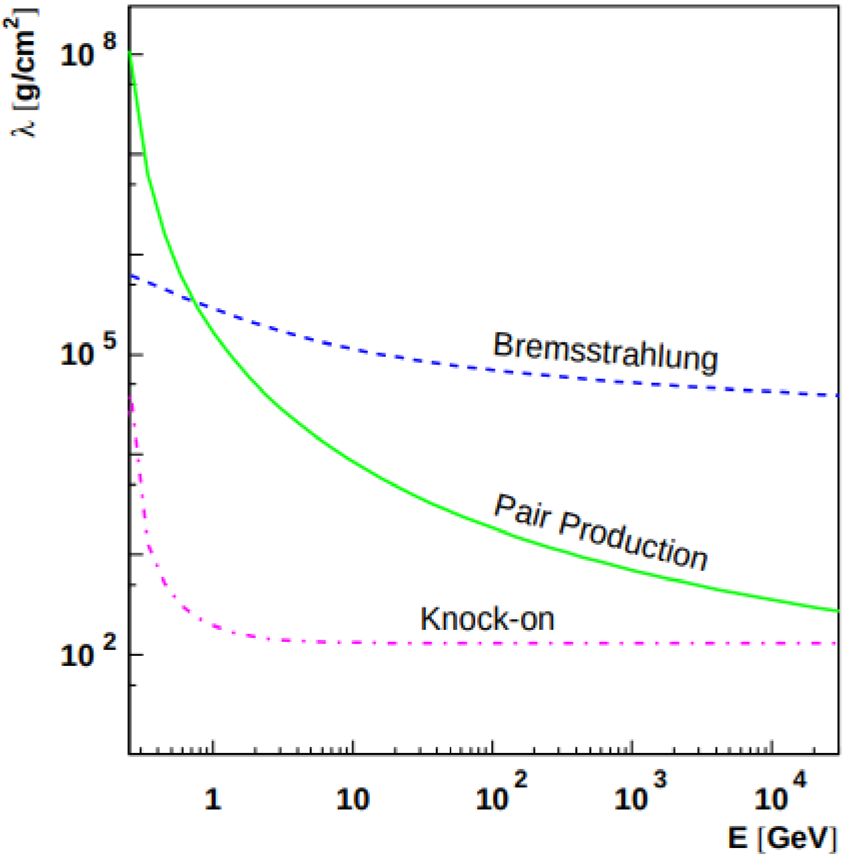


Figure 5: Mean free path in air for different muonic interactions as a function of initial kinetic energy [12].

2.3 Energy Spectrum of Cosmic Rays

The energy spectrum of cosmic rays spans a huge energy range, going up to 10^{20} electronvolts (eV). The flux across this energy range follows a power law proportional to $E^{-2.7}$ [22]. There are three changes to the power law that occur across the energy range. At 4x10^{15} eV, the spectrum steepens and again at 8x10^{18} eV, called the so-called knees. At $10^{18.5}$ eV, the spectrum flattens out in the so-called ankle [22].

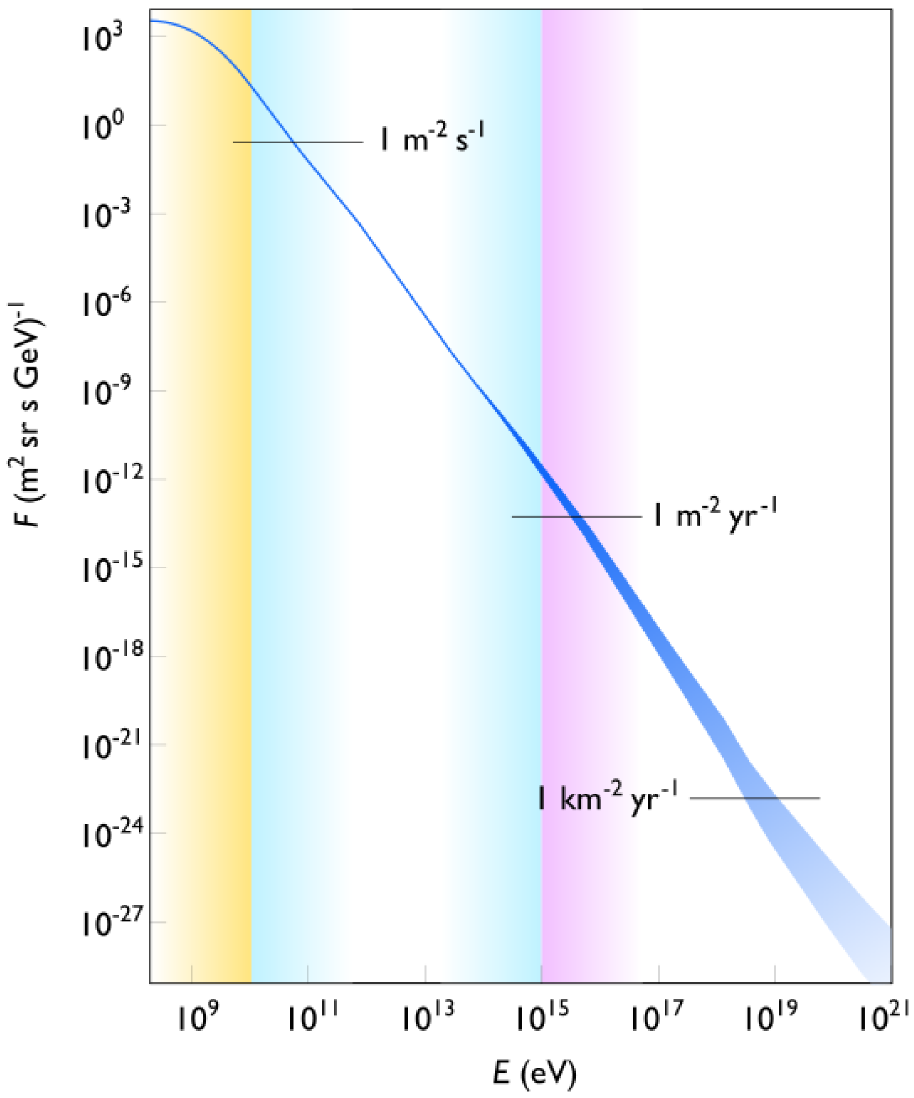


Figure 6: Cosmic ray flux versus particle energy [23].

3 Gamma-Ray-Initiated Air Shower

Gamma-ray-initiated air showers are dominated by the processes of bremsstrahlung, pair production, and positron-electron annihilation. A gamma-ray air shower is started by a very energetic photon that interacts with a particle in the atmosphere, creating an electron-positron pair through pair-production. Following this first interaction, the positron and electron interact with other particles in the atmosphere, creating photons through bremsstrahlung and positron-electron annihilation. These processes create more high-energy photons, kicking off smaller subshowers as those photons undergo pair-production and repeat the process. This continues until the particles in the shower do not have enough energy to undergo such interactions and the process stops. A diagram of a gamma-ray-initiated air shower is shown in Fig. 7.

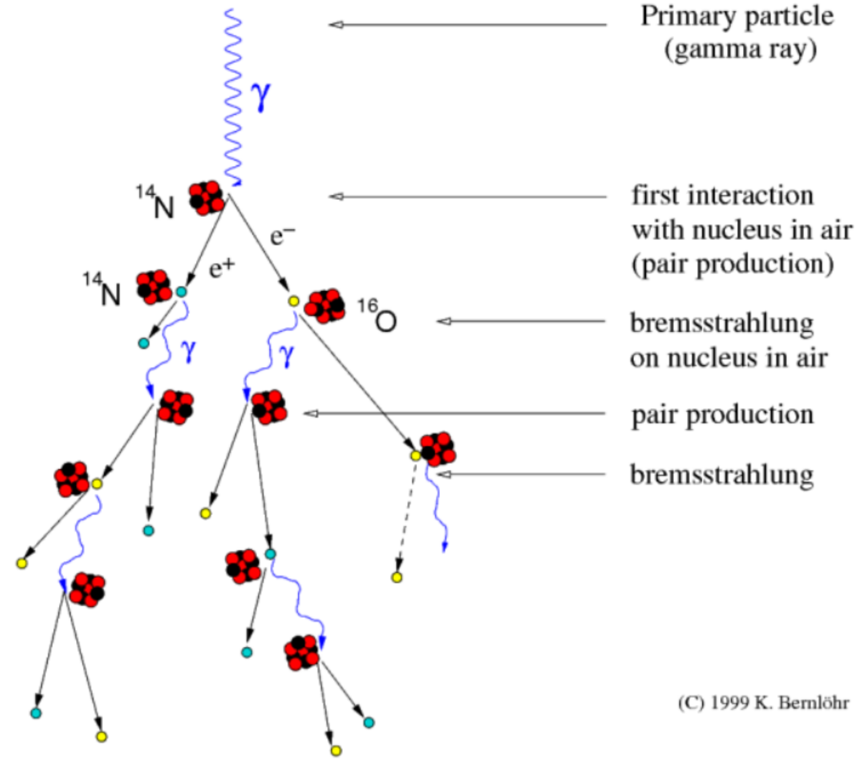


Figure 7: Diagram of a gamma-ray-initiated air shower [24].

One way to model electromagnetic air showers is by using the Heilter model of electromagnetic air showers [13]. In the model, the primary particle is the photon. After n interactions in the atmosphere, there are 2^n particles in the shower. An interaction occurs every X/X_0 where X_0 is the interaction length of the particle in the medium and X is the atmospheric depth of the particle in $\frac{g}{cm^2}$, given in section

2.1.2 as $X_0 \approx 36.7 \frac{g}{cm^2}$ [12, 13].

Heitler's model assumes that the shower maximum is reached when $E(X) = E_c$, where E_c is the critical energy when energy loss processes, such as ionization, dominate over interactions, such as bremsstrahlung and pair production. For air, $E_c \approx 85$ MeV [12, 13, 25]. This gives a maximum number of particles in the shower as $N_{max} = E_0/E_c$ [12, 13].

Putting that together, we get the shower's maximum size, X_{max}, as approximately

$$X_{max} \approx X_0 \frac{\ln(E_0/E_c)}{\ln 2} \tag{18}$$

While the Heitler model offers a good approximation, there are some corrections needed to obtain a more accurate picture of an electromagnetic air shower. The most important refinement is the ratio of positrons and electrons to photons in the shower, as that ratio changes the development of the shower. The model overestimates the ratio of positrons and electrons to photons as $N_{e^\pm} \approx 2/3 N_{total}$, while in reality the number of photons is around a factor of six larger than the number of $e^\pm$ [25]. This is mainly because in bremsstrahlung, multiple photons can be emitted, not just the single one that the model assumes. To roughly obtain the number of electrons in the actual shower, a closer approximation is $N_{total}/10$ [25]. In addition, the LPM and geomagnetic effects produce large fluctuations in X_{max}, although this model's predictions do fall within the range of the fluctuations [12].

4 Hadron-Initiated Air Shower

Hadron-initiated air showers are dominated by pion interactions, bremsstrahlung, and pair production. A hadron-initiated air shower first starts when the hadron interacts with a particle in the atmosphere. That first interaction is the most energetic of the whole air shower, usually destroying both particles in the collision. The collision generates many new particles, consisting of mostly pi-mesons, or pions. The pions can be charged, like the proton, or neutral. If they are neutral, they quickly decay to two gamma-rays, while the charged pions last longer. Because the charged pions do not immediately decay into gamma-rays, some of the charged pions last long enough to collide with other particles in the atmosphere. These collisions start secondary cascade reactions similar to the first hadron-collision, although with less overall energy. The two gamma-rays that come from each neutral pion can also start their own electromagnetic subshowers of particles. Those subshowers follow the way gamma-ray-initiated air showers develop [26]. The decay of the neutral and charged pions affect the shape of hadronic air showers, causing them to be more spread out and "lumpier" than those of gamma-ray-initiated ones [17, 26]. A diagram of a hadronic air shower is shown in Fig. 8.

Similarly to a gamma-ray-initiated air shower, hadronic air showers can be modelled by what is called the Heitler-Matthews model. In hadronic showers, the cascade ends with the particles decaying when the energy of the particle drops below the decay energy E_{decay}, where $E_{decay} = E_0/(n_{tot})^n = E(X)$. Here, $n_{tot} = n_{charged}+n_{neutral}$, which is the total number of pions in the shower, both charged and neutral. Since

