

# Towards Light Charge Association in Liquid Argon Time Projection Chambers

*John Newsom*



Electrical Engineering and Computer Sciences  
University of California, Berkeley

Technical Report No. UCB/EECS-2021-52

<http://www2.eecs.berkeley.edu/Pubs/TechRpts/2021/EECS-2021-52.html>

May 12, 2021

Copyright © 2021, by the author(s).  
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

#### Acknowledgement

To everyone who helped me along the way, thank you.

**Towards Light Charge Association in Liquid Argon Time Projection  
Chambers**

by John Newsom

**Research Project**

Submitted to the Department of Electrical Engineering and Computer Sciences,  
University of California at Berkeley, in partial satisfaction of the requirements for the  
degree of **Master of Science, Plan II**.

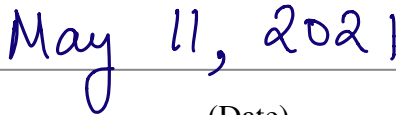
Approval for the Report and Comprehensive Examination:

**Committee:**



---

Professor Gireeja Ranade  
Research Advisor



---

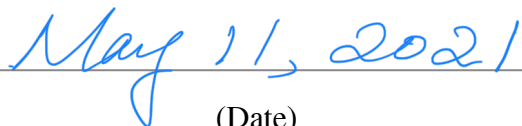
(Date)

\*\*\*\*\*



---

Professor Kam-Biu Luk  
Second Reader



---

(Date)

---

# Towards Light Charge Association in Liquid Argon Time Projection Chambers

---

John Newsom

## Abstract

The Deep Underground Neutrino Experiment aims to better understand the physics governing neutrinos using a number of different types of detectors and measurement devices. One such detector, the liquid argon time projection chamber, records the charge deposited and light emitted by neutrino interactions to measure the topology of these events. However, it is expected that the interactions in these time projection chambers located near the neutrino source will occur with such high density that resolving overlapping events will be difficult. To alleviate this problem, we propose a new model architecture, called Y-Net, that utilizes the time-varying scintillation light to improve instance segmentation accuracy on these neutrino interactions. Experiments on simulated neutrino interaction events show that by integrating the timing information carried by scintillation light, we are able to improve performance over existing techniques on the most difficult instances of piled-up interactions.

## 1 Introduction

The Deep Underground Neutrino Experiment (DUNE) is a massive experiment underway at Fermilab in Batavia, Illinois. The DUNE collaboration aims to improve understanding of the physics of neutrinos, near-massless particles that can pass through most normal matter undetected. Specifically, the experiment's massive collaboration has the following objectives:

1. Obtain high statistics measurements of physics parameters responsible for the  $\nu_\mu \rightarrow \nu_e$  and  $\bar{\nu}_\mu \rightarrow \bar{\nu}_e$  oscillation dynamics, which would help explain charge-parity violation and the neutrino mass ordering,
2. Search for several important modes of proton decay to help reach a Grand Unified Theory, and
3. Measure  $\nu_e$  production from intragalactic supernovae if they were to occur [1].

To perform the oscillation experiments, the DUNE collaboration will create a neutrino beam with the Long-Baseline Neutrino Facility (LBNF) at Fermilab as depicted in [Figure 1](#). This neutrino beam will be aimed underground where it will travel over 1300 kilometers to the so-called 'far detector' at the Sanford Underground Research Facility (SURF) in Lead, South Dakota. In order to understand the nature of the neutrino beam before it has travelled from its source at LBNF to SURF, another detector aptly named the 'near detector', placed approximately 500 meters downstream of the beam production target, will include a liquid argon time projection chamber and will measure the absolute flux and energy information of four neutrino species ( $\nu_\mu, \bar{\nu}_\mu, \nu_e, \bar{\nu}_e$ ) to predict the flux at the far detectors as a function of energy.

A particularly important detector used at DUNE is the aforementioned liquid argon time projection chamber (LArTPC), which measures the ionization charge deposited by neutrino interactions to determine event topologies. It is known that for most types of interaction events, these charge depositions are sufficient to calculate extremely high quality instance segmentation maps, allowing

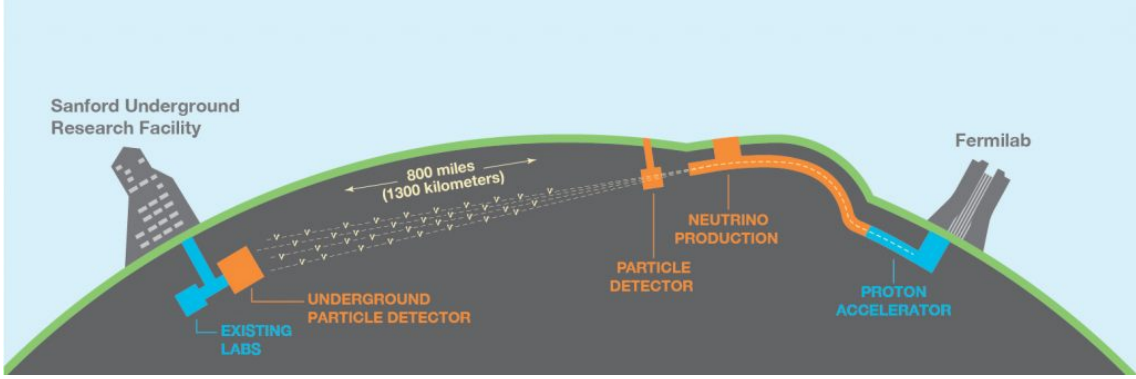


Figure 1: The DUNE Experiment. A beam produced at Fermilab in Illinois will travel over 1300 kilometers underground to the Sanford Underground Research Facility and will be studied to understand physics governing neutrinos [2].

physicists to reconstruct distinct neutrino interactions within the detector volume. However, the relatively short distance between the neutrino beam source and near detector imply that the number of interactions within the DUNE LArTPC detector volume will be as high as sixty, giving rise to a problem referred to as pileup [3]. When interactions occur in increasingly high densities, resolving distinct events based on charge depositions alone becomes more difficult as depositions begin to overlap. For this reason, the DUNE collaboration is investigating the integration of so-called silicon photomultiplier (SiPM) devices into the LArTPC. SiPMs record the time-varying light signal on the level of individual photons emitted through a fast process called scintillation that occurs in parallel with charge deposition. Because the light signal associated with a particular event occurs on the scale of nanoseconds to microseconds, it is believed that a sufficiently sophisticated algorithm could incorporate SiPM measurements to resolve some of these more difficult instances of pileup.

To this end, we present a novel model architecture called Y-Net that combines both charge depositions and scintillation light readouts to create higher quality instance segmentations than previous techniques. We are careful not to assert that the methods presented in this paper are sufficient by themselves to perform the types of end-to-end event reconstruction required for the task of pileup resolution. Instead, we expect that our methods could be integrated into existing reconstruction chains like those presented in Drielsma et al. as a replacement for U-ResNet-like autoencoders [3].

