**Research Plan: Improving Particle Discrimination in WIMP Dark Matter Detection Experiments Using Neural Networks**

**Rationale**

Detection of dark matter is one of the foremost endeavors in modern particle physics. Due to unexpectedly large gravitational forces affecting the rotation of galaxies and the deflection of photons, it is hypothesized that there are a large number of “dark” particles that rarely interact via forces other than gravity. One of the most-researched candidates for such a particle is the Weakly Interacting Massive Particle, or WIMP.

The identification of background radiation events is a major hurdle to overcome in all experiments for WIMP detection. The current practice of manually developing a discriminator function to eliminate background events is challenging and time-consuming when available calibration data is frequently impure and present in limited quantities. Machine learning, particularly in the form of neural networks, has the potential to be a powerful solution, because it permits automation of the process.

However, there are several problems that must be overcome. For instance, impure data can be detrimental to the training of many types of models, and data often comes in formats that are non-trivial to process, such as audio and 3D geometries. It is not always clear which of these formats can be used to produce the most effective discriminator, nor is it obvious what kind of neural network will learn the most efficiently and accurately on a given data format.

If solutions to these issues can be found, it stands to reason that the pace of many dark matter research projects could be considerably improved. There could potentially be a great reduction in the engineering time that is spent on manually deriving a discriminator function every time some aspect of the experiment changes.

**Background on Experiments**

**PICO-60**

The PICO-60 experiment is a bubble chamber containing superheated C3F8. It is engineered to be almost entirely insensitive to gamma rays and many other forms of low-energy radiation. This means that the majority of unwanted background events are caused by alpha particles. Alpha particles are emitted by nuclear decays of radon and polonium atoms inside the detector, meaning that they can not be easily removed by ignoring events near the walls of the detector. For development of discriminators that use other means to detect background events, there is an available set of 99%-pure alpha background events collected in 2016 and 2017.

Hypothesized signal events are nuclear recoils, and WIMPs in the mass and energy ranges PICO-60 searches within are expected to produce nuclear recoils indistinguishable from those produced by neutrons. Thus, to train a discriminator to recognize nuclear recoil events, approximately 90%-pure calibration data produced using neutrons from americium and californium sources is available.

Sensory data for each recorded event comes from two piezoelectric microphones (piezos) mounted to the outside of the silica vessel, and also from three externally mounted cameras viewing the transparent detector at different angles.

**DEAP-3600**

The DEAP-3600 experiment makes use of 255 photomultiplier tubes (PMTs) mounted uniformly over the surface of a spherical vessel filled with liquid argon. The principle of the detector is that when an argon atom is struck by another particle (such as an alpha particle or a WIMP), some of its electrons are excited, and when they fall back to ground state, they emit photons that are captured by some of the PMTs. Based on the counts and timings of photons that reach each of the PMTs, it is possible to approximate the energy and location of any event that occurs in the body of the detector. This allows alpha, beta and gamma background radiation to be removed.

Remaining background radiation consists of alpha radiation events that do not occur in the spherical body of the detector, but rather in the neck, above the hole through which the detector was filled. Because many photons are occluded by the corners and sides of the neck, the position of the event is predicted to be in the lower middle of the detector, rather than the top. Also, because the neck is filled with gas, the alpha event appears to be much lower-energy than it actually is; in fact, it appears to be in the same energy range as WIMP candidates.

It is impractical to create significant amounts of useful calibration data in the DEAP-3600 detector. Thus, a simulation is used instead. It is possible to create thousands of WIMP and alpha events for training in this environment. However, how well the simulation replicates real-world events remains an open question. There is a set of 30 real-world events, containing both expected nuclear recoils and expected neck alpha events, that can be used for testing any discriminator.

**Engineering Goal**

In summary, the goal of this project is to develop machine learning systems that can effectively and efficiently learn to discriminate between signal and background events in the context of the PICO-60 and DEAP-3600 WIMP detection experiments. This will be achieved by experimenting with a variety of different input formats, network architectures, and novel systems for improving data efficiency.

Furthermore, based on the results that come out of this investigation, a secondary goal of this project is to provide some insight on the properties of the input data that are taken advantage of by the neural network discriminators.

**Procedures**

**PICO-60**

For the PICO-60 experiment, there are a number of different input formats that can potentially be used as input data for a neural network. The format used in past research on the experiment is a Fourier transform integrated into eight frequency bands. This was used as the input to a conventional discriminator known as the Acoustic Parameter (AP), as well as to a multi-layer perceptron that was applied experimentally as part the original PICO-60 study.