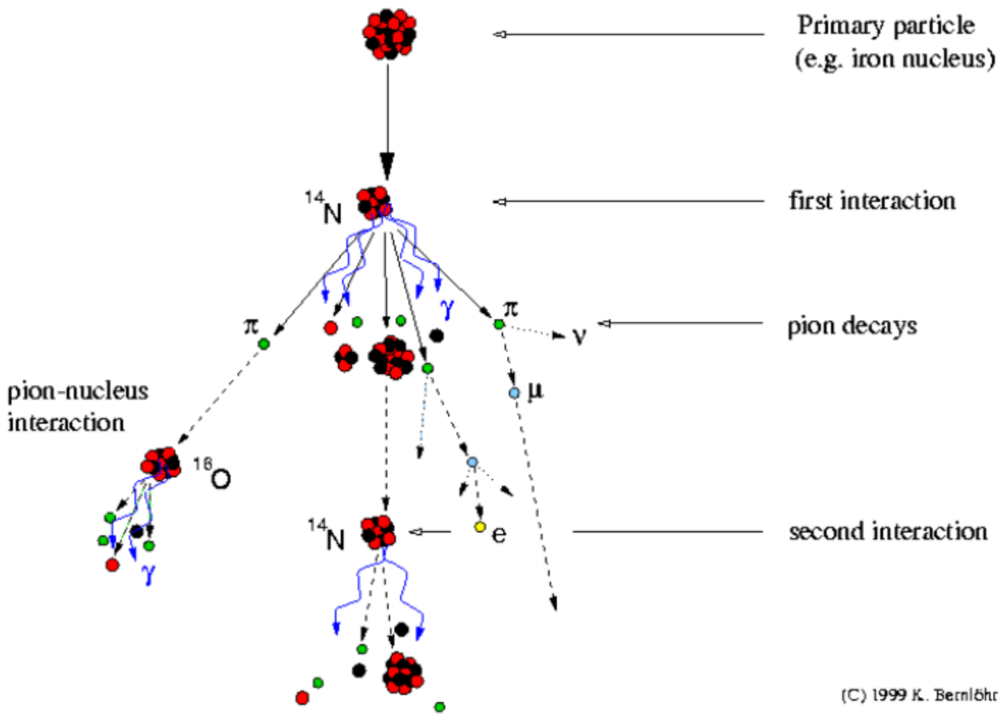
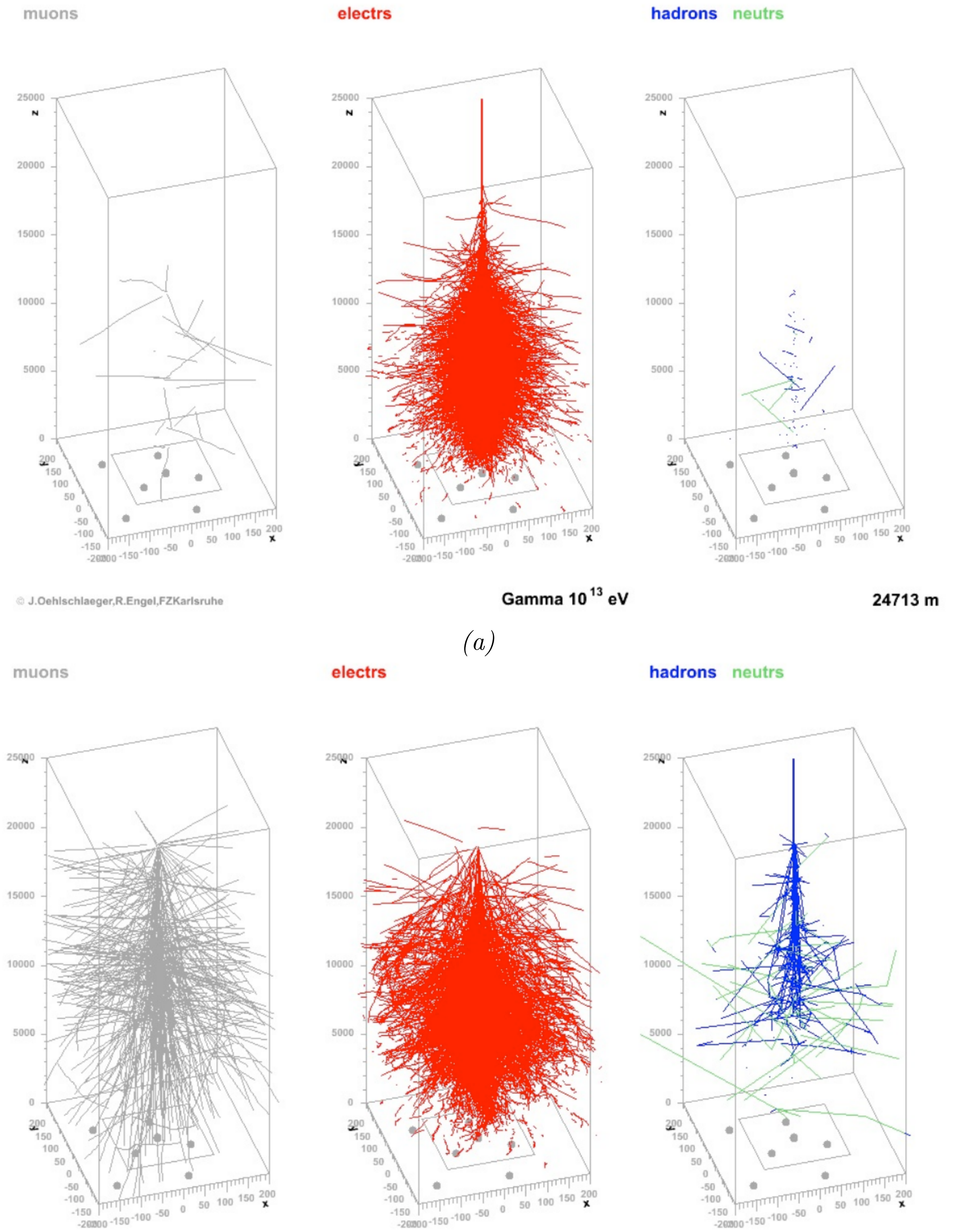


Figure 8: Diagram of a hadron-initiated air shower [24].

only charged pions can last long enough to initiate new hadronic cascades, the number of particles in a purely hadronic cascade can be modeled as n^n_{charge} after n interactions, while the energy per particle after n interactions is $E_0/(n_{tot})^n$ [13]. At each stage of particle production, the hadronic part of the cascade loses a third of the energy to the electromagnetic shower through the decay of neutral pions [25]. As the shower develops, the amount of energy in the hadronic part of the shower goes towards $(\frac{2}{3}(n_{tot})^n)E_0$ because each collision between a hadron in the shower and a particle in the atmosphere creates two charged pions, which continue the hadronic part of the shower, and a neutral pion, which almost immediately decays into gamma-rays that start electromagnetic subshowers. Those electromagnetic showers contain $(1 - 2/3(n_{tot})^n)E_0$ energy after n interactions [13, 25]. For charged pions, once their individual energies drop below E_{decay}, they become more likely to decay before they interact again, stopping the development of more subshowers. Instead the charged pions decay to muons, which are assumed to hit the ground [25].



(a)

(b)

Figure 9: (a) Tracks of particles in a simulated gamma-ray-initiated air shower. (b) Tracks of particles in a simulated hadron-initiated air shower.

5 Air Showers in the First G-APD Cherenkov Telescope (FACT)

Located in La Palma in the Canary Islands, the FACT telescope is the first operational IACT of its kind using a camera equipped with a hexagonal grid of silicon photomultipliers (G-APD aka SiPM) to primarily detect gamma-ray-initiated air showers [4]. FACT has been operational since 2011 and during that time has performed near-continuous observations of the night sky. Once an air shower has converted the original particle's energy into photons through processes outlined in Sec. 2.2, those photons are reflected off the mirrors of FACT to be recorded as an image on a detector.

5.1 Events in FACT

Each air shower creates a Cherenkov light cone that can cover an area hundreds of meters in diameter. Because of that, FACT and telescopes like it generally will not see the entire event but only obtain a partial view of a given shower. An illustration of that process is shown in Fig. 10. Examples of both a gamma-ray event and hadron event in FACT are shown in Fig. 11.

As can be seen, gamma-ray and hadron events are quite similar, but do have some morphological differences. Those differences can be parameterized to differentiate between the two event types. These are called the Hillas parameters, a collection of shape and orientation parameters. The shape parameters are the size, length, and width of the image. The size is defined as the total number of photons detected

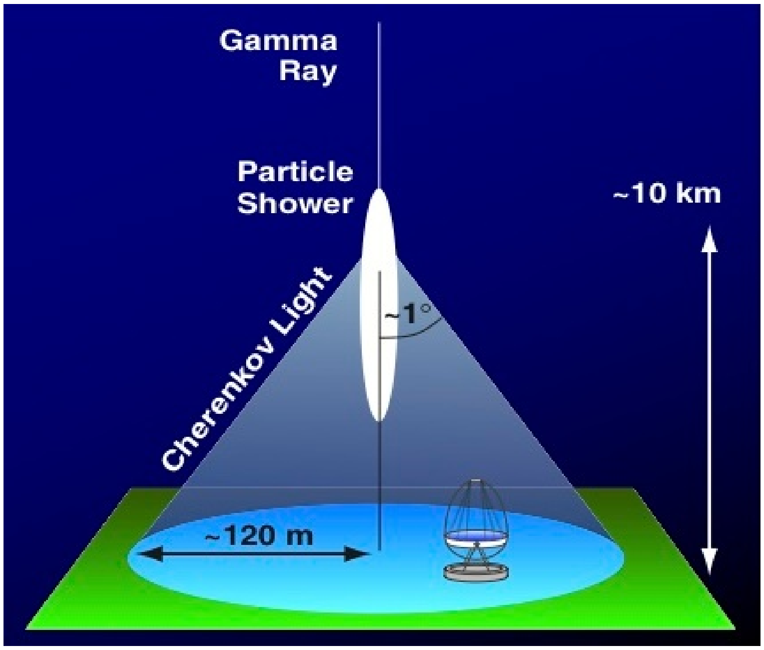
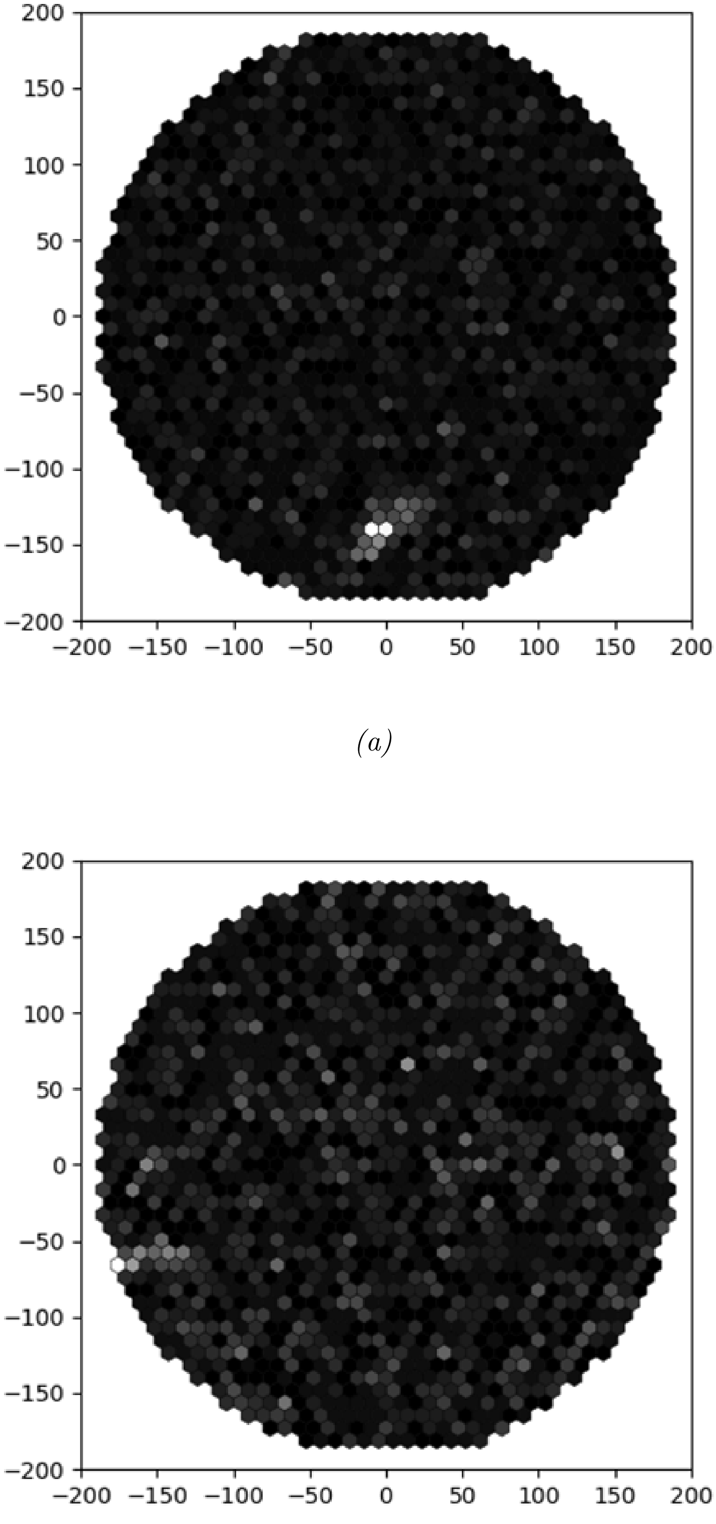


Figure 10: Cherenkov Radiation. [27]

in the shower and depends on the energy of the primary particle. The length is the root mean squared spread of light along the major axis of the image and is used to parameterize the longitudinal development of the shower. The final shape parameter, the width, is the root mean squared spread of light along the minor axis of the image and is related to the latitudinal development of the shower.

The orientation parameters are composed of the dist and alpha parameters. Dist is the distance from the camera's center of field of view to the center-of-gravity of the shower. Alpha is the angle between the major axis of the image and the radius drawn from the center of the camera to the center of the image. These parameters are also used for determining the source of the shower. These parameters are shown in Fig. 12.



(a)

(b)

Figure 11: (a) Gamma-ray event in the detector (b) Hadron event in the detector.

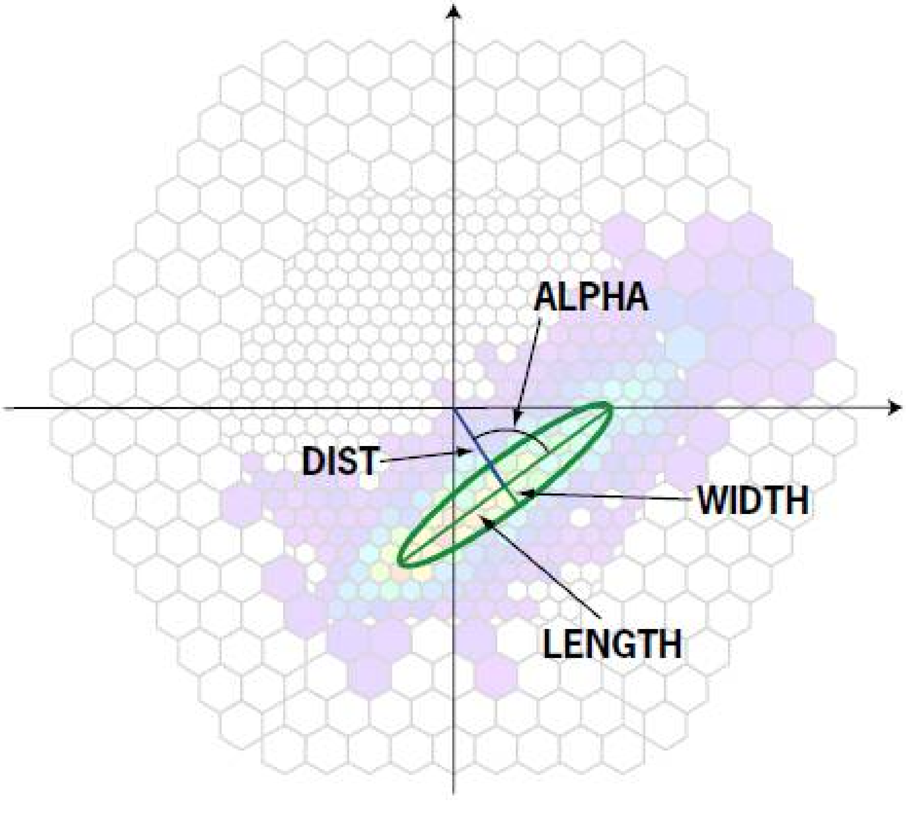


Figure 12: Hillas Parameters. [27]

5.2 From Image to Source

To go from the raw image to the primary particle's type, energy, and source position is a multistep process. The first step is to differentiate between the hadron and gamma-ray events. Because it is difficult to determine whether an air shower's primary particle is a gamma-ray or hadron, most analysis starts by using simulated events to determine what each type of event should look like in FACT's detector. Using these simulated events, a classifier is trained to differentiate between the two particle types and is then applied to the observation data. Similarly, for determining the original particle's energy and source location, regressors are trained on the simulated data to predict primary particle energy and source location given an image.