## 2 Preliminaries

### 2.1 About Neutrinos

In the early 20th century, the prevailing view among physicists was that matter was composed entirely of protons and electrons - no other particles were necessary in their physical theories explaining the universe. However, in 1930, Wolfgang Pauli proposed such an extraneous particle, called the neutrino, to explain the conservation of momentum and energy in a process called beta decay. A few years after Pauli's proposal, Enrico Fermi published a quickly accepted complete theory of beta decay dependent on the existence of the neutrino [4].

Neutrinos are now known to exist in at least three eigenstates, called flavors:

- electron neutrino  $\nu_e$
- muon neutrino  $\nu_\mu$
- tau neutrino  $\nu_\tau$

There are also antineutrinos corresponding to each flavor:

- electron antineutrino  $\bar{\nu}_e$
- muon antineutrino  $\bar{\nu}_\mu$

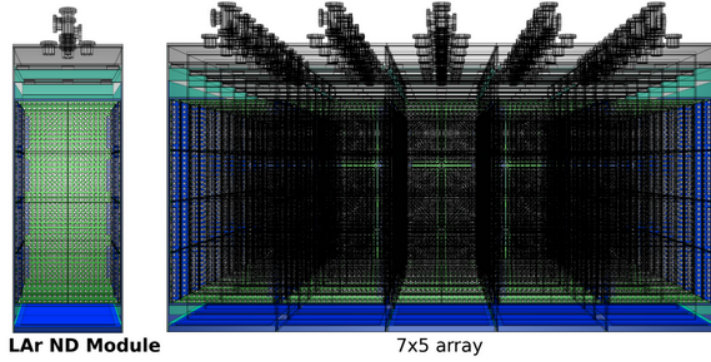


Figure 2: Proposed liquid argon time projection chamber design for the DUNE near detector [7].

- tau antineutrino  $\bar{\nu}_\tau$

An interesting feature of neutrinos is that these flavor eigenstates are not the same as the mass eigenstates. Specifically, each of the flavor eigenstates is understood to be a unique linear combination of the mass eigenstates [5]. Because these mass eigenstates propagate through space at different speeds, the corresponding flavor superpositions can change as neutrinos travel in a process called flavor oscillation [6]. Because the exact masses of the mass eigenstates are not known, the dynamics that dictate these flavor oscillations are not well understood. With flux measurements from the near and far detectors of the DUNE experiment, one will be able to calculate a number of parameters governing flavor oscillation, including the CP-violating phase and mass eigenstate hierarchy [1].

## 2.2 Near Detector Physics

The main objective of the near detector is to provide characterization of the flux, energy spectrum, and flavor composition of the neutrino beam near the source. This data will help control for systematic uncertainties in precision and provide reference values to compare far detector measurements against. Specifically, the near detector will measure the absolute flux of the muon and electron neutrinos and their corresponding antineutrinos as a function of beam energy. This data will be used to calculate

$$\mathcal{A} = \frac{P(\nu_\mu \rightarrow \nu_e) - P(\bar{\nu}_\mu \rightarrow \bar{\nu}_e)}{P(\nu_\mu \rightarrow \nu_e) + P(\bar{\nu}_\mu \rightarrow \bar{\nu}_e)}, \quad (1)$$

the asymmetry in oscillation probabilities between muon neutrinos,  $P(\nu_\mu \rightarrow \nu_e)$ , and antineutrinos,  $P(\bar{\nu}_\mu \rightarrow \bar{\nu}_e)$ , at the far detector. With this information, one could then determine the neutrino mass hierarchy and  $\delta_{CP}$ , the CP-violating phase, simultaneously [1].

## 2.3 Liquid Argon Time Projection Chambers

Liquid argon time projection chambers (LArTPCs) are measurement devices often used in high energy physics experiments to measure ionizing radiation. They are an instance of a general class of measurement devices called time projection chambers which are used to measure the three dimensional position of ionizing particles. As neutrinos pass through the liquid argon, they have a small probability of interacting with it. Such interactions produce a number of different particles. The initial location of these particles is usually referred to as a 'vertex' [3]. In TPCs, a great deal of effort is employed to apply a spatially constant electric field across the medium. Because of the nature of liquefied argon, electrons have a drift speed several orders of magnitude higher than their partner Argon ions  $\text{Ar}^+$ . A charge-readout device is used as the anode plane to measure the x- and y-positions of the drift electrons as well as their drift time. Because the electric field is constant, the z-positions of the electrons can be interpolated from these drift times, assuming knowledge of relative event interaction time (which can be determined using scintillation light) [8]. Figure 2 depicts the proposed LArTPC design for the DUNE near detector. It will use a pixelated array of electrodes to measure charge depositions and will have an (x, y, z) scale of 3 meters by 5 meters by 7 meters [9].

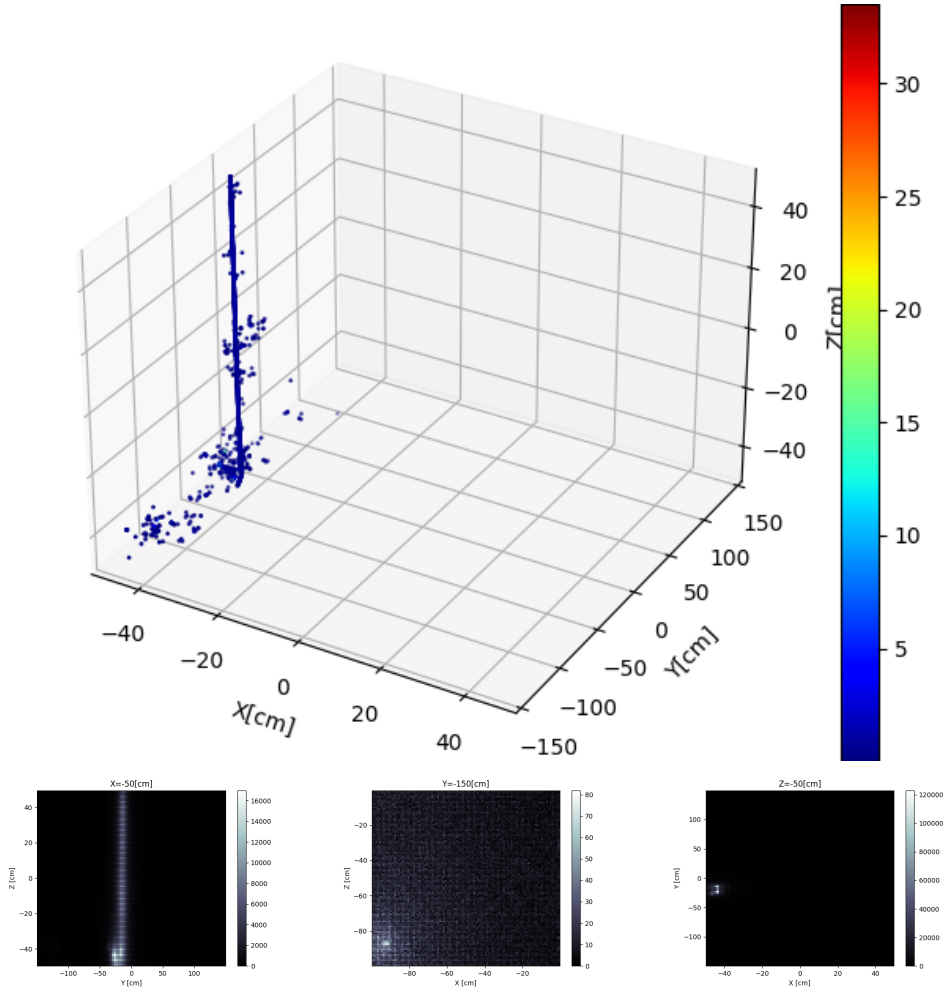


Figure 3: Simulated charge depositions and scintillation light for a neutrino interaction.