There are three other data formats that will be investigated in the context of PICO-60:

1. A full-resolution analogue to the banded Fourier transform, using all values in the high-resolution curve directly without any integration
2. A full-resolution audio waveform, which includes time domain information (such as how the frequency distribution changes over time) which may be useful
3. Images captured by cameras in the detector, which have been used only for reconstructing the positions of events but are not proven to be inapplicable for background event identification

Using whichever of these data formats is observed to produce the most effective discriminator, the intent is to experiment with a wide variety of neural network hyperparameters, including the following (depending on the type of neural network used):

* Dropout regularization
* L2 regularization
* Number of layers (convolutional and dense)
* Number of convolutional filters
* Size of convolutional kernel

In addition to a variety of hyperparameters applied to dense and convolutional neural networks (CNNs) trained with conventional supervised learning, novel semi-supervised learning algorithms will be applied to improve performance. The essential concept of semi-supervised learning is that a limited amount of labeled (classified) training events is used alongside unlabeled events, which help the network more accurately learn a decision boundary between the two classes.

More specifically, the problem in the PICO-60 experiment is that the training data is impure, so it is not desirable to fit perfectly (which would amount to finding some element of the audio, such as noise, that differs based on when the data was collected). Rather, one would hope that a machine learning system could ignore the segments of the data that are impure and rather learn to separate particle types.

I hypothesize that semi-supervised learning will be able to improve the effectiveness of training by taking advantage of the neural network’s *most confident* predictions (those where the output is extremely close to either 0 or 1) on unlabeled data. I expect that these predictions should correlate with the actual particle types, and that incorrect predictions caused strictly by overfitting will be less confident. I have invented two semi-supervised learning algorithms, that I intend to test, that should take advantage of this predicted property:

* Iterative cluster nucleation: This is an algorithm that, given a set of labeled data *X* and a set of unlabeled data *U*, iteratively determines labels for examples in *U* and adds them to *X*. It applies the following procedure:
  + Train a neural network for a small number of epochs on *X* (perhaps 30).
  + Run inference on the entirety of *U*; call the vector of predictions *P*.
  + Find all values in *P* that are very close to either 0 or 1 (suppose less than 0.05 or greater than 0.95), representing high confidence.
  + For each of the corresponding examples in *U*, assign it a label according to which of the two predictions it is given, and add it to *X*.
  + Repeat, training on the newly updated labeled set *X*.
* Gravitational differentiation: This is a related algorithm that can be thought of as a more analog alternative to iterative cluster nucleation. Rather than making a binary decision on whether or not to add a given example to the labeled training set, it makes use of a distortion function that calculates final-layer derivatives for each example in *U* such that an example with a prediction of 0.5 (neutral) has no effect on training, and unlabeled examples with predictions closer to 0 or 1 have progressively more effect on the training process.  
  This can be formalized in the following piecewise exponential function *GravDiff*, where *p* is the neural network’s prediction on the training example in question, *ψ* is a parameter that defines the degree of distortion, and *g* is a parameter that defines the overall influence of unlabeled examples on the training process:

**DEAP-3600**

For the DEAP-3600 experiment, large amounts of simulated training data are available. This means data efficiency systems such as semi-supervised learning are less applicable than PICO-60. However, the format of the data presents a very different set of challenges. There are two main open questions:

1. Each of the photomultiplier tubes receives multiple photons at different times. What is the best way to encode this information for a neural network? I will attempt to use both the total number of photons received by each PMT, and the time of arrival of the first photon at each PMT.
2. The 255 photomultiplier tubes are arranged in a mostly-hexagonal lattice on the surface of the spherical vessel, minus a large hole in the top. What is the best neural network architecture to process this spatial arrangement of data points? I have considered the following options:
   * A dense neural network is an obvious and simple choice, but it will likely require more parameters than something optimized for this spatial arrangement (for the same reasons that dense neural networks are suboptimal for image processing compared to convolutional neural networks). It should thus have a greater tendency to overfit and train more slowly.
   * A common CNN cannot be applied without deforming the spherical topology, because they are designed to operate on flat rectangular images, and (as we know from world maps) a sphere cannot be projected onto a rectangle without distortion. However, the distortions introduced by world maps do not prevent us from reading the map, and similarly, applying a Mercator-like map projection to the sphere may allow a conventional CNN to work effectively.
   * I have invented a novel neural network architecture called a topological CNN, which may be an effective solution to this problem. It is much like a conventional CNN, except that rather than a kernel being convolved over a square grid, a kernel is convolved over an arbitrary geometric lattice of polygonal nodes (hexagonal, in this case).  
     Each node is assigned six neighbors in a clockwise order, and a kernel consists of all node within a certain distance of its center node. (Distance is defined by the minimum number of connections that must be taken to get from one node to the next.) In a topological convolutional layer, such a kernel is formed at every possible center node, and the output of the layer is an identical topology (excepting nodes at the edges where kernels can not be formed without hitting the edge of the topology).  
     To ensure that the orientation of every kernel is the same (that is, that they are not randomly rotated in different positions on the topology), the clockwise order of the connections of each node must be rotated such that the top node (the node with the highest vertical position) comes first. This way, orientations of kernels will be symmetrical along the vertical axis (on opposite sides of a sphere, for instance).