5.3 Current Analysis Pipeline

The current pipeline starts with generating simulated gamma and hadron events using the CORSIKA [28] simulation package. With CORSIKA, gamma and hadron events and their interactions with the atmosphere are simulated as well as how the events would appear in FACT's detector. After the simulations are generated, the events are preprocessed using FACT-Tools [29] to extract the Hillas parameters and calculate secondary and tertiary features, such as the number of clumps of photons in the image called islands, the number of photons in the outer ring of photomultipliers called the leakage, and other physically useful features for later use in the pipeline. The last step before the random forest analysis starts is the splitting of the simulation data into training and testing sets.

The FACT collaboration uses a random forest classifier to perform gamma/hadron separation. A more detailed view of random forests is presented in Sec 6.1. The classifier uses a total of 200 decision tress, each looking at 15 different features, to estimate the "gammaness" of each event. The classifier outputs a number between 0.0 and 1.0 indicating the confidence that a given event is a gamma-ray event. To minimize false positives, only events with a confidence of 0.85 or higher are considered gamma-ray events. All others are considered proton events. For energy regression, the same type of random forest is used. Instead of giving a confidence value between 0 and 1, the regressor outputs an estimate of the original particle's energy.

The final step in the process is the source determination. To do that, one classifier and one regressor are both trained on a different dataset than the previous two random forests. The new dataset is composed of diffuse gamma-ray events where the source location is spread over a large area of the sky. This process uses the Disp method for finding the source location. This method takes advantage of the fact that for gamma-ray events the image in the detector usually points to the source position along the length axis. In this method, the distance between the center-of-gravity of the image and the source position is calculated. In addition, the sign of the distance is also predicted to determine which side of the image the source is on. In this step, the random forest regressor is trained to predict the disp parameter, while the classifier is trained to determine the sign of the disp. The source position in the camera's coordinate system can be determined from these two values and the center-of-gravity of the image. From the camera coordinates, the source position in the sky can be determined by transforming the camera's coordinate system into the equatorial coordinate system.

6 Machine Learning

Both random forests and neural networks are examples of machine learning. Machine learning is the process where software is designed so that it can improve its predictions with more experience. That experience comes from feeding in inputs with known labels and having the model learn how to predict the correct output.

6.1 Random Forests

Random forests are collections of decision trees. Decision trees are a way to classify or regress on input based on predefined features. A decision tree is composed of branches where a true or false question is asked of the input. If the input satisfies the condition it goes down one side of the branch, if not, it continues down the other. The final output is made when the input reaches one of the leaves of the decision tree, where the value at the leaf becomes the tree's prediction for the input data. The decision tree learns what features are important for its task through training on labelled data where the outcome is known [30].

Random forests are constructed by creating an ensemble of decision trees and taking the mean value of their outputs. The benefits of the random forest over a single decision tree is that the random forest helps prevent overfitting. With a single decision tree, it can quickly learn to predict the training dataset perfectly, but then become increasingly poorer at predicting on any other data. A random forest helps to mitigate the overfitting, as it uses the inputs from multiple different trees for the final determination [31].

Random forests have the benefit of being white boxes, as in all of the features and information that it uses to make a decision is known and set by the programmer. For tasks where certain features are known to be important for final prediction, such as the Hillas parameters in gamma/hadron separation, the user can help the random forest's performance by ensuring that it is learning on the correct features that they want.

6.2 Neural Networks

Neural networks are a type of machine learning that has exploded in popularity in recent years because of their ability to perform image classification and other tasks with very high accuracy [32]. Neural networks are composed of a collection of nodes, called neurons, arranged in different layers. The neurons interact by signaling other neurons they are connected to if they become activated. The neurons learn to activate on specific features in the data on which they are trained, creating a network that learns the important features for a classification or regression task on its own. This makes a neural network a black box, in contrast to the random forest. Since the network learns the features on its own, if the training data has features that the non-training data does not, the network can learn on the wrong features.

6.2.1 Building Blocks of Neural Networks

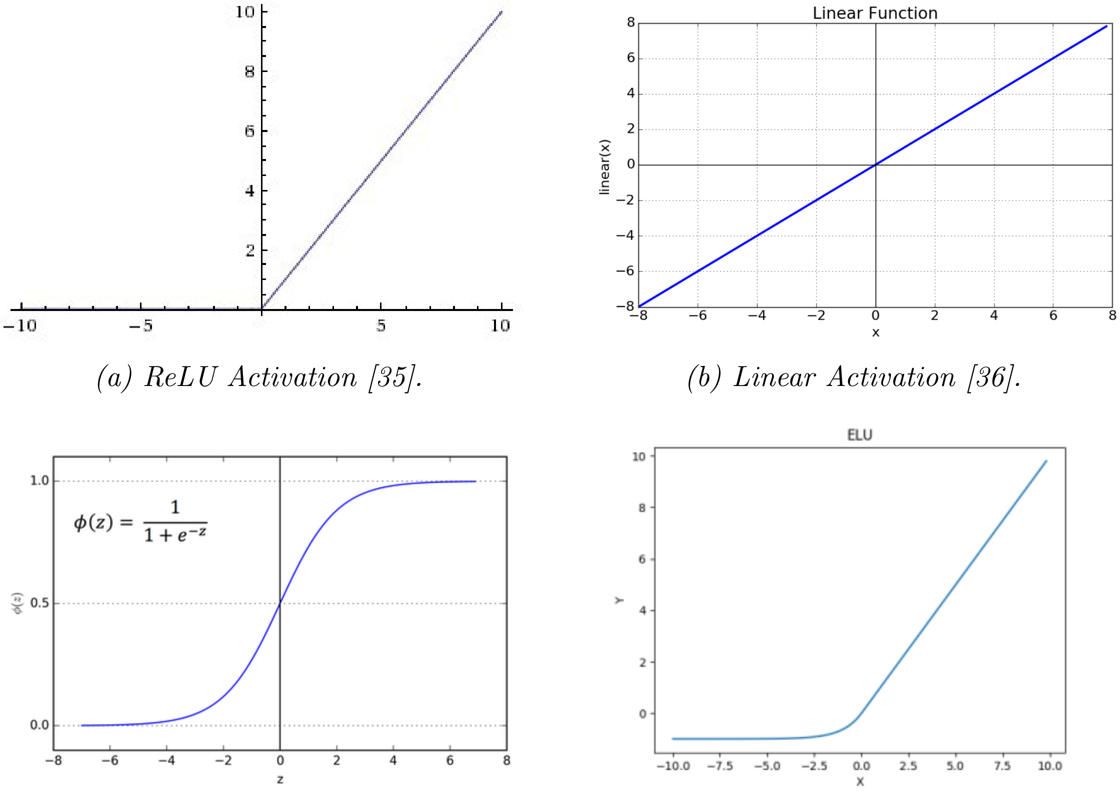
Neural networks are built on artificial neurons. Each neuron is made up of a set of inputs, set of weights, an optional bias, and an activation function. A neuron takes its input and obtains a value by $\sum(inputs * weights) + bias = output$. The output is then fed into an activation function to determine if the neuron is considered activated or not. The output from the neuron in one part of the network can then be passed into later parts, making a given neuron's output the input for a neuron in the next layer of the model [33].

The activation function takes the output of the neuron and maps the output to the range of 0 to 1 or -1 to 1, depending on the function [33]. For this project, linear, softmax, Rectified Linear Unit (ReLU), and Exponential Linear Unit (ELU) activations were used, shown in Fig. 6.2.1. The ReLU and ELU activations are usually used as the activations for within the network, while the linear and softmax activations are used as the activations for the output layer. The ReLU activation is described by $Output = max(0, Input)$, while the ELU is a modified version of the ReLU, described by

$$Output = \begin{cases} \alpha(e^{input} - 1) & input < 0 \\ \\ input & input \geq 0 \end{cases}$$

where α is a parameter, usually set to one [34]. The linear activation is simply $Output = c * Input$, where c is some constant. The softmax activation is based on the sigmoid activation, which is described by $Output = \frac{1}{1+e^{-input}}$. As can be seen in

Fig. 13c, the sigmoid activation pushes the input to either end of the function and the output is bound between 0 and 1, making it ideal for classification tasks.



(a) ReLU Activation [35].

(b) Linear Activation [36].

(c) Sigmoid Activation [36].

(d) ELU Activation [37].

These structures are called neurons because they can learn over time. To teach the neurons, backpropogation is used. In this, the network is given an input and an expected output. The input is fed through the network and gives a predicted output. If the network's output is different than the expected output, the network modifies the weights for all the neurons so that the next time the same input is given to it, it predicts a value closer to the expected value. The amount by which the weights are changed for iteration is called the learning rate.

Neurons are generally organized into layers that do different tasks. Convolutional layers form the basis for convolutional neural networks. These types of layers work by looking at individual parts, or patches, of the overall image. As the kernel, a matrix of real numbers, is convolved across the entire image, the neurons in the layer learn the filters that activate when there is some type of visual input, such as an edge, color, or larger patterns [38]. Each neuron in this type of layer only receives inputs from a smaller region of the entire image, and so learns based on the local differences in inputs.

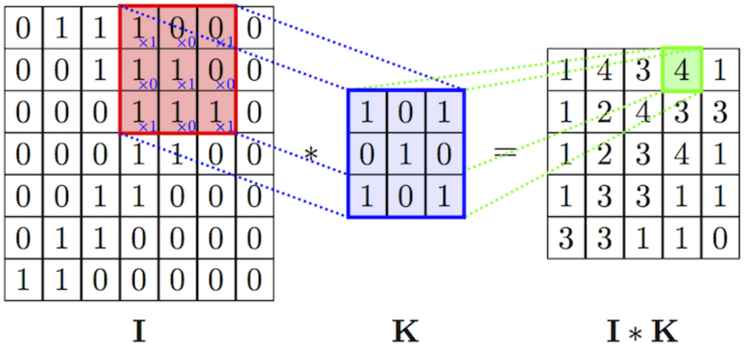


Figure 14: Convolution layer example [38].

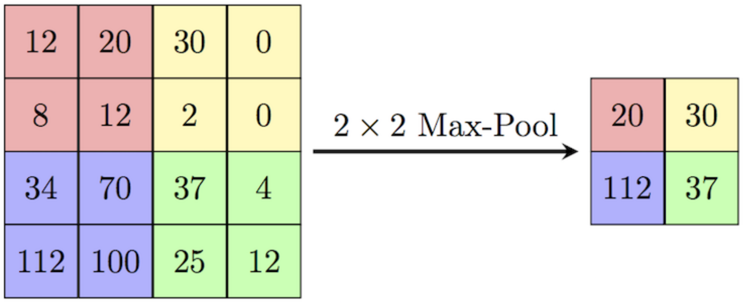


Figure 15: Pooling example with MaxPooling [38].

The primary purpose of the convolution is to extract features from the input images. Through the use of small patches, the spatial relationship between features is kept. As the kernel slides over the image, at every position in the patch, the element-wise multiplication between the kernel and the pixels the patch overlays occurs, as shown in Fig. 14. The sum of that multiplication gives the value for one element in the output matrix [38].

Pooling layers work by reducing the dimensionality of the output from previous layers but still keeping the most important information [39]. Given a spatial region to look at, such as a 2x2 pixel window, this layer takes the maximum value in each window, as shown in Fig. 15. The reason for doing this is to reduce the number of parameters in the network to reduce overfitting, while still keeping the most important features on which to train [39]. Pooling layers also function to make the network more robust against rotations or shifts in the input data.

A fully connected layer, or dense layer, is a layer where every neuron is connected to every neuron in the layer before it and after it. The purpose of these layers is to assign labels to the input for classification tasks, or to reduce the output of previous layers down to N output numbers for regression tasks, where N is the number of neurons in the fully connected layer [39].

Dropout layers are used to reduce the overfitting of a network to the training data. This type of layer works by randomly removing the connections between neurons in the layer before the dropout layer and the layer after so that some information is lost. That lost information helps to ensure that the network can generalize to other data. A dropout layer with a dropout rate of 1.0 would remove all connections between the two layers, while one with a rate of 0.0 would not remove any connections [40].

The loss is what the neural network works to minimize and is the measure by which the network measures its effectiveness. For classification problems, such as gamma/hadron separation, the loss function is generally the cross-entropy. The cross-entropy for a binary classification problem is defined as

$$CE = -(y \ln(p) + (1-y)\ln(1-p)) \tag{19}$$

where y is either one or zero depending if the label c is the correct classification and p is the predicted probability that the observation is part of class c [41].

For regression problems, such as energy reconstruction or source finding, the mean squared error tends to be used as it punishes predictions that are far from the true value much more than those that are close to the true value. The mean squared error is defined as

$$MSE = (y_{pred} - y_{true})^2 \tag{20}$$

6.3 Replacing Random Forests with Neural Networks

Gamma/hadron separation is a difficult task. Both hadron and gamma-ray air showers can create similar images and previous research has shown the difficulties of performing gamma/hadron separation with convolutional neural networks for FACT [6]. Other research with IACT arrays has suggested different image preprocessing steps can improve the classification to outperform random forests [1–3].

The energy of the primary particle is an important measure for characterizing the particle's source and that source's energy spectrum. Since the energy of the primary particle is heavily correlated with the number of photons in the shower [22], a convolutional neural network applied to the images of showers should be able to estimate the energy fairly well. Previous work on energy estimation for other

IACT telescopes has been successful at using neural networks on raw photon counts [2, 42].

The goal of source detection is to either find new sources of gamma-rays in the night sky or to increase the significance of the signals for sources that have already been found. As FACT is unable to detect new sources of gamma-rays in comparison to the much larger telescope arrays currently running, this part of the project attempts to increase the significance of the currently detected sources using the same amount of data. An increase in significance would come from a neural network better estimating the source positions of the incoming particles. For FACT, this takes two forms. In the first, the convolutional network would attempt to directly estimate the source position. This is the approach taken in other source detection research [1]. The second option would be to estimate the disp value and sign so that the source position can be calculated using those two values and the center-of-gravity of the image. This is the approach that the FACT analysis pipeline currently uses.