The neutrino beam will be produced in 'spills', or time increments roughly 10 microseconds long. In this short period, it is expected that roughly twenty to sixty interaction events will occur in the thirty five LArTPC modules, with additional events occurring outside the active detector volumes and spilling into the modules. These types of events are called fiducial and non-fiducial, respectively. Since the drift speed of the electrons within the liquid argon is approximately 1.6 millimeters per microsecond, the charge signals as measured by the pixelated readout plane will appear to have occurred simultaneously [2]. Therefore, accurate measurements of scintillation light signal are critical for high accuracy associations between charge and light signals.

## 2.4 Scintillation Light

As neutrinos interact with the liquid argon medium of a LArTPC, the charged daughter particles can excite the argon atoms to a higher energy state instead of ionizing them. These excited atoms eventually emit scintillation photons as they return to their ground state. Additionally, the recombination process occurring between an  $\text{Ar}^+$  ion and an electron will also emit scintillation light. Conveniently, liquid argon will not absorb its own scintillation light, allowing it to be measured at the inner surfaces of the detector with light-readout devices like SiPMs [8]. Figure 3 shows a simulated neutrino interaction in liquid argon and the scintillation light emitted by it detected on the detector walls.

As excited argon singlet and triplet states decay to their ground states, a scintillation photon with wavelength 128 nm is released. The lifetimes of these emissions are several orders of magnitude

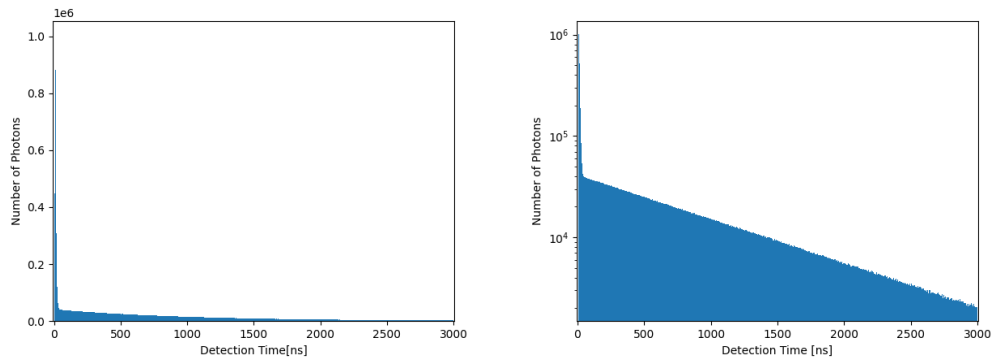


Figure 4: Detection time of the scintillation light emitted by a neutrino interaction in linear-linear on the left and log-linear scale on the right. The fast component is visible as the early spike in the number of detected photons lasting only a few nanoseconds, while the slow component appears as a slow decay occurring over a few microseconds. The difference in intensity between the two components is visible in the log-linear plot to the right.

apart: the singlet has a lifetime of 5 ns and the triplet has a lifetime of  $1 \mu\text{s}$ . The resulting combined light signal is generally modeled as a sum of Poisson processes, irrespective of the excitation process [7], [10]. An example of a typical scintillation light profile is presented in Figure 4. By imposing a 25 nanosecond time window, the 'fast' component of this scintillation light can be used for light signal determination with time resolution on the order of nanoseconds. One can thus very easily separate the fast light signals associated with any two given events.

The LArTPC liquid argon volume is tiled with two types of photon detectors with distinct modes of operation: ArCLight and LCM. The use of these two variants in tandem provides light signal localization in addition to timing information, so we are able to glean some information about event topology from the scintillation light measurements themselves. This implicit light-topology association enables the class of models we introduce for the pileup resolution task [11].

## 2.5 Event Reconstruction

The ultimate goal of liquid argon time projection chambers is to enable rich physics analysis of the neutrino interactions that occur within their volume. Generally, the types of questions physicists are interested in answering about these events are twofold: identifying the particular charge depositions belonging to a given interaction and identifying the various particle types present in the detector volume. Determining these types of deep, global features is generally not performed in a single step, due to, in part, the high complexity of simultaneously providing semantic (or species) labels and instance (or interaction) labels for piled-up event topologies. First, more local features like interaction vertices and particle-level clustering are determined using a feature extractor like a U-Net. These local features can be gradually combined and refined into the desired global features in a number of ways, but graph neural networks have proven to have very high performance on this task [3].

## 2.6 Feature Extractors

Recent advances in deep learning have enabled computer vision algorithms to reach extremely high levels of accuracy on tasks like classification and segmentation, even surpassing human benchmarks [12]. An extremely common tool that acts as the backbone for most modern computer vision algorithms is the convolutional neural network (CNN). CNNs can generally be understood to act as 'feature extractors', calculating features on their inputs with varying levels of locality. Such features range from low-level local edges and corners to deep, global traits like shape, orientation, and scene [13].

U-Net is a variant of the original CNN devised in 2015 by Ronneberger et al. for medical imaging that has become massively popular for a wide range of computer vision tasks, including segmentation [14]. It acts in a similar fashion to a typical convolutional neural network: it takes an image as



input and applies layers of convolutions to this input to calculate deep features. However, by intelligently leveraging concatenation operations on the outputs of their convolutional layers and applying transposed convolutions, U-Net models and their derivatives are able to produce feature maps at the original spatial dimensions of the input image. This enables U-Nets to produce high resolution segmentations, making it a very effective solution for a variety of tasks [15].

In general, U-Net models work as follows: the first half of the model works like a typical CNN - iteratively applying convolution and pooling operations to some pre-determined 'depth'. At its deepest point, the model produces tensors with a much lower spatial resolution than the input image, but a large number of feature channels. After this first half of the model has produced its output tensors, the second half of the model applies some type of upsampling operation, like transposed convolution or unpooling, followed by a standard convolution operation to gradually restore the tensor to its original spatial size. As the tensor travels through the decoder layers, it is concatenated with the output tensors of the corresponding encoder layers, restoring more spatial information that was discarded during encoding. At its final layer, the U-Net produces a tensor with the same spatial size as the original input and some number of feature channels, which can be interpreted in a number of ways, including as unnormalized class scores for classification or segmentation tasks [15].

Later improvements to U-Net include the addition of residual connections as in He et al.'s seminal 2015 paper, three-dimensional reformulations, and conversion to sparse convolutions [16], [17]. These additions allowed for much deeper networks to calculate richer sets of features on three dimensional inputs, enabling these techniques to be applied to the types of three dimensional data that appears in tasks like neutrino interaction event reconstruction [16].

## 2.7 Sparse Convolution

Sparse convolutions are an implementation of the convolution operation for sparse tensors. Sparse operations on tensors have proved massively important for performing deep learning in higher dimensions as they significantly reduce the amount of memory used by a network. In particular, they prevent the memory used by a network from scaling exponentially with the dimensionality of the data being used, a problem referred to as the curse of dimensionality [18]. For example, in the standard representation, the amount of memory required to store a three dimensional tensor with side length  $l$  is at least  $l$  times more than the amount required for a two dimensional tensor with the same side length. However, sparse representations only track the total number of 'active', or non-zero, locations in these tensors, so the multiplicative memory cost associated with the increase in dimensionality above is at most  $l$  and very likely significantly less [19].

They have become particularly useful for analyzing LArTPC data, wherein less than 1% of the voxels in the entire detector volume contain some amount of deposited charge. Additionally, since the computation required for sparse convolutions scales only with the number of active sites and not the total volume size, it is a strong candidate for the three hundred ton LAr volume present at the DUNE near detector site [2], [19].

In 2019, Dominé and Terao introduced their Sparse U-ResNet architecture and provided semantic segmentation performance benchmarks on a public dataset of liquid argon particle interactions, demonstrating high efficiency and purity while using relatively low amounts of computation [16]. To calculate their sparse operations, they used Facebook's public SparseConvNet library [20]. Their Sparse U-ResNet was later used as a feature extractor by Drielsma et al. in their reconstruction chain to calculate initial clusterings and points of interest [3].

Internally, SparseConvNet tracks the history of sparsity patterns of tensors as they are downsampled with convolutions and pooling operations which enables later upsampling operations to maintain the same sparsity pattern. This step is critical for maintaining a constant number of active sites in the input, avoiding the 'active diffusion' that occurs with standard convolution operations [20].

## 3 Related Work

In 2016, the MicroBooNE collaboration produced a study applying convolutional networks to the outputs of the MicroBooNE LArTPC. Because the MicroBooNE detector used wireplanes in place of pixelated charge readouts, it produced three two dimensional images corresponding to the detector's two induction planes and single collection plane. These images, each containing on the order of 20

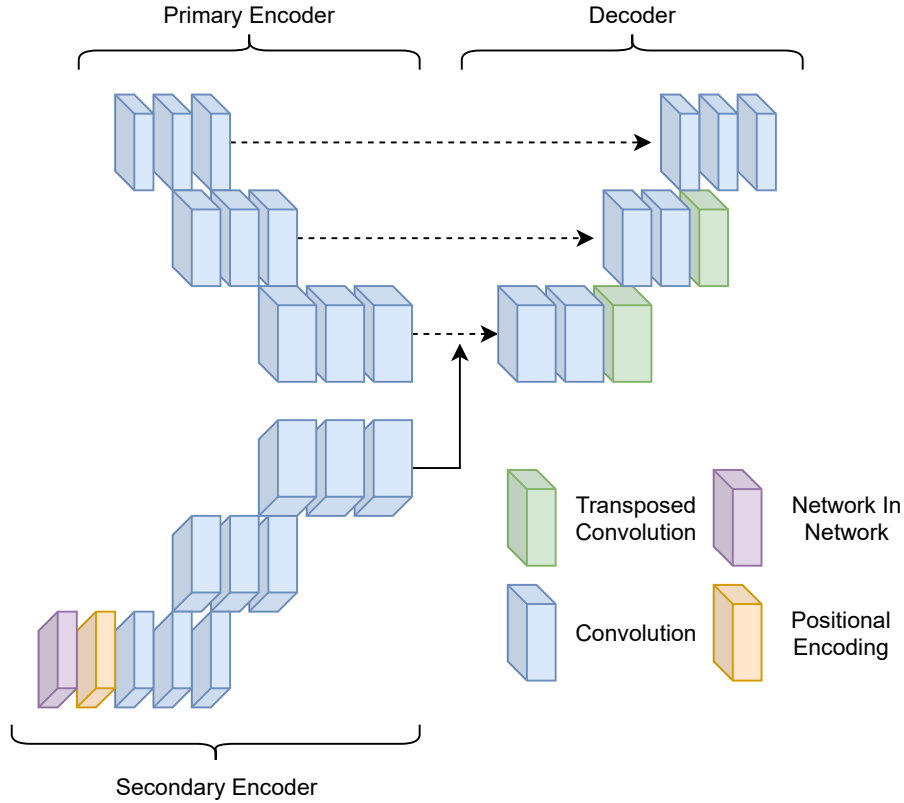


Figure 5: Y-Net model for segmentation tasks using charge depositions and scintillation light. The model in this diagram is three blocks 'deep'. The upper encoder block is referred to as the primary encoder path and the lower encoder block is referred to as the secondary encoder path. The features produced by these two paths are combined in the model's latent space, then upsampled back along the model's decoder path up to the original spatial size of the input. The dotted lines denote concatenation operations and the single solid line represents the pooling-unpooling operation performed on the secondary encoder's latent features.

million pixels, were too large to fit into memory directly, so the team was forced to downsize the images and discard some amount of spatial information. However, they were able to demonstrate high performance on tasks including localization and particle classification using GoogLeNet on these images [21].

The MicroBooNE collaboration produced another study in 2018 investigating applications of U-Net. Because of U-Net's full-resolution outputs, the collaboration was able to produce pixel-level class predictions, resolving tracks, showers, and background pixels. They used a variant of U-Net with residual connections called U-ResNet that achieved 6.0% and 3.9% incorrectly classified pixel fraction on  $\nu_e$  and  $\nu_\mu$  interactions, respectively [22].

In February 2021, Drielsma et al. presented their end-to-end reconstruction chain, intended to be used for identifying neutrino interactions and performing particle identification from charge depositions alone. They use two U-ResNets in parallel to calculate initial sub-particle clusterings and to identify points of interest in the charge depositions. These segmentation outputs are then fed into a pair of graph neural networks acting in series that refine the segmentation outputs, ultimately producing interaction instance labels and particle species labels for each voxelized charge deposition in the initial input. Their end-to-end approach achieves an average of 98.7 % purity and 99.5 % efficiency [3].

## 4 Design

To make use of the relationship between scintillation light and charge deposition topology in the DUNE LArTPC measurements, we propose Y-Net. We consider Y-Net to be a descendant of the U-Net family of models, as its operating principle is the same: calculate full-resolution features on some image. Y-Net is most closely related to Sparse U-ResNet since we leverage sparse convolutions and residual connections to reduce computation requirements and improve performance. Y-Net’s principal distinguishing trait is its ‘secondary’ encoder path. In addition to encoding the charge depositions through its ‘primary’ encoder path, Y-Net also encodes the scintillation light into the same latent space. Then, these features are combined in the latent space, and travel through the single decoder path back to the original sparsity pattern of the charge depositions, making Y-Net capable of producing the same types of outputs as Sparse U-ResNet [16]. A diagram of a depth-three Y-Net is shown in Figure 5.

Of primary importance in the model design is a strategy for integrating the timing information carried by the scintillation light signal. As mentioned previously, it is known that the light signal corresponding to any given interaction can be separated with high accuracy from that of any other interaction. Thus, for this study, we assume that this preprocessing has already been performed on the light signal of each interaction and that we can assign some distinct relative interaction time to each.