7 Methods

7.1 Datasets

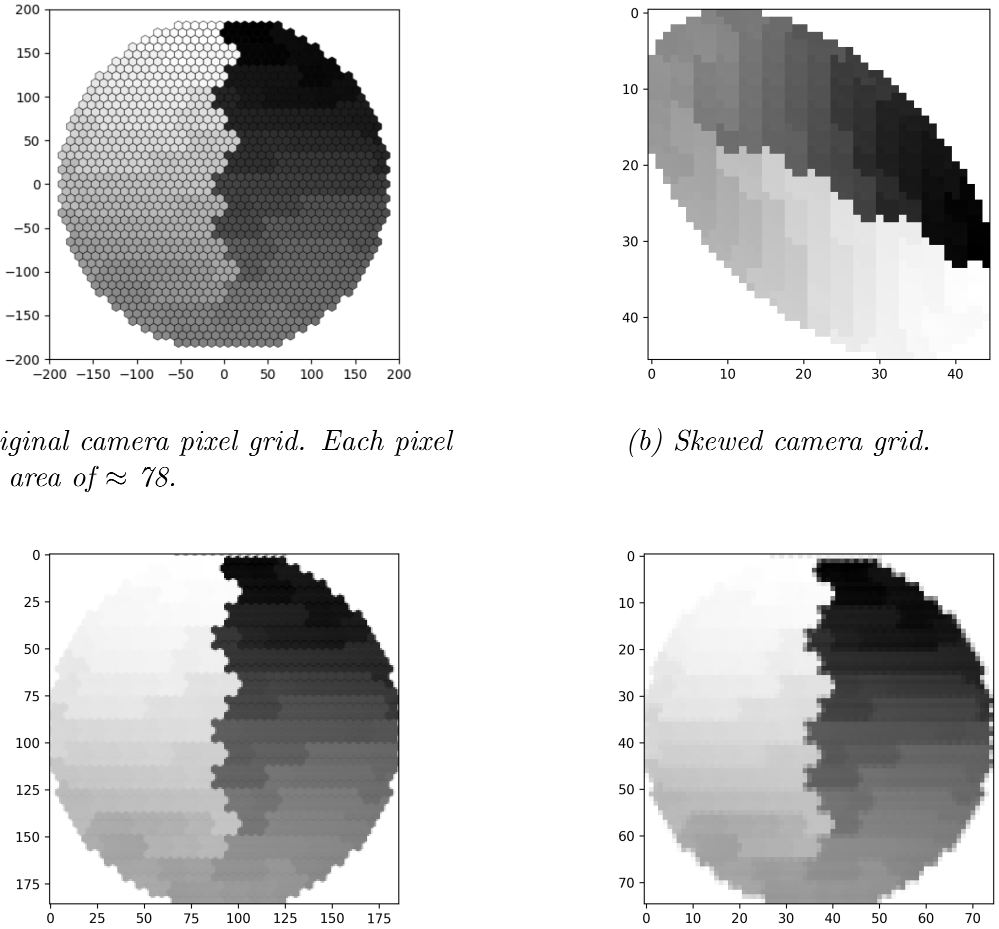
There are three datasets that have been used for this project. The first is the simulated event dataset used by the current classifier, consisting of 529000 diffuse gamma-ray events, 431000 point source gamma-ray events, and 130000 hadron events. The gamma-ray events range in energy from 200 GeV to 50000 GeV. The second is Crab

Nebula data, consisting of 674000 events taken from Crab Nebula observation runs in 2013 and 2014 [43, 44]. The final dataset consists of 448000 events from observations of Markarian 501 in 2014. Both observational datasets are from very bright gamma-ray point sources.

7.2 Data Preprocessing

One of the issues with using the raw images is the physical layout of the detector's pixels. The pixels are hexagonal in nature, and so cannot be directly used as input for the deep learning library used, Keras [45]. There are multiple methods to convert the physical detector grid into something that can be used. One, used in previous neural network research for FACT, shifts the hexagonal detector into a more ellipse shape as show in 16b. The benefits of this approach was that the resultant image is fairly small, 46 by 45 pixels, and quick to load into the software. On the other hand, by shifting the hexagonal shape into a quadratic shape, information is lost or distorted, including the shape of the image and neighboring pixel information. Other research that used this type of conversion include previous research into energy regression [42] and event classification [46–49].

The other method used was inspired by previous research into deep learning for IACTs, where instead of shifting the hexagons a grid of squares is overlaid on top of the detector representation. Then the amount of overlap between each square and hexagon is computed and the pixel's final value is computed from the sums of the fractions of the hexagons that it overlaps as shown in Fig. 17. This



(a) Original camera pixel grid. Each pixel has an area of ≈ 78.

(b) Skewed camera grid.

(c) Camera rebinned with squares of area 4. The distortion of the camera is almost nonexistent.

(d) Camera rebinned with squares of area 25. The shape is still kept better than the skewed mapping, but more detail is lost.

Figure 16: Various camera mappings. Pixels are colored according to their hardware ID. For rebinned grids, the pixels are colored by the sum of the percentage of each overlapping physical pixel.

method has the benefit of limiting the distortion of the shape of the image in the detector. A comparison of how the different types of preprocessing affect the detector representation is shown in Fig. 16. With a 75 by 75 pixel grid, the conversion stretches the camera grid by roughly 2% in the y-direction and not at all in the x-direction compared to the original hexagonal grid.

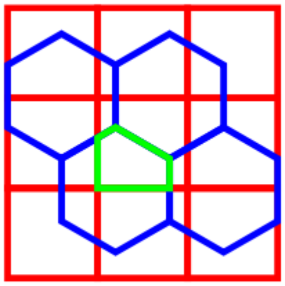


Figure 17: Example of rebinning hexagonal grid to square grid. The green outline is the fraction of that hexagon assigned to the overlaying square pixel [2].

The images used for source determination and gamma/hadron separation had one more step of preprocessing applied. For these, the images were scaled so that the image had a mean intensity of zero and standard deviation of one. This was done to accelerate convergence of the models and to ensure the networks rely significantly on the shape of the image and not the overall photon count [1, 22]. Images used for energy regression did not have this step applied because of the correlation between the total photon count and the initial energy of the particle [2, 22].

The preprocessed images were loaded into HDF5 files to facilitate quick and easy loading of events for the analysis. For all images, the pointing azimuth and zenith of the telescope and center-of-gravity of the image are included. For simulation events the initial energy of the particle from CORSIKA was added, while for observation events the assumed source position, night, run ID, event number, and UTC time of the observation were included.

7.3 Gamma/Hadron Separation

There are two different tasks that were performed in this project regarding gamma/hadron separation. In the first step various neural networks were trained on simulated and observational data to determine which architecture worked best for gamma/hadron classification. To determine the best type of convolutional network for the task, multiple different architectures were tested. The first was based on previous research into gamma/hadron separation with neural networks for FACT [6]. The next was a VGG-style network chosen because of this type of network's success at image classification problems including muon event classification [3]. The cross-entropy loss function was used for all models. Diagrams of these architectures are shown in Fig. 28. The next step was to take the best neural network trained on the simulated data and retrain it with Crab Nebula observation data. Since there is no set ground truth for the Crab Nebula events, in this project an assumption that all events that are predicted to come from within a θ^2 cut[1] of 0.025 degrees of the Crab Nebula is a gamma-ray event, while those outside that value are classified as hadron events. This θ^2 cut was chosen because it is the threshold value used to decide whether an event came from a given source or not.

These architectures were first trained on 200000 simulated events in a one-to-one ratio of gamma-ray to hadron events. The dataset was further divided into an 60% training, 20% validation, and 20% testing datasets. Each network was trained for 500 epochs with early stopping if there was no improvement on the validation data after 15 epochs. Each network also underwent hyperparameter tuning, where the rate of dropout between layers, the number of neurons in each layer, and the number of layers was varied to determine which architecture performed the best.

[1] The square of the angular difference between the reconstructed air shower position and the source position.

The best architecture was then trained on Crab Nebula data. Because of the limited number of events (~ 45000) within the θ^2 cut used, the assumed gamma-ray event dataset was augmented by flipping vertically, horizontally, and rotating in 90°increments. Combinations of the all three augmentation were used to increase the number of training events to ~ 270000 assumed gamma-ray events. An equal number of assumed hadron events were added to the dataset to make a one-to-one ratio of events.

7.3.1 Performance Evaluation

The performance of a classification task can be measured using the rate of true positives TP, true negatives TN, false positives FP, and false negatives FN. The number of true positives and false negatives can be combined to give the true postive rate, $TPR = \frac{TP}{TP+FN}$. Similarly, the number of true negatives and false positives can be combined to give the false positive rate $FPR = \frac{FP}{TN+FP}$ [50]. The TPR and FPR can then be plotted against each other to give a curve, called the Receiving-Operating-Characteristic (ROC) curve. The area under the curve (AUC) then gives a metric as to how well a given model classifies input. If a model has an AUC of 0.5 then it predicts no better than chance, while an AUC of 1.0 would be a perfect classifier.

7.4 Energy Estimation

To estimate the initial energy of the particle a purely convolutional neural network composed of a varying number of convolutional layers and a single dense output layer was used, based off the success of this type of network on multi-telescope energy regression [2]. The mean squared error was used as the loss function. This network was trained and tested on the simulated point-source gamma-ray events. Because there is no simple way to estimate the energy of an observation for training without using the random forest's estimations, no training was performed on observation data. If the random forests energy estimations on observation events were used as training data, then the neural networks would learn the values that the random forest estimated. This puts a upper limit on how well the neural network could perform in comparison to the random forest. Since the neural network's ground truth is the random forest's estimation, the neural network will not achieve a better performance than the random forest.

7.4.1 Performance Evaluation

The performance of a regression task can be found by the coefficient of determination R^2, which measures how well the estimation x_{est} matches the true value x_{true} [22]. With N test events, the R^2 value can be calculated using

$$R^2 = 1 - \frac{\sum_i^N (x_{true_i} - x_{est_i)^2}}{\sum_i^N (x_{true_i} - \overline{x}_{true)^2}} \tag{21}$$

where $\overline{x}_{true} = \sum_i^N x_{true_i}$ [22]. If $R^2 = 1$, then the regression task was perfect at estimating the original value, while anything less than 1 indicates errors in the regression.

7.5 Source Estimation

For this task there were two approaches taken. The first approach was to directly estimate the source position in the camera's coordinate system. Two types of neural networks were tested, one that minimized the loss over the combined source x and y coordinates and one where the x and y values had separate losses. These networks used architectures based on the successful architecture used on HESS multi-telescope events [1] and shown in Fig. 31. For the second approach, the disp and sign were calculated with a regressor and classifier respectively. The neural network for the disp regression was composed of multiple convolutional layers, one pooling layer, and a small number of dense layers to ensure that as much spatial information as possible was used for the disp estimation. For the sign classifier, neural networks with multiple convolutional and pooling layers and four dense layers were used. Diagrams of the different architectures are shown in Fig. 10. The direct source and disp regressors minimized the mean squared loss on the outputs, while the classifier used the cross-entropy loss to determine its success. All the networks were trained on the simulated diffuse gamma-ray events with known source positions.

In a similar vein to the energy regression, there are no easy ways of estimating the source position of observation data without using a regressor based on simulated data. For that reason, these networks were only trained and tested on simulation data.

8 Results and Discussion

The overall results are mixed. For gamma/hadron separation, when trained on simulated data, the convolutional neural network performs worse than the random forest classifier. Furthermore, when the neural network is trained on Crab Nebula observation data, the network does not predict better than random chance. For energy estimation, the neural network approaches the performance of the random forest, but does not achieve the same results. For the final step of source determination, predicting the source coordinates directly was unsuccessful, while the neural network outperformed the random forest on estimating the disp and sign of the disp, defined in Sec. 5.3.

8.1 Gamma/Hadron Separation

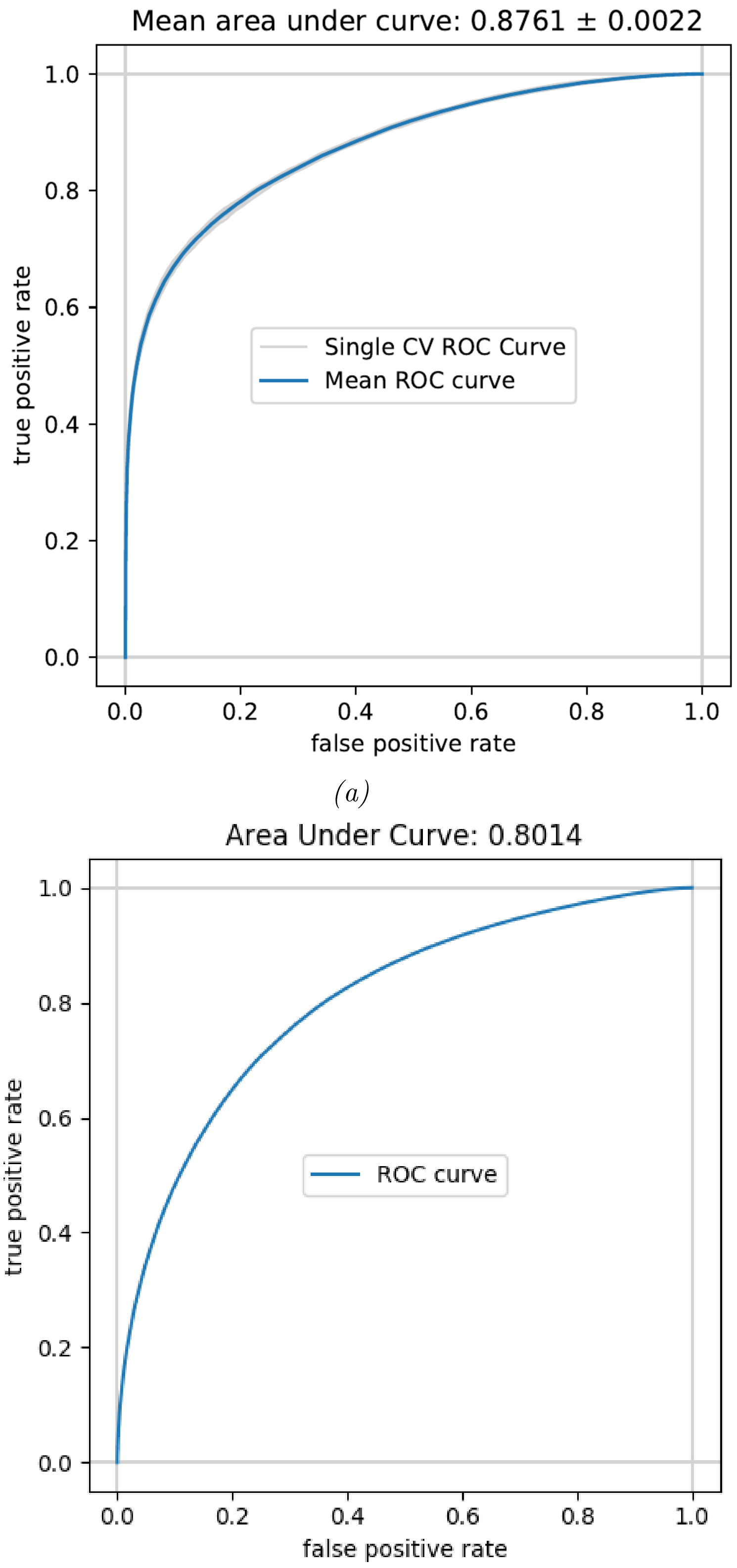
The gamma/hadron separation task showed the difficulties neural networks have with respect to classifying very similar images. For example, although the VGG-style network has been used successfully for many types of image classification tasks, including separating muons from background events [3], the network failed to distinguish between gamma-rays and hadrons. It obtained an AUC of 0.53 on the training data, and 0.5 on the testing dataset.