To leverage these timing labels, we employ a strategy similar to that of Vaswani et al. [23]. In particular, we add timing encodings represented as sine and cosine functions of the interaction time labels  $t$  to the tensors:

$$PE(t, c) = \begin{cases} \sin \frac{t}{10000^{\frac{2c}{d_{\text{model}}}}}, c & \text{mod } 2 = 0 \\ \cos \frac{t}{10000^{\frac{2c}{d_{\text{model}}}}}, c & \text{mod } 2 = 1 \end{cases} \quad (2)$$

where  $t$  is the relative interaction time,  $c$  is the channel dimension, and  $d_{\text{model}}$  is the number of channels in the secondary tensor. As shown in Vaswani et al., using sines and cosines of varying frequencies makes it much easier for the model to identify distinct light signals because  $PE(t, c)$  is a linear function of  $PE(t + \Delta t, c)$  for fixed  $\Delta t$  [23]. This holds because the timing encoding function in (2) is designed such that the frequencies follow a geometric series from  $2\pi$  to  $10000 \cdot 2\pi$ : the exponential form of the denominator of both terms leads to this property. We choose 10,000 as the base of this exponential form because it defines a range large enough to easily distinguish the timing encodings associated with distinct timing labels or distinct channel dimensions.

We opt for these fixed timing encodings in place of learned encodings because we expect the number of fiducial events to vary greatly and learned encodings may make the model more sensitive to changes in this quantity [23]. We opt to add these timing encodings to the secondary tensor not immediately, but after the tensor has first been passed through a network-in-network block which acts as an initial feature encoder [24].

Because we ultimately want to combine the learned features along the primary and secondary encoder paths in the latent space, both encoder branches have the same ‘depth’, or number of convolutional and pooling blocks. This trait allows us to combine the two feature maps in this latent space. To combine features, we follow the strategy presented in Figure 6. First, we map the active sites of the latent secondary tensor onto the latent primary tensor’s active sites. To do this, we perform an additional pooling operation on the latent secondary tensor, replace its sparsity pattern history with that of the latent primary tensor, then unpool back onto the latent primary tensor’s sparsity pattern. This trick allows us to directly add the features learned on the secondary tensor onto the primary tensor.

## 5 Experiments

In this section, we introduce the set of tests we use to evaluate Y-Net against U-ResNet.

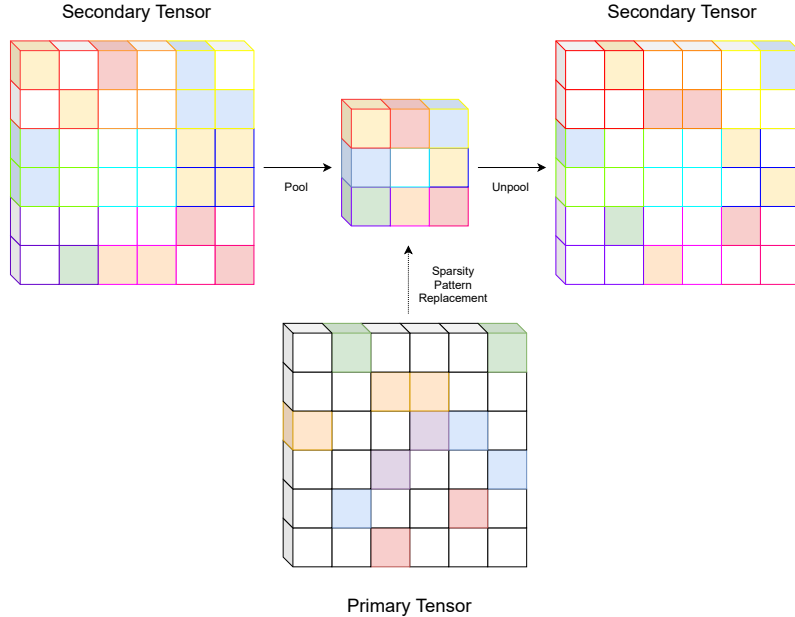


Figure 6: Feature mapping strategy for Y-Net. The secondary tensor is first downsampled to some lower resolution with a pooling operation. Then, the sparsity pattern history of the secondary tensor is replaced by that of the primary tensor. As a result, an unpooling operation on this downsampled secondary tensor returns it to its original spatial size, but with the same sparsity pattern as the primary tensor. This allows the features of the two sparse tensors to be added directly. In the figure, the dotted line from the primary tensor to the downsampled secondary tensor represents the sparsity pattern replacement operation.

## 5.1 Tasks

We test Y-Net on a dense clustering task, related to the one used by Drielsma et al. [3]. In this task, we aim to only cluster together voxels belonging to a common instance, penalizing predictions lumping voxels of different instances together. Ultimately, the utility of a model calculating these types of dense clusters is in a larger reconstruction chain, wherein the dense clusters can be refined into larger, interaction event level classes, possibly segmented by semantic class.

## 5.2 Dataset

We use a novel dataset of 26,083 neutrino interactions, simulated using the GENIE Neutrino Monte Carlo Generator. The dataset is split into 20,857 training events and 5,226 validation events. We select a beam profile that matches that of the neutrino beam at DUNE’s near detector site: it has a peak energy of 3 GeV and is primarily made up of muon neutrinos. GENIE produces the final-state particles and their trajectories and energies for each neutrino interaction [25]. This information is then fed into a Geant4 and EDepSim simulation of the near detector LArTPC modules. The final output of this simulation chain is a list of active voxels within the detector volume, the energy deposited at each of these active voxels, and some metadata about the final-state particles [26], [27]. Figure 7 shows three events from the validation dataset and the instance segmentations predicted by Y-Net on these events.

To calculate the scintillation light emitted by these events, we make use of Greenberg et al.’s light simulation. This simulator takes charge depositions as input and produces measurements of the time-varying scintillation light signal on the sides of the detector volume, assuming a geometry wherein the detector walls are tiled with  $3 \times 3 \text{ mm}^2$  SiPMs. Internally, photon production is modeled as a combination of Argon excitation and recombination, the latter of which is in turn modeled as a Doke-Birks process. After photon production in simulation, photons are individually propagated

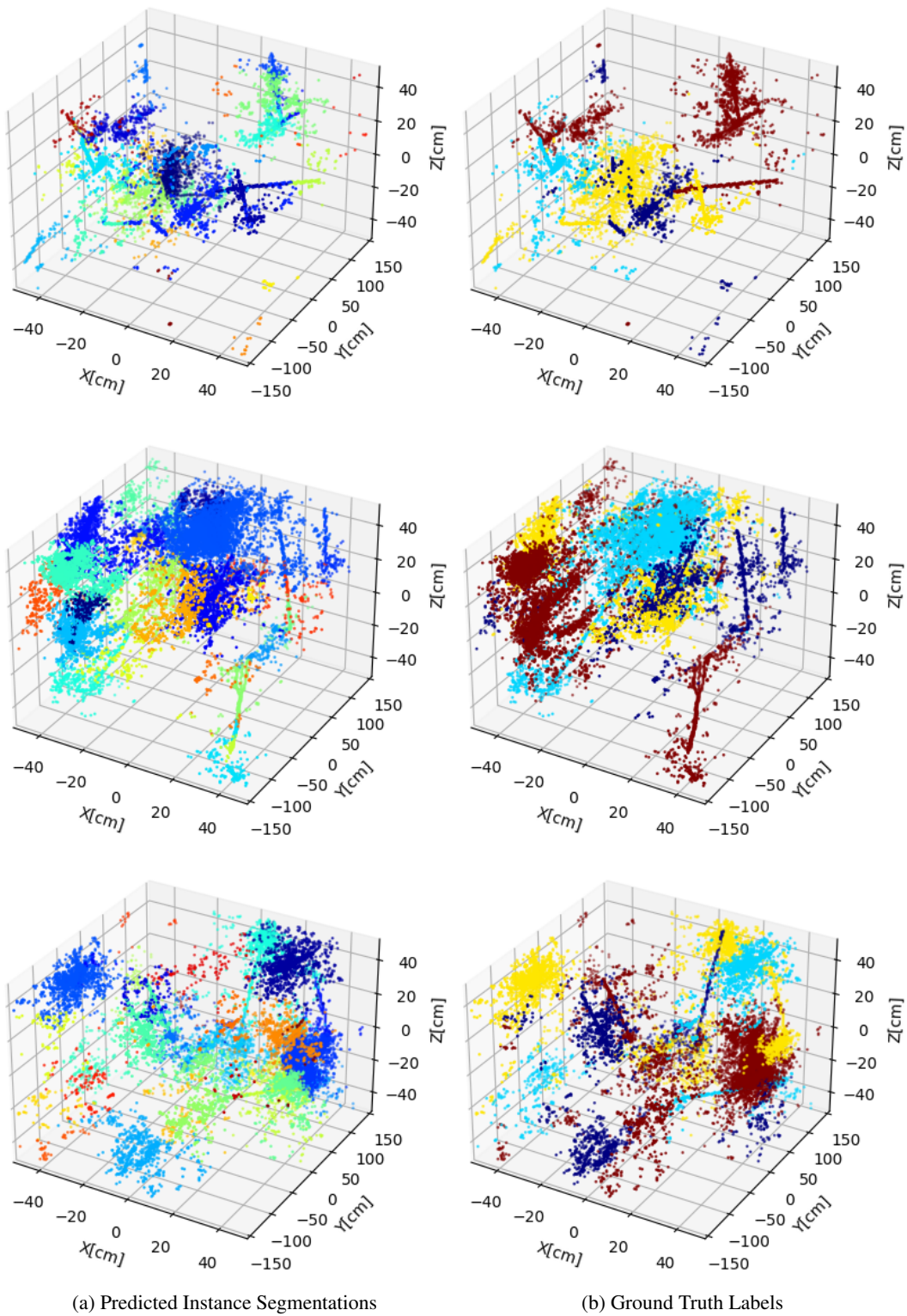


Figure 7: Three events from our validation set. The left column contains predicted instance segmentations produced by Y-Net and the right column contains the ground truth event labels. There are four interactions in each instance. Note that the ground truth and predicted segmentation colors are not necessarily the same.

from their production site to the detector walls where their propagation time is recorded by the SiPMs [28].

### 5.3 Metrics

To evaluate and compare the performance of Y-Net and U-ResNet, we use efficiency, purity, and binary intersection-over-union, common clustering metrics. We define efficiency on a per-interaction basis as the percentage of a given interaction correctly predicted as belonging to the same cluster. We similarly define purity on a per-predicted cluster basis as the percentage of a predicted cluster belonging to the same interaction.

Intersection-over-union (IoU) is a popular clustering metric that represents the similarity between the ground truth and predicted label sets. We use the binary intersection-over-union, averaged across all true clusters.

### 5.4 Training

Depth	Y-Net	U-ResNet
2	378752	222112
3	1207872	789664
4	2824640	1938880
5	5496320	3877632
6	9490176	6813792
7	15073472	10955232
8	22513472	16509824

Table 1: Number of parameters used by Y-Net and U-ResNet at various depths. Y-Net’s secondary encoder branch adds another roughly 50% more parameters to the model.

We trained depth-5 U-ResNet and Y-Net models on our dataset for ten epochs, or roughly 200,000 iterations, using a masked variant of the Lovász hinge loss on an RTX2070. Each model was built with dropout and batch normalization to stabilize training. Plots of the Y-Net’s training performance are included in Figure 8. Interestingly, efficiency and purity converge in only a few iterations, but intersection-over-union takes much longer to reach its maximum value. This trend implies the loss function used to train the model implicitly places a higher penalty on low purity clusterings than on clusterings with low intersection-over-union score.

As in Drielsma et al., to generate segmentation predictions from these model, we define each to produce an ‘embedding’ feature map and a ‘seediness’ feature map [3]. The ‘embedding’ feature map is interpreted as a transformation on the coordinates of the active voxels in the original input, embedding these voxels into some lower dimensional space. The ‘seediness’ layer is then interpreted as the probability that voxels are embedded close to voxels of the same cluster. Using these two layers, we can then employ the following algorithm to calculate instance labels:

First, we sort the voxels by their seediness values. Then, in descending order, we determine the voxel with the next highest seediness. We define a Gaussian kernel using this voxel’s embedding feature, then assign some unique instance label to every other voxel with probability, as defined by the Gaussian kernel, above some predetermined threshold  $p_0 = 0.5$ . We remove all these labeled voxels from our pool and repeat, terminating when every voxel has been assigned a unique label. We can then interpret these as instance labels. The labelling algorithm is presented in Algorithm 1.

Training U-ResNet on our GPU took approximately six hours, while training Y-Net took about fifteen hours.