The testing of multiple architectures for the gamma/hadron separation resulted in a confirmation of previous results. The network with the highest AUC was composed of three convolutional layers with pooling layers in-between and three dense layers, as shown in Fig. 28a [2, 6]. The AUC of this model tested on simulated data matches that of previous work for FACT gamma/hadron separation [6], but still falls short of the AUC for the random forest classifier.

When the same neural network was then trained on Crab Nebula observation data, the network could not distinguish between simulated gamma-ray events and simulated hadron events. When the Crab Nebula trained network predicted on both the simulated events and on Markarian 501 observation data, the network did no better than chance, as shown in Fig. 19.

The most successful neural network and the default classifier relied on different features for determining gammaness. Both classifier predicted on Markarian 501 events and all events classified as a gamma-ray event with a confidence of 0.95 or higher were compared. The random forest had 2634 events that fulfilled the criteria, while the neural network estimated 868. In comparing those events, it became apparent that the neural network classified events using different criteria than the random forest. The neural network's events had an average 26% larger length, 40% more islands in the image, 530% higher leakage, and a 26% larger width. Overall, the neural network estimated events that were more spread out, more likely to be on the edge of the detector, and "lumpier" than the random forest. These characteristics skewed the events the neural network classified as gamma-ray events as more hadronic than the random forest's predictions. This suggests that the random forest classified events that are more likely to be actual gamma-ray events than the neural network. The neural network trained on Crab Nebula data did not classify any events with a higher than 0.02 confidence, and so is not included in this comparison.

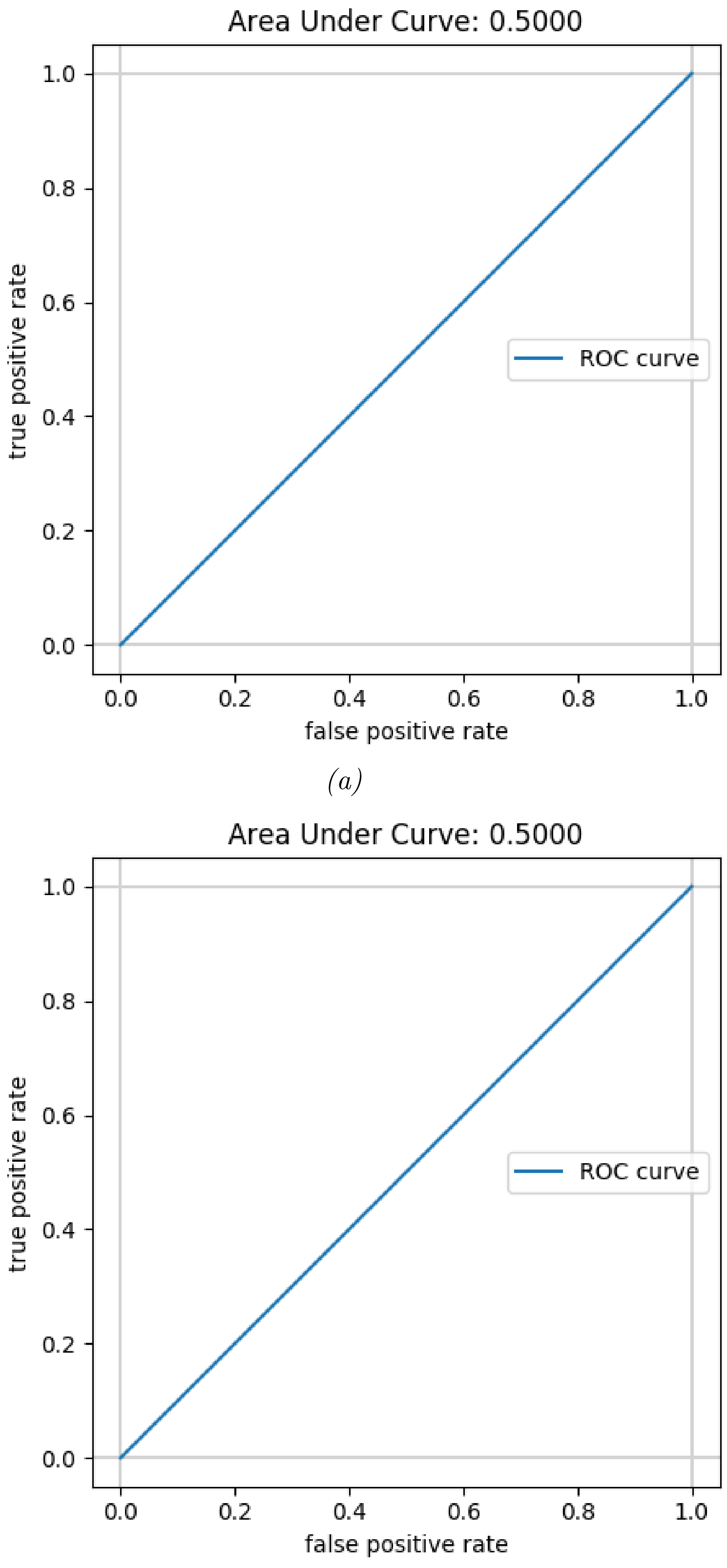
As can be seen in Fig. 20, the best performing neural network distinguished between simulated protons and simulated hadrons but was not able to separate them as clearly as the random forest did. The same seemed to occur for the Markarian 501 observation data, as shown in Fig. 21. The random forest assigned most of the events as protons, as is expected in observations since the vast majority of events detected are hadron events. The neural network, on the other hand, had a much smaller spike and was much more likely to classify events as gamma-ray events. Both classifiers had a bump around 0.6 confidence and had similar overall shapes, demonstrating their overall similar predictions. In comparison to the two simulation trained classifiers, the neural network trained on Crab Nebula observation data did much worse. The Crab Nebula network gave the same predictions for essentially all of the events, both simulated and Markarian 501, as shown in Fig. 22. This suggests that the Crab Nebula trained classifier was either not given enough data on which to train or the assumption that there were enough actual gamma-ray events within 0.025 θ^2 of the Crab Nebula is incorrect and instead the network trained on essentially two sets of hadron events. To explore this, the neural network was again trained on Crab Nebula events, this time using a θ^2 cut of 0.001. The results were unchanged. Smaller θ^2 values gave too few events on which to train ($\sim$ 275 for θ^2 = 0.0001), and while data augmentation was used, the total number of events could only be increased by six times without distorting the shape of the image.



(a)

(b)

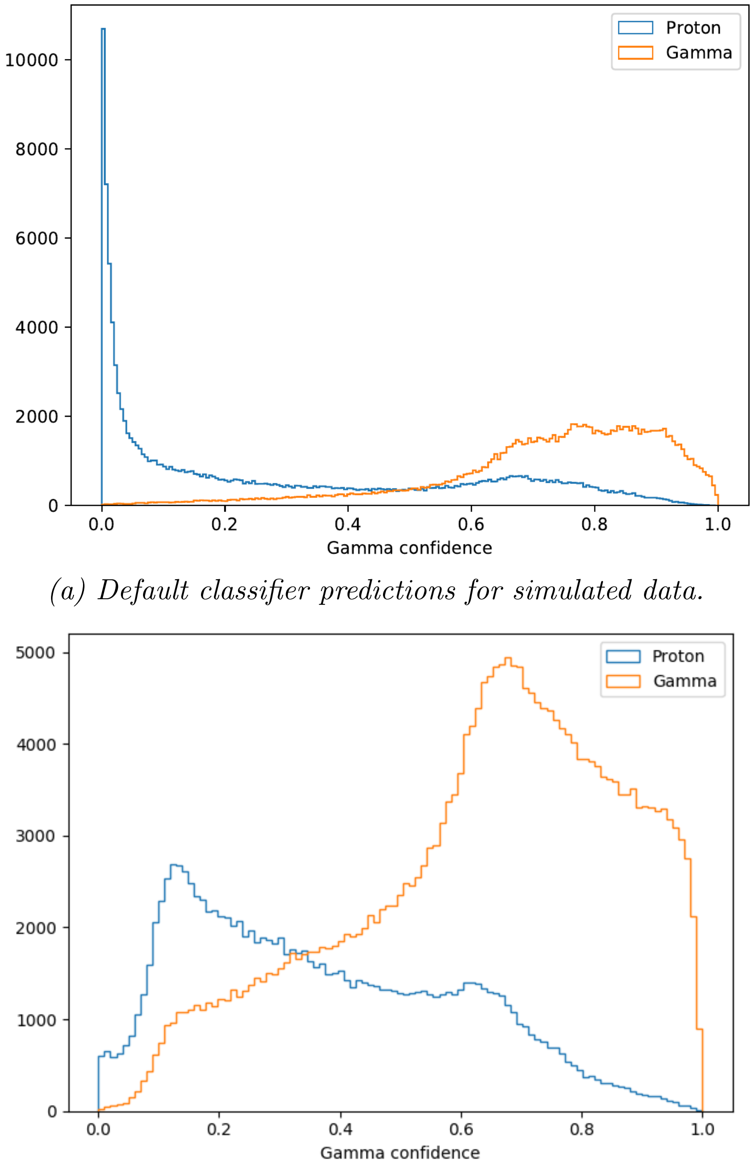
Figure 18: ROC curve for (a) the random forest classifier, (b) the best convolutional network trained on simulated events both predicting on the test simulated events.



(a)

(b)

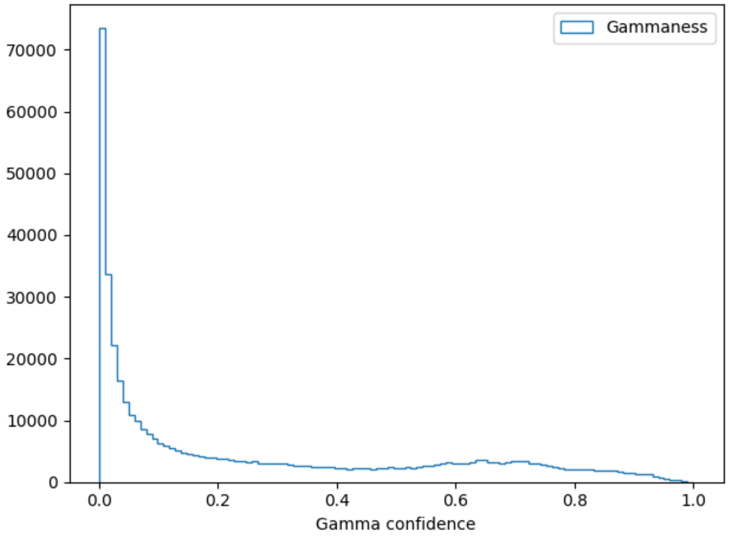
Figure 19: ROC and AUC for Crab Nebula trained neural network on predicting (a) simulated events (b) Markarian 501 events, where events within $\theta^2 < 0.025$ are assumed gamma-ray events, and the rest assumed hadron.



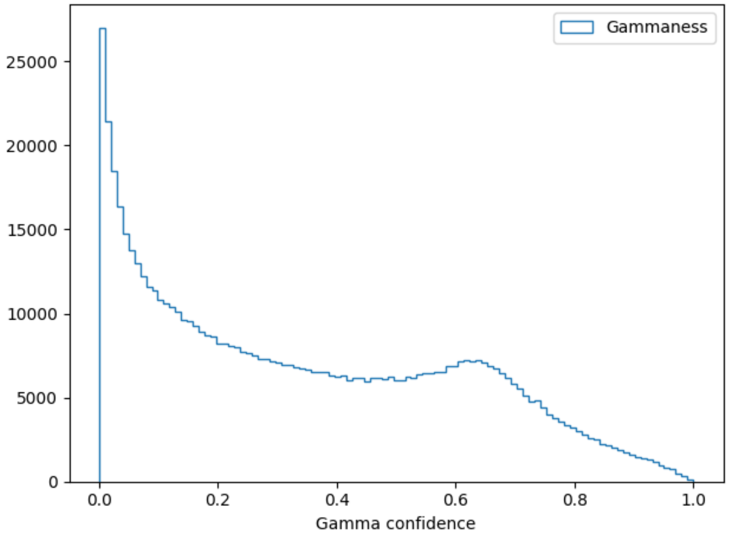
(a) Default classifier predictions for simulated data.

(b) Convolutional neural network predictions for simulated data.

Figure 20: Event classification confidences for simulated events. A value of 0 means the event is classified as a proton event, while a value of 1 means it is classified as a gamma-ray event.

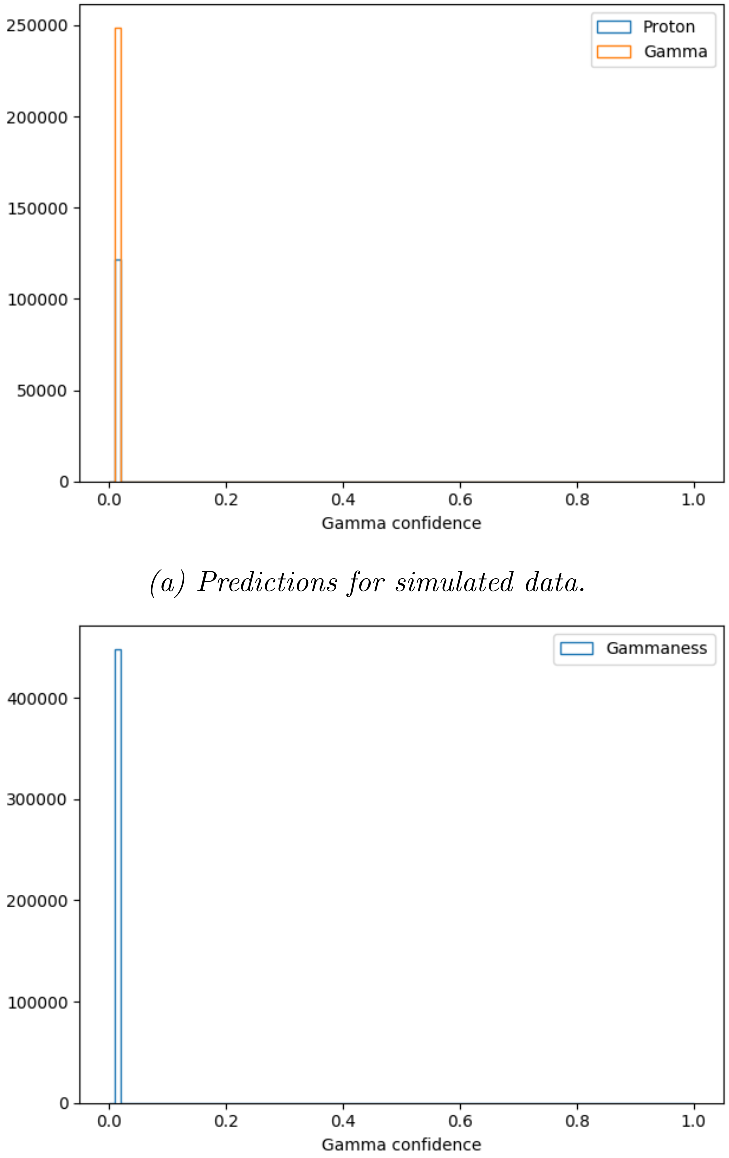


(a) Default classifier predictions for Markarian 501 data.



(b) Convolutional neural network predictions for Markarian 501 data.

Figure 21: The same classifiers as in Fig. 20, applied to Markarian 501 observations.

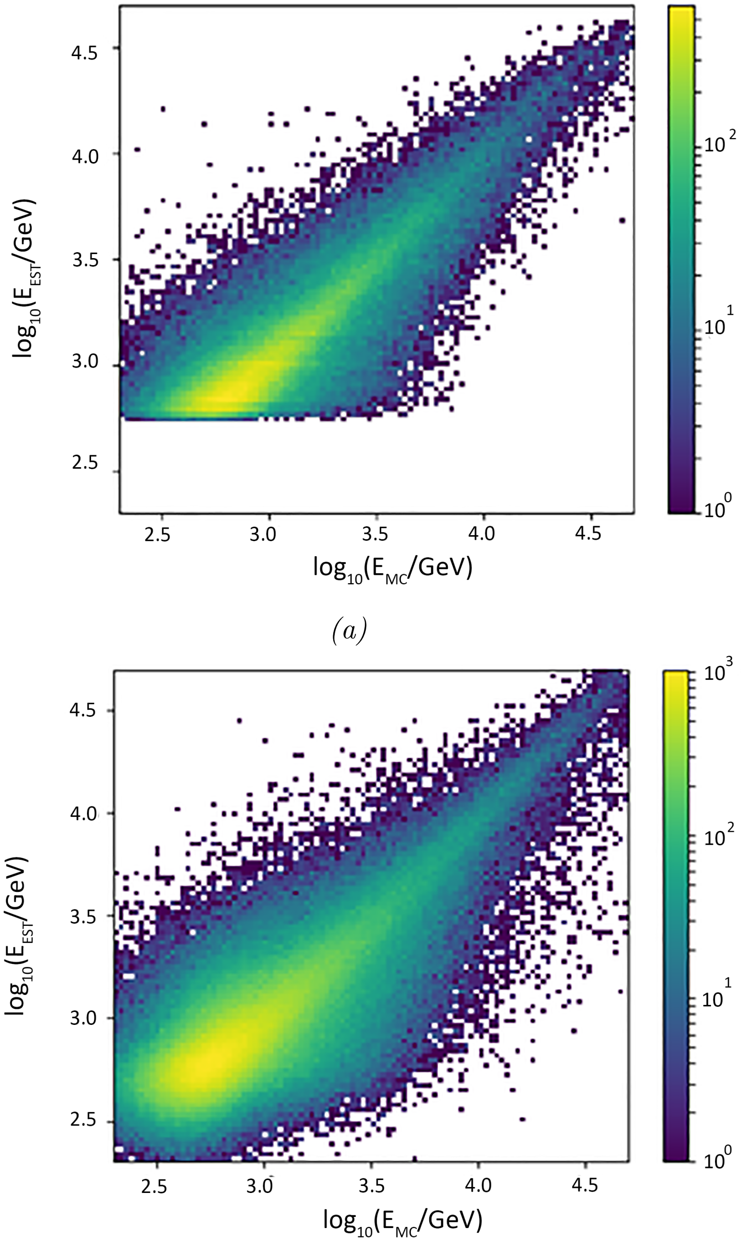


(a) Predictions for simulated data.

(b) Predictions for Markarian 501 data.

Figure 22: Neural network classifier trained on Crab Nebula observation data.

8.2 Energy Estimation



(a)

(b)

Figure 23: Energy estimation results for (a) random forest (b) convolutional neural network.

In this case the best neural network approached the performance of the random forest. The R^2 value of the current regressor's estimation is 0.80, while that of the neural network is 0.77, as shown in Fig. 23. Both have a wide spread in the predicted energy values, but in the high energy ($>$ 1 TeV) regime, the neural network has a wider spread of values versus that of the random forest classifier. This could be because of the lack of training data for high energy particles. The gap at the bottom of the random forest's results is from the random forest not predicting any values below $\approx$ 400 GeV. Out of the 250000 training events, only 5000 of those events were above 1 TeV, while the vast majority were between 0.1 TeV and 1 TeV. Both the random forest and the neural network have a large spread for lower energies. This seems to be because the low-energy ($<$ 0.1 TeV) events that are recorded by FACT deposit a statistically unusually large amount of light onto the detector [22], creating images that have similar photon counts to particles in the 0.1 TeV to 1 TeV range.

8.3 Source Estimation

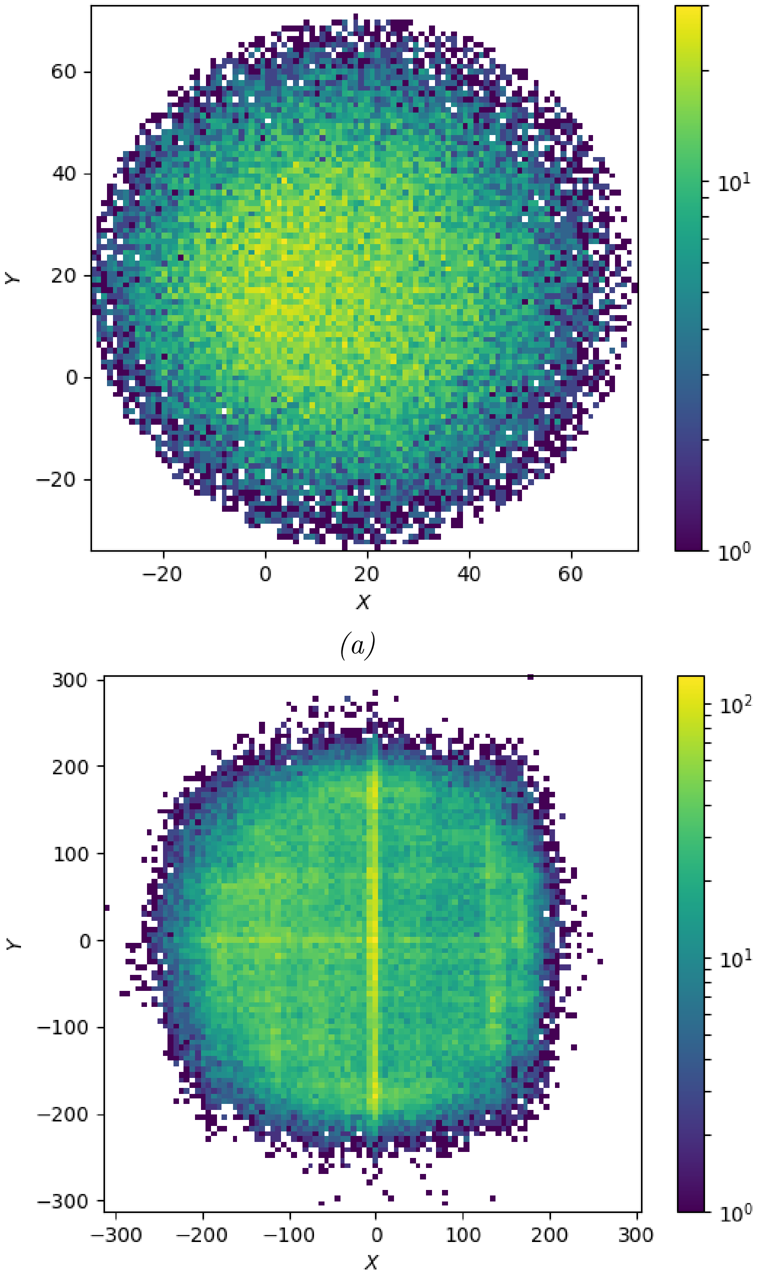
The source estimation results were mixed overall. While estimating the source location in the camera directly was unsuccessful, estimating the disp and the sign for the disp method performed better than the random forest.

8.3.1 Estimating source location directly

In all cases, the direct source estimation on FACT data was vastly different than results found with other telescopes [1, 2]. The neural network would only predict the mean value of the training data for all inputs, roughly the center of the detector. The only architectures that performed better were networks where the x and y values were estimated separately and their predictions combined after. In that case, the neural network would still not predict correct x and y values. The results from the best performing network are shown in Fig. 24. The root cause of why this occurs is not clear, although it seems to be that the images do not contain enough information to accurately reconstruct the source position. Shilon et. al's results depended on there being at least two different views of a given shower for their source estimator to predict correctly [1].

The comparison of the true and estimated source positions in Fig. 24 shows the bias of the neural network to predict sources where either the x or y coordinate is 0. While the predicted source coordinates does somewhat resemble the true source positions, by looking at the individual estimated x and y coordinates versus the true values reveals that the network does not find any patterns in the data that it can use to accurately estimate the position of the source. The R^2 values for both the y-coordinate and x-coordinate estimation are 0.0, and increasing the number of training epochs did not improve this.

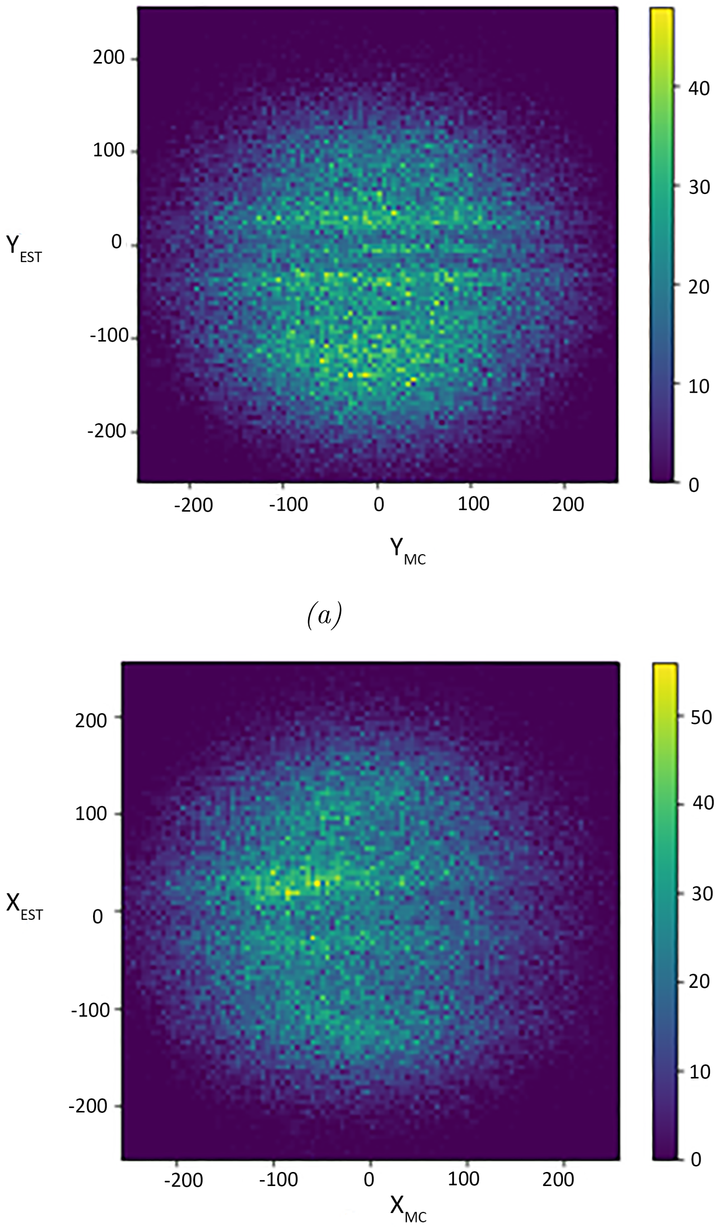
For the networks that used the mean squared error for x and y together as a single loss function, the results were worse. All of these type of networks would train quickly, reaching the minimum loss within the first few epochs and only predicting the mean value of the training data on all inputs after that point. Lowering the learning rate did not improve the predictions, and neither did varying the number of layers or level of dropout.