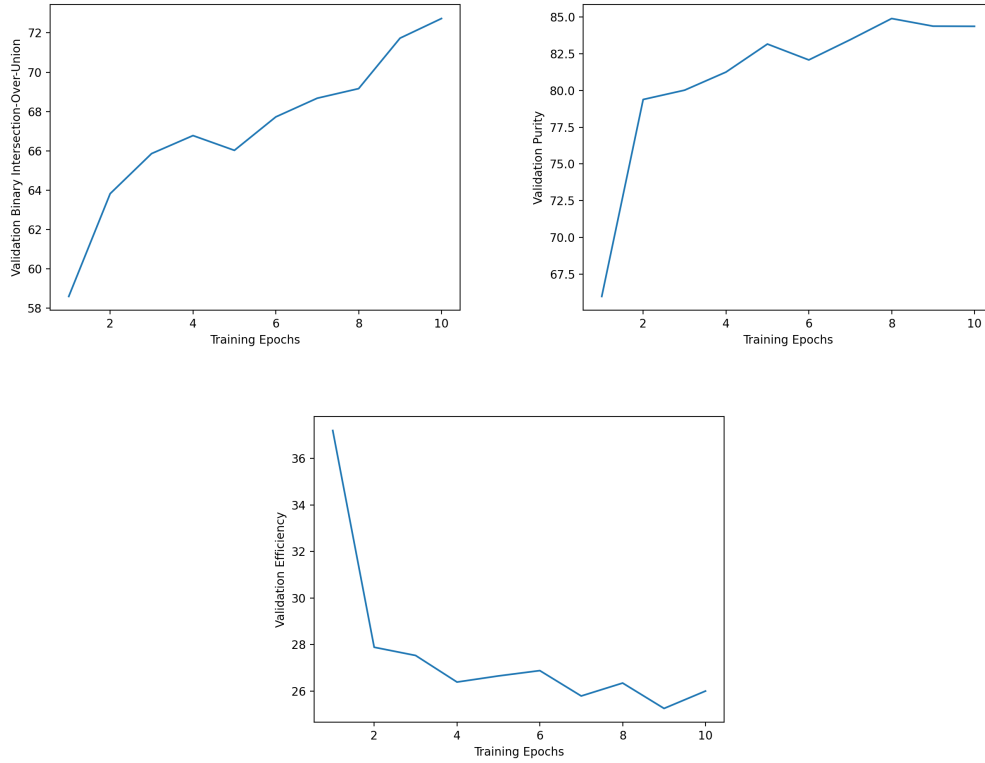


Figure 8: Validation binary intersection-over-union, purity, and efficiency metrics from training Y-Net for ten epochs.

---

### Algorithm 1 Instance Labelling

---

```

1: procedure LABEL( $e[]$ ,  $s[]$ ,  $m[]$ )
2:   ( $e[]$ ,  $s[]$ ,  $m[]$ )  $\leftarrow$  sort(( $e[]$ ,  $s[]$ ,  $m[]$ ))
3:   while not all points labelled do
4:      $\mu$ ,  $s$ ,  $\sigma \leftarrow$  max(( $e[]$ ,  $s[]$ ,  $m[]$ ))
5:      $p \leftarrow p(\cdot; \mu, \sigma)$ 
6:     instance  $\leftarrow []$ 
7:     for (emb, sed, mar) in ( $e[]$ ,  $s[]$ ,  $m[]$ ) do
8:       if  $p(\text{emb}) > p_0$  then
9:         Add point to instance
10:        remove emb, sed, mar from  $e[]$ ,  $s[]$ ,  $m[]$ 
11:      end if
12:    end for
13:    remove  $\mu$ ,  $s$ ,  $\sigma$  from  $e[]$ ,  $s[]$ ,  $m[]$ 
14:  end while
15: end procedure

```

$\triangleright$  embeddings, seediness, margins  
 $\triangleright$  Sort by seediness in descending order  
 $\triangleright$  Max over  $s$   
 $\triangleright$  Define Gaussian kernel  
 $\triangleright$  Same instance if probability above threshold

---

## 5.5 Results

After training, we observed a mild improvement to validation intersection-over-union, purity, and efficiency, mostly at the highest levels of pileup. All of the measured results are presented in [Table 2](#). We observe improvements of 4.58 %, 0.352 %, and 2.72 % in intersection-over-union, purity, and efficiency, respectively, at sixteen events, the highest interaction multiplicity. However, at lower interaction multiplicities, U-ResNet performs either similarly to or slightly better than Y-Net.

Interactions	Y - IoU	U - IoU	Y - Purity	U - Purity	Y - Eff.	U - Eff.
1	88.12 %	<b>89.52 %</b>	100 %	100 %	<b>27.93 %</b>	27.24 %
2	82.55 %	<b>83.473 %</b>	<b>94.36 %</b>	94.26 %	26.58 %	<b>27.20 %</b>
3	77.25 %	<b>77.33 %</b>	<b>89.173 %</b>	89.04 %	25.64 %	<b>26.04 %</b>
4	<b>72.73 %</b>	72.09 %	<b>84.36 %</b>	84.33 %	<b>26.00 %</b>	25.59 %
5	<b>67.91 %</b>	66.70 %	<b>80.043 %</b>	80.09 %	25.23 %	<b>25.24 %</b>
8	<b>57.69 %</b>	55.39 %	<b>69.428 %</b>	69.00 %	<b>24.73 %</b>	23.77 %
16	<b>41.49 %</b>	36.91 %	<b>51.252 %</b>	50.90 %	<b>25.54 %</b>	22.82 %

Table 2: Y-Net and U-ResNet validation binary intersection-over-union with varying levels of pileup

## 6 Discussion

Our results imply that Y-Net and U-ResNet have comparable performance at low interaction multiplicities as one might expect. With lower levels of pileup, previous studies have shown that resolving overlapping interaction events based on charge depositions alone is known to be actually quite performant [3], [16].

However, as event multiplicity increases, the gap in performance between Y-Net and U-ResNet grows. Indeed, the combined information about topology and timing carried by the scintillation light becomes increasingly important as the number of piled-up events increases.

Because we expect the utility of a network like Y-Net or U-ResNet not to be as an end-to-end event reconstruction model but as a submodule in a larger reconstruction chain, the low clustering efficiencies presented by both models is not particularly alarming. As in Drielsma et al., these low efficiency predictions can be easily lumped together into high efficiency predictions using graph neural networks [3]. As a result, we only aim to produce clusters with high purity so that these later refining models will in turn produce larger clusters with high purity. Alternatively, to avoid the extra computation required for these graph neural networks, a simpler classical clustering algorithm, like DBSCAN, could be used to produce more refined clusters [29].

An important consideration is the large increase in memory use Y-Net presents. Training Y-Net was not straightforward, as the model barely fit onto our GPU. Fastidious out-of-memory errors were not uncommon and required careful handling to address. We saved models frequently and often had to revert training to a recent checkpoint after an error.

### 6.1 Future Work

The utility of Y-Net is that it presents a learnable, first-order approach to light-charge association, but it requires nearly fifty percent more memory for only a minor increase in performance. We expect that despite the large increase in parameters Y-Net has relative to U-ResNet, U-ResNet alone is likely expressive enough to perform these types of high resolution segmentations. However, more work would be required to find strategies for determining some kind of relative timing tag for each active voxel.

To further improve Y-Net’s performance, other instances of Y-Net with greater depth could be explored. The increased receptive field of the network would likely improve its ability to learn useful features along the secondary encoder branch. However, with these deeper networks, careful memory use will be more important. To enable these deeper models to fit in memory in modern computers, more work could be done on increasing the sparsity of the scintillation light signal while maintaining as much timing and intensity information as possible.

Additionally, to make better use of the scintillation light information, future work could be done to explore different techniques to combine the primary and secondary latent features, including convolutional feature mappings or feedforward feature mappings.

An important next step is to study light-charge association under more realistic light simulations to help resolve pileup with reduced topological information, perhaps with optical simulations like P. Koller’s ArgonCube Optical Simulation [30].



## 7 Conclusion

In this paper, we presented Y-Net, a model architecture for integrating scintillation light information into charge depositions to help resolve the neutrino interaction pileup problem of DUNE's LArTPC near detector. We motivate how the timing information provided by scintillation light can be used to address some of the more difficult instances of piled up interactions. Additionally, we present results showing that Y-Net improves over previous techniques on some metrics for segmenting neutrino interactions in liquid argon.

## Acknowledgments

I would like to thank my academic advisor Prof. Kam-Biu Luk for his help and patience with this project. I would also like to thank Professor Chris Marshall for his advice concerning the light simulation and Professor Gireeja Ranade for enabling this study by acting as an additional reader and as an on-paper advisor.