(a)

(b)

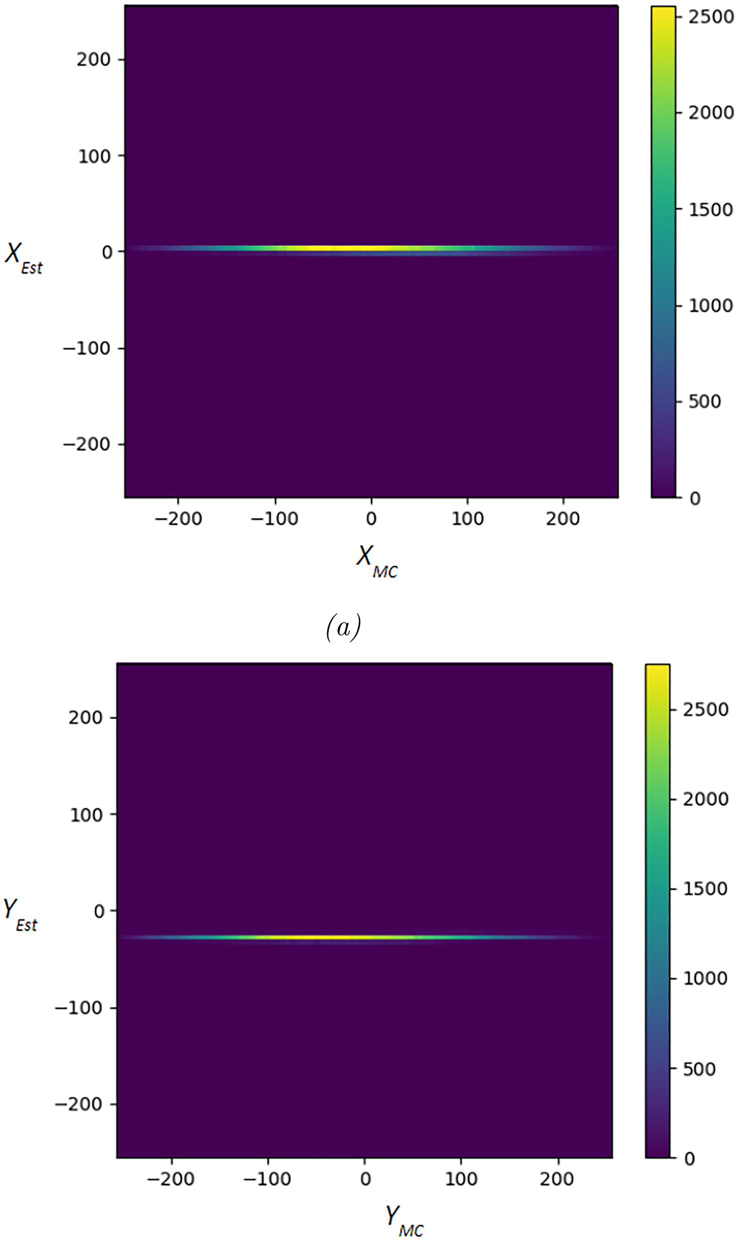
Figure 24: Source location (a) truth and (b) prediction for simulated diffuse gamma-rays.



(a)

(b)

Figure 25: Source location estimation results for (a) x-coordinate and (b) y-coordinate for convolutional neural network with separate outputs for each coordinate.



(a)

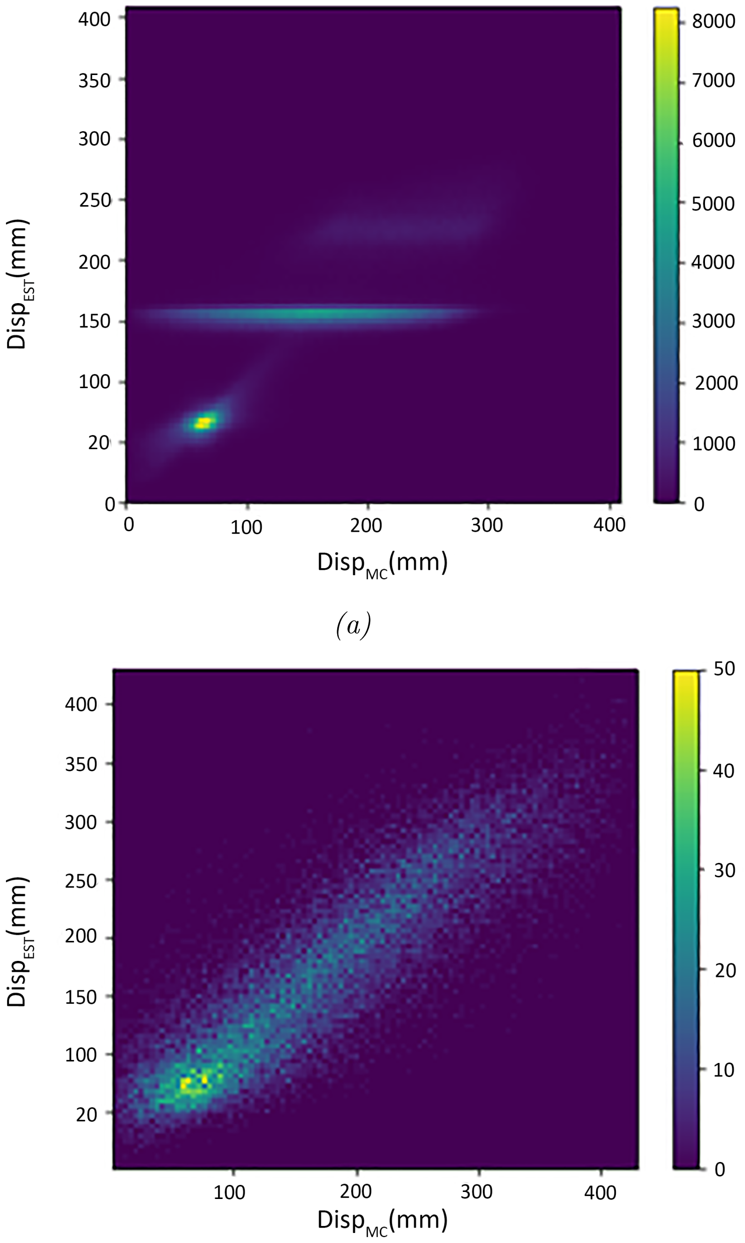
(b)

Figure 26: Source location estimation results for (a) x-coordinate and (b) y-coordinate for convolutional neural network with single output.

8.3.2 Estimating disp and sign

Estimating the disp value and sign were more successful than for the direct source estimation. On the simulated diffuse gamma-ray events, the results are shown in Fig. 27. For the disp regression, the convolutional neural network outperformed the random forest significantly. The random forest result had an R^2 value of 0.23, while the neural network achieved an R^2 value of 0.60. In addition, the neural network performed well throughout the whole range of disp values, while the random forest had two values, ≈ 60 and ≈ 150 where it tended to cluster its estimations. For the sign classification, the neural network also outperformed the random forest, with an accuracy of 66% versus an accuracy of 60%. This result is the most promising, suggesting that neural networks could be used to improve the disp estimation, and therefore the accuracy of the source estimation.

That the disp method neural network giving reasonable results but the direct source networks not, suggests that there might not be enough information available with single telescope data to reconstruct the source position directly. A neural network designed to estimate the disp value works much better for single-telescope data.



(a)

(b)

Figure 27: Disp estimation for (a) random forest (b) convolutional neural network.

9 Conclusion

While neural networks have been proven to perform well at classification and regression tasks, the difficulty in applying convolutional networks to the FACT analysis and source detection reveals some important insights. First, that the limitations of single-telescope data versus that of multi-telescope data is readily apparent. Using the same network architectures and similar data preprocessing, both the VERITAS and HESS collaborations have been able to perform gamma/hadron separation, energy regression, and source detection with much better performance than in this project [1–3]. Second, when the features of random forests are chosen carefully, random forests can outperform neural networks at difficult tasks. In addition, the random forest seems to be better at classifying real events that are physically gamma-ray events based upon comparing their Hillas parameters. Training on observation data went much worse than expected. While the assumption that all events within a given θ^2 cut were gamma-ray events was known to not be entirely correct, the neural networks could not distinguish between any events after training on Crab Nebula data. Hopefully there is some preselection other than θ^2 cut that can create a dataset that is mostly real gamma-ray events.

9.1 Future Directions

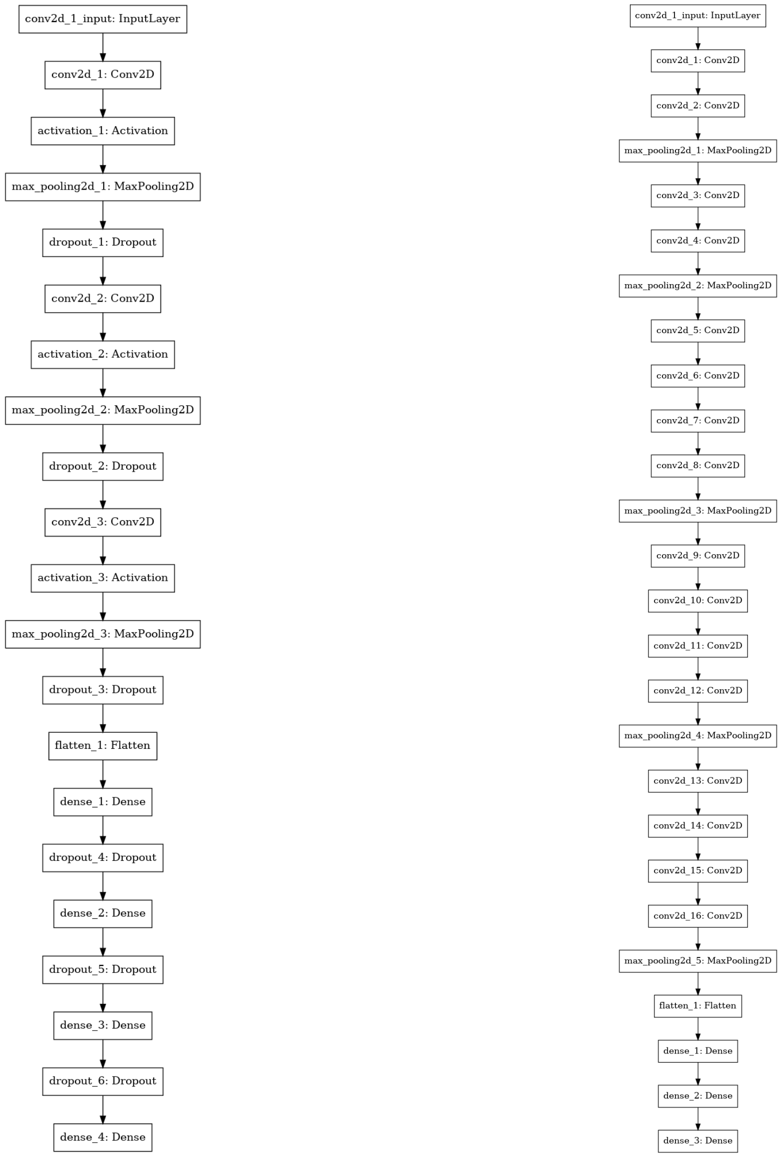
One future step would be to include more information along with the images that are fed to the neural networks, such as the photon arrival time information. By including the timing information, a convolutional neural network with an additional recurrent layer could be used to take advantage of that information, much like the gamma/hadron separation neural network used with HESS data [1, 51].

While training on observation data was not successful, having a larger set of training data might allow for better discrimination between event types. There was a total of 40000 events within 0.025 θ^2 of Markarian 501, but there was only 3000 events within 0.01 and 275 events within 0.001 θ^2. The smaller the θ^2 cut around a source is used, the more likely that a given observed event is a gamma-ray. A larger set of observation data could relieve the issue of not enough training data for the network to properly learn the features needed to discriminate between the two classes of events.

We can also refine the energy estimation by generating larger simulation datasets, especially for high energy events, as well as testing out more neural network architectures to see if they can improve the estimation. Future work to improve the source detection could rely on finding the center-of-gravity of the image through the use of networks that work well for segmentation, such as the UNET architecture used in biological imaging [52]. Such a network could segment the main clump of the image and find the ”center-of-gravity” of that clump for use in the disp regressor without needing to preprocess the data first with FACT-Tools.

One of the largest next generation gamma-ray telescope projects is the Cherenkov Telescope Array, or CTA, that is comprised of multiple IACTs of various sizes built into an array for better sensitivity. With an expected online date of 2024, CTA is still years away from operation, but lessons learned from this project FACT's event classification and source detection could improve CTA's performance. This could allow for fainter sources to be detected. thereby providing a more refined view of our Universe.

10 Appendix



(a) Best performing convolutional neural network for gamma/hadron separation.

(b) VGG-style network

Figure 28: Separation architectures used.

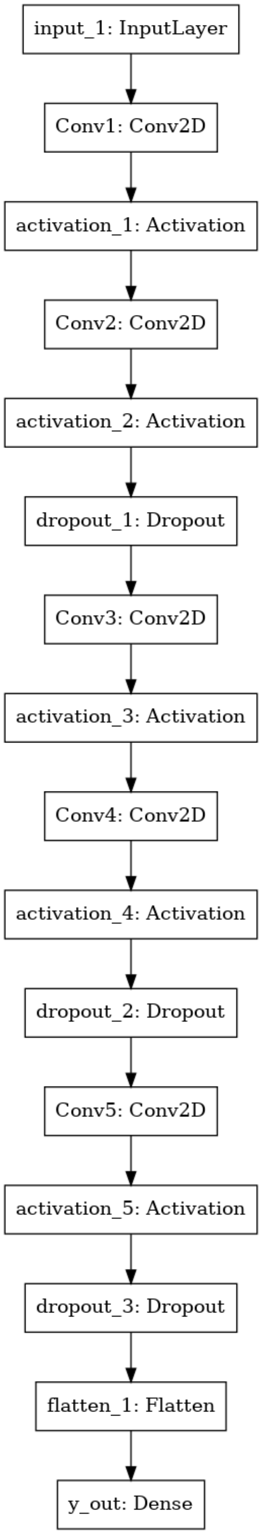
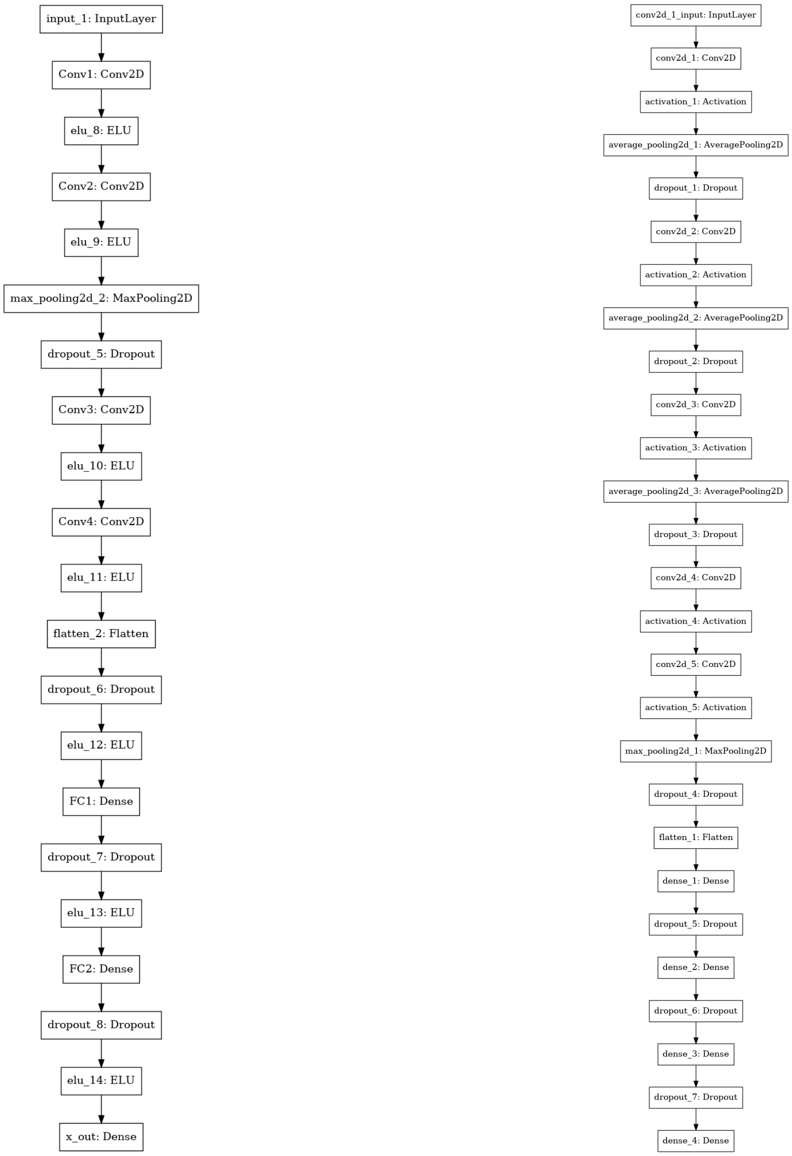


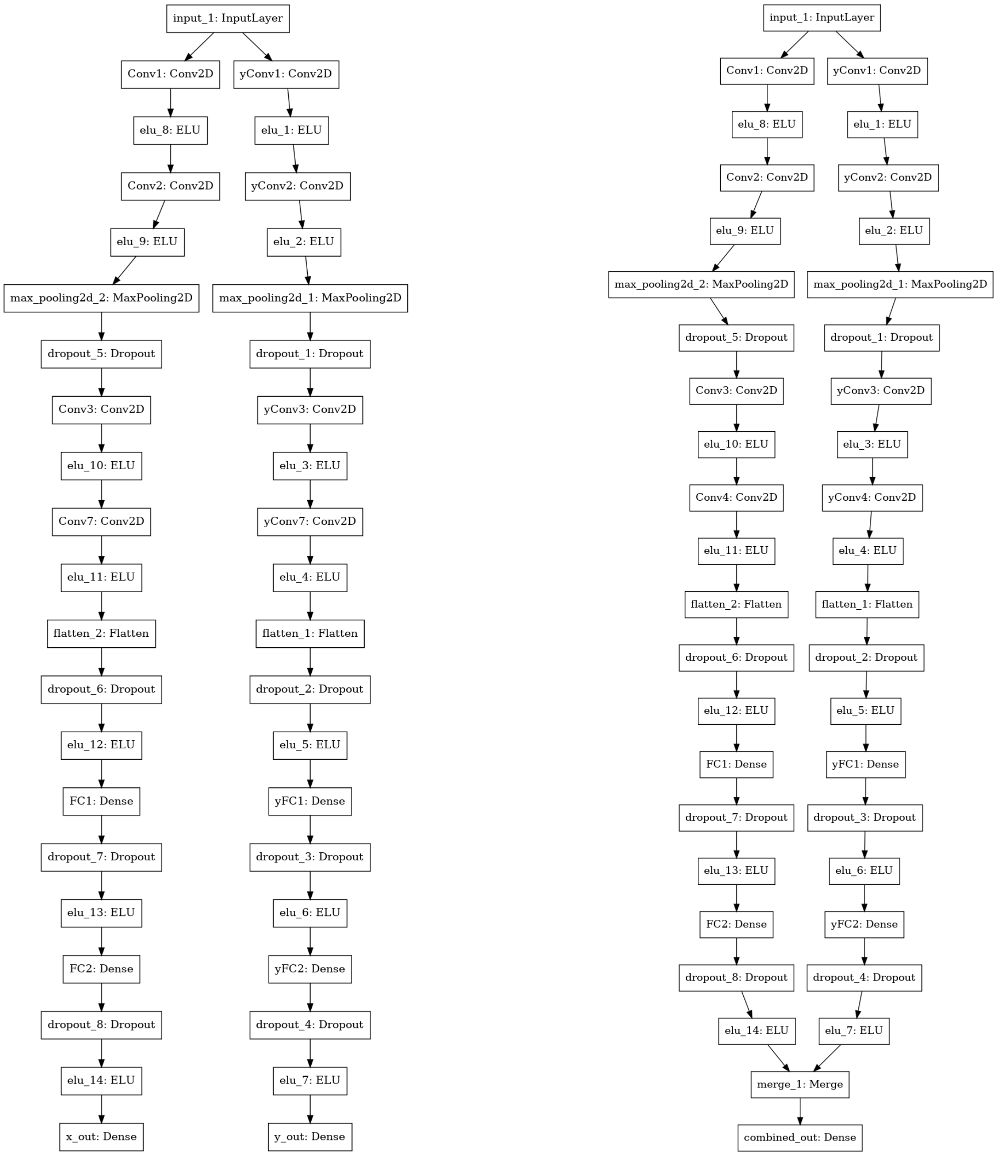
Figure 29: Best performing convolutional neural network for energy estimation.



(a) Best performing convolutional neural network for Disp regression.

(b) Best performing network for sign classification.

Figure 30: Disp and sign architecture used.



(a) Best separate loss function model for x,y source estimation.

(b) Best combined loss function model for x,y source estimation.

Figure 31: Direct source estimation architectures used.

Bibliography

[1] I. Shilon, M. Kraus, M. Büchele, K. Egberts, T. Fischer, T. L. Holch, T. Lohse, U. Schwanke, C. Steppa, and S. Funk, "Application of deep learning methods to analysis of imaging atmospheric cherenkov telescopes data", arXiv preprint arXiv:1803.10698 (2018).

[2] T. L. Holch, I. Shilon, M. Büchele, T. Fischer, S. Funk, N. Groeger, D. Jankowsky, T. Lohse, U. Schwanke, and P. Wagner, "Probing convolutional neural networks for event reconstruction in $\{\gamma\}-rayastronomywithcherenkovtelescopes$", arXiv preprint arXiv:1711.06298 (2017).

[3] Q. Feng, T. T. Lin, V. Collaboration, et al., "The analysis of veritas muon images using convolutional neural networks", Proceedings of the International Astronomical Union **12**, 173–179 (2016).

[4] H. Anderhub, M. Backes, A. Biland, A. Boller, I. Braun, T. Bretz, S. Commichau, V. Commichau, D. Dorner, A. Gendotti, O. Grimm, H. von Gunten, D. Hildebrand, U. Horisberger, J.-H. Köhne, T. Krähenbühl, D. Kranich, E. Lorenz, W. Lustermann, K. Mannheim, D. Neise, F. Pauss, D. Renker, W. Rhode, M. Rissi, M. Ribordy, U. Röser, L. Stark, J.-P. Stucki, O. Tibolla, G. Viertel, P. Vogler, and Q. Weitzel, "A g-apd based camera for imaging atmospheric cherenkov telescopes", Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **628**, VCI 2010, 107–110 (2011).

[5] J. Holder, “Atmospheric cherenkov gamma-ray telescopes”, arXiv preprint arXiv:1510.05675 (2015).

[6] J. M. Behnken, *Cnn classification with fact images*, https://github.com/JMBehnken/CNN_Classification_with_FACT_images, 2017.

[7] Wikipedia, *Bhabha scattering*, https://en.wikipedia.org/wiki/Bhabha_scattering.

[8] Wikipedia, *Møller scattering*, https://en.wikipedia.org/wiki/M%5C%C3%5C%B8ller_scattering.

[9] D. V. Schroeder, *Feynman diagrams and electron-positron annihilation*, (2002) https://physics.weber.edu/schroeder/feynman/ (visited on 10/30/2002).

[10] N. Aeronautic and S. Adminstration, *Gamma ray detectors*, (2013) https://imagine.gsfc.nasa.gov/science/toolbox/gamma_detectors2.html (visited on 10/2013).

[11] A. Sandage, “Encyclopedia of astronomy and astrophysics”, (2001).

[12] L. Anchordoqui, M. T. Dova, A. Mariazzi, T. McCauley, T. Paul, S. Reucroft, and J. Swain, “High energy physics in the atmosphere: phenomenology of cosmic ray air showers”, Annals of Physics **314**, 145–207 (2004).

[13] R. Engel, “Theory and phenomenology of extensive air showers”, Forschungszentrum Karlsruhe, Germany-available at http://moriond. in2p3. fr J **5** (2005).

[14] R. Egerton, *Pair production and annihilation*, http://web.pdx.edu/~egertonr/ph311-12/pair-p&a.htm.

[15] C. Pfendner, *The lpm effect: a summary*, (2013) https://u.osu.edu/connolly/files/2013/12/LPM-Effect-10f7n7w.pdf.

[16] K. Bernlöhr, *Atmospheric cherenkov light*, (2018) https://www.mpi-hd.mpg.de/hfm/CosmicRay/ChLight/Cherenkov.html (visited on 04/23/2018).

[17] H. Collaboration, *Detecting cosmic rays*, (2018) https://www.hawc-observatory.org/science/detection.php#sec:cherenkov (visited on 04/23/2018).

[18] A. Zilles, "Modeling of radio emission from particle/air showers", in *Emission of radio waves in particle showers* (Springer, 2017), pp. 15–30.

[19] Wikipedia, *Muon*, https://en.wikipedia.org/wiki/Muon.

[20] A. Cillis and S. Sciutto, "Extended air showers and muon interactions", Physical Review D **64**, 013010 (2001).

[21] K. Kainz, "Radiation oncology physics: a handbook for teachers and students", Medical Physics **33**, 1920–1920 (2006).

[22] T. F. Temme, "On the hunt for photons: analysis of crab nebula data obtained by the first g-apd cherenkov telescope", PhD thesis (Technische Universität Dortmund, 2016).

[23] *Cosmic rays*, https://en.wikipedia.org/wiki/Cosmic_ray.

[24] K. Bernlöhr, *Cosmic-ray air showers*, (2018) `https://www.mpi-hd.mpg.de/hfm/CosmicRay/Showers.html` (visited on 04/23/2018).

[25] J. Matthews, “A heitler model of extensive air showers”, Astroparticle Physics **22**, 387–397 (2005).

[26] J. Holder, “Atmospheric Cherenkov Gamma-ray Telescopes”, ArXiv e-prints (2015).

[27] ISDC, *Very high energy gamma-rays*, (2018) `http://www.isdc.unige.ch/cta/outreach/data` (visited on 04/23/2018).

[28] D. Heck, G. Schatz, J. Knapp, T. Thouw, and J. Capdevielle, *Corsika: a monte carlo code to simulate extensive air showers*, tech. rep. (1998).

[29] F. Collaboration, *Fact tools*, `https://github.com/fact-project/fact-tools`.

[30] *Decision trees in machine learning*, `https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052`.

[31] *The random forest algorithm*, `https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd`.

[32] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition”, arXiv preprint arXiv:1409.1556 (2014).

[33] J. Fox, *Neural networks 101*, `https://github.com/cazala/synaptic/wiki/Neural-Networks-101`.

[34] D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)", CoRR **abs/1511.07289** (2015).

[35] *Cs321*, `http://cs231n.github.io/neural-networks-1/#summary`.

[36] S. Sharma, *Activation functions*, `https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6`.

[37] S. Basak, *Exponential linear units are the new cool*, `http://saikatbasak.in/sigmoid-vs-relu-vs-elu/`.

[38] C. Spark, *Deep learning for complete beginners*, `https://cambridgespark.com/content/tutorials/convolutional-neural-networks-with-keras/index.html`.

[39] ujjwalkarn, *An intuitive explanation of convolutional neural networks*, `https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/`.

[40] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting", Journal of Machine Learning Research **15**, 1929–1958 (2014).

[41] *Machine learning cheatsheet*, `http://ml-cheatsheet.readthedocs.io/en/latest/loss_functions.html`.

[42] T. Lukas Holch, I. Shilon, M. Büchele, T. Fischer, S. Funk, N. Groeger, D. Jankowsky, T. Lohse, U. Schwanke, and P. Wagner, "Probing Convolutional Neural Networks for Event Reconstruction in γ-Ray Astronomy with Cherenkov Telescopes", ArXiv e-prints (2017).

[43] H. Anderhub, M. Backes, A. Biland, V. Boccone, I. Braun, T. Bretz, J. Buß, F. Cadoux, V. Commichau, L. Djambazov, et al., "Design and operation of fact–the first g-apd cherenkov telescope", Journal of Instrumentation **8**, P06008 (2013).

[44] A. Biland, T. Bretz, J. Buß, V. Commichau, L. Djambazov, D. Dorner, S. Einecke, D. Eisenacher, J. Freiwald, O. Grimm, et al., "Calibration and performance of the photon sensor response of fact—the first g-apd cherenkov telescope", Journal of Instrumentation **9**, P10012 (2014).

[45] F. Chollet et al., *Keras*, `https://github.com/fchollet/keras`, 2015.

[46] D. Nieto, A. Brill, B. Kim, T. B. Humensky, and f. t. Cherenkov Telescope Array, "Exploring deep learning as an event classification method for the Cherenkov Telescope Array", ArXiv e-prints (2017).

[47] H. Krawczynski, D. A. Carter-Lewis, C. Duke, J. Holder, G. Maier, S. Le Bohec, and G. Sembroski, "Gamma hadron separation methods for the VERITAS array of four imaging atmospheric Cherenkov telescopes", Astroparticle Physics **25**, 380–390 (2006).

[48] P. Boinee, F. Barbarino, A. De Angelis, A. Saggion, and M. Zacchello, "Neural networks for gamma-hadron separation in magic", Frontiers of Fundamental Physics, 297–302 (2006).

[49] T. Murach, M. Gajdus, and R. D. Parsons, "A neural network-based monoscopic reconstruction algorithm for hess ii", arXiv preprint arXiv:1509.00794 (2015).

[50] *What does auc stand for and what is it*, `https://stats.stackexchange.com/questions/132777/what-does-auc-stand-for-and-what-is-it`.

[51] H. Prokoph, "Investigations on gamma-hadron separation for imaging cherenkov telescopes exploiting the time development of particle cascades", PhD thesis ().

[52] O. Ronneberger, P. Fischer, and T. Brox, "U-net: convolutional networks for biomedical image segmentation", in International conference on medical image computing and computer-assisted intervention (Springer, 2015), pp. 234–241.