## References

- [1] R. Acciarri et al. [The DUNE Collaboration]. Long-Baseline Neutrino Facility (LBNF) and Deep Underground Neutrino Experiment (DUNE), Conceptual Design Report 1, 2016. URL <https://arxiv.org/pdf/1601.05471.pdf>.
- [2] D. Dwyer. DUNE Near Detector: Physics and System Overview. <https://indico.fnal.gov/event/43949/contributions/189448/attachments/130800/159733/Dwyer-NDPhysicsOverview-NDCDRJUL2020.pdf>, 2020.
- [3] F. Drielsma, K. Terao, L. Dominé, and D. H. Koh. Scalable, end-to-end, deep-learning-based data reconstruction chain for particle imaging detectors. *Machine Learning and the Physical Sciences*, (149), 2020.
- [4] L. M. Brown. The idea of the neutrino. *Physics Today*, 31(9):23–28, 1978. doi: <https://doi.org/10.1063/1.2995181>. URL <https://doi.org/10.1063/1.2995181>.
- [5] F. Capozzi, E. Lisi, A. Marrone, D. Montanino, and A. Palazzo. Neutrino masses and mixings: Status of known and unknown 3 nu parameters. *Nuclear Physics B*, 908:218–234, 2016. doi: <https://doi.org/10.1016/j.nuclphysb.2016.02.016>. URL <https://www.sciencedirect.com/science/article/pii/S0550321316000602>, Neutrino Oscillations: Celebrating the Nobel Prize in Physics 2015.
- [6] S. Bilenky. Neutrino oscillations: From a historical perspective to the present status. *Nuclear Physics B*, 908: 2–13, 2016. doi: <https://doi.org/10.1016/j.nuclphysb.2016.01.025>. URL <https://www.sciencedirect.com/science/article/pii/S0550321316000353>, Neutrino Oscillations: Celebrating the Nobel Prize in Physics 2015.
- [7] P. Adhikari et al. [The DEAP Collaboration]. The liquid-argon scintillation pulseshape in deap-3600. *Eur. Phys. J. C*, 80(4):303, 2020. doi: 10.1140/epjc/s10052-020-7789-x. URL <https://doi.org/10.1140/epjc/s10052-020-7789-x>.
- [8] M. Schenk. *Studies with a Liquid Argon Time Projection Chamber*. Springer Spektrum, Weisbaden, 2015. doi: <https://doi.org/10.1007/978-3-658-09430-0>.
- [9] D.A. Dwyer, M. Garcia-Sciveres, D. Gnani, C. Grace, S. Kohn et al. LArPix: demonstration of low-power 3d pixelated charge readout for liquid argon time projection chambers. *Journal of Instrumentation*, 13(10): P10007–P10007, Oct 2018. doi: 10.1088/1748-0221/13/10/p10007. URL <https://doi.org/10.1088/1748-0221/13/10/p10007>.
- [10] M.J. Carvalho and G. Klein. Luminescence decay in condensed argon under high energy excitation. *Journal of Luminescence*, 18-19:487–490, 1979. doi: [https://doi.org/10.1016/0022-2313\(79\)90167-4](https://doi.org/10.1016/0022-2313(79)90167-4). URL <https://www.sciencedirect.com/science/article/pii/0022231379901674>.
- [11] M. Weber and D. Dwyer. DUNE ArgonCube ND LArTPC. [https://indico.fnal.gov/event/43949/contributions/189453/attachments/130807/159711/ArgonCube-NDLArTPC\\_CDR-Weber.pdf](https://indico.fnal.gov/event/43949/contributions/189453/attachments/130807/159711/ArgonCube-NDLArTPC_CDR-Weber.pdf), 2020.
- [12] Papers with Code. Papers with Code - Image Classification on ImageNet. <https://paperswithcode.com/sota/image-classification-on-imagenet>, 2020.
- [13] F. Li, A. Karpathy, and J. Johnson. Lecture 7: Convolutional Neural Networks. [http://cs231n.stanford.edu/slides/2016/winter1516\\_lecture7.pdf](http://cs231n.stanford.edu/slides/2016/winter1516_lecture7.pdf), 2016.

- [14] Papers with Code. Papers with Code - U-Net Explained. <https://paperswithcode.com/method/u-net>, 2020.
- [15] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. ISBN 978-3-319-24574-4.
- [16] L. Dominé and K. Terao. Scalable deep convolutional neural networks for sparse, locally dense liquid argon time projection chamber data. *Phys. Rev. D*, 102:012005, Jul 2020. doi: 10.1103/PhysRevD.102.012005. URL <https://link.aps.org/doi/10.1103/PhysRevD.102.012005>.
- [17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. doi: 10.1109/CVPR.2016.90.
- [18] K. Weinberger. Lecture 2: k-nearest neighbors. [https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote02\\_kNN.html](https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote02_kNN.html), 2018.
- [19] B. Graham. Sparse 3d convolutional neural networks. In M. W. Jones X. Xie and G. K. L. Tam, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 150.1–150.9. BMVA Press, September 2015. ISBN 1-901725-53-7. doi: 10.5244/C.29.150. URL <https://dx.doi.org/10.5244/C.29.150>.
- [20] Facebook Research. SparseConvNet. <https://github.com/facebookresearch/SparseConvNet/>, 2017.
- [21] R. Acciarri et al. [The MicroBooNE Collaboration]. Convolutional neural networks applied to neutrino events in a liquid argon time projection chamber. *Journal of Instrumentation*, 12(03):P03011–P03011, Mar 2017. doi: 10.1088/1748-0221/12/03/p03011. URL <https://doi.org/10.1088/1748-0221/12/03/p03011>.
- [22] C. Adams et al. [The MicroBooNE Collaboration]. Deep neural network for pixel-level electromagnetic particle identification in the microboone liquid argon time projection chamber. *Phys. Rev. D*, 99:092001, May 2019. doi: 10.1103/PhysRevD.99.092001. URL <https://link.aps.org/doi/10.1103/PhysRevD.99.092001>.
- [23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, et al. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.
- [24] M. Lin, Q. Chen, and S. Yan. Network in network. *CoRR*, abs/1312.4400, 2014.
- [25] C. Andreopoulos, A. Bell, D. Bhattacharya, F. Cavanna, J. Dobson, et al. The genie neutrino monte carlo generator. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 614(1):87–104, 2010. doi: <https://doi.org/10.1016/j.nima.2009.12.009>. URL <https://www.sciencedirect.com/science/article/pii/S0168900209023043>.
- [26] Geant Collaboration. Geant4. <https://geant4.web.cern.ch/>, 1998.
- [27] edep-sim Collaboration. edep-sim. <https://github.com/ClarkMcGrew/edep-sim>, 2015.
- [28] S. Greenberg, D. Dwyer, and K. B. Luk. Simulation of Scintillation Light in LArTPCs. *BLUR*, 2020.
- [29] M. Ester, H. P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD’96*, page 226–231. AAAI Press, 1996.
- [30] P. Koller. ArgonCube Geometry and Optical Simulation University of Bern. DUNE LAr ND Analysis Meeting, September 9, 2020.