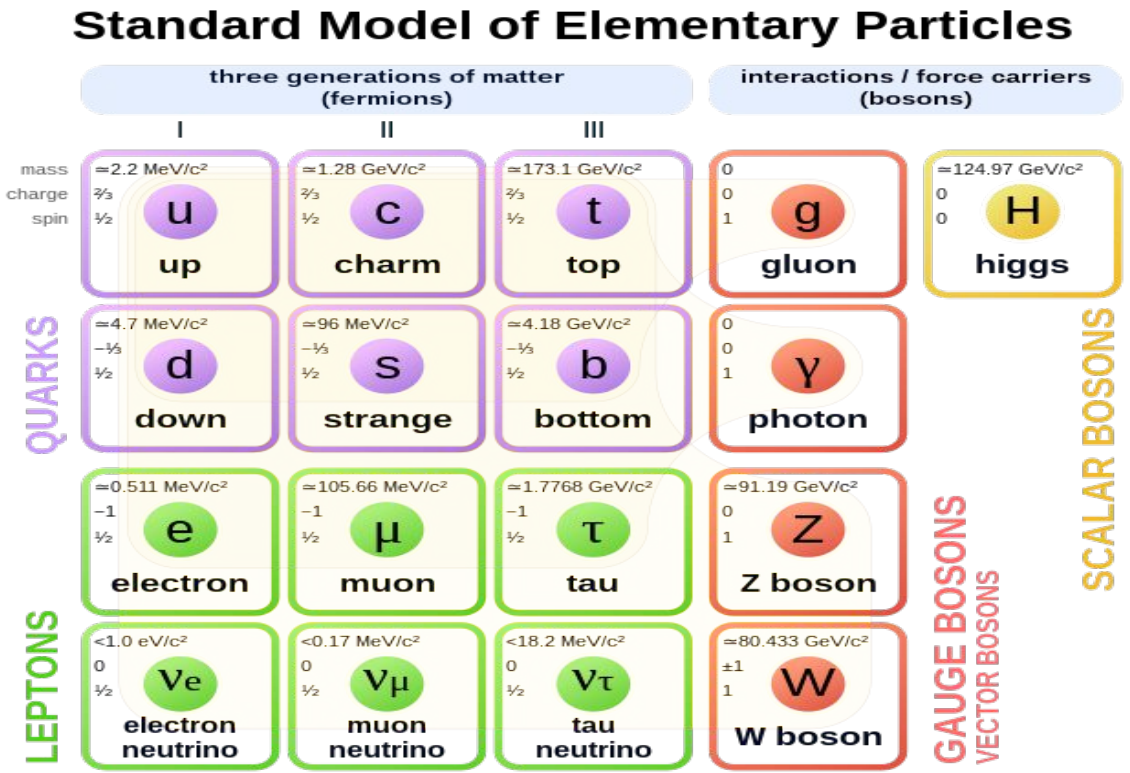
Application of R-CNN models to reconstruction of CRES events

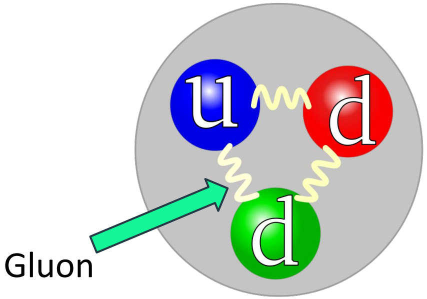
Winston DeGraw

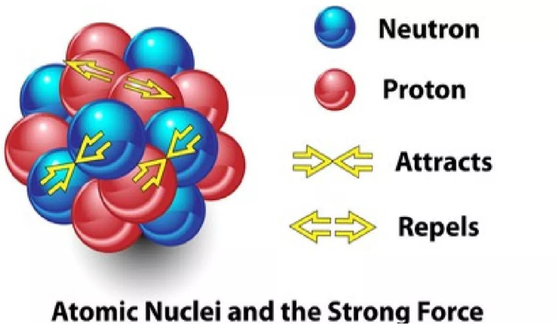
Our current understanding - the Standard Model



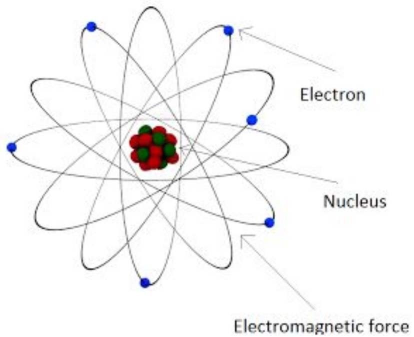
The fundamental forces

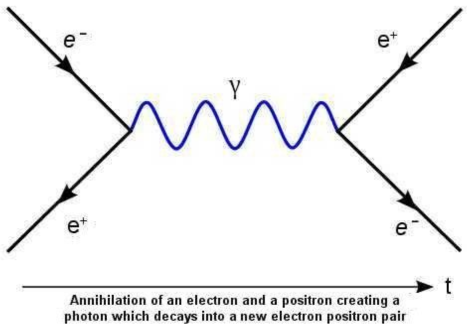
Strong Nuclear



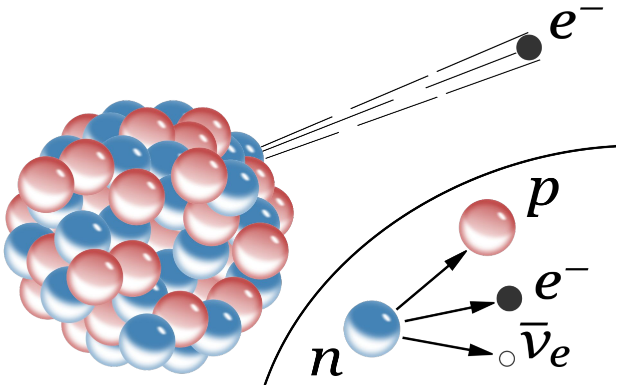


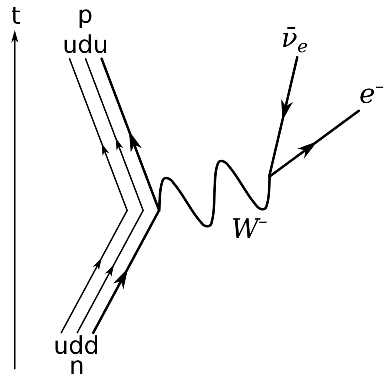
Electromagnetic





Weak Nuclear





That's cool and all but what about…?

Despite its successes, the SM leaves some big open questions:

How does gravity fit in?

Why is there more matter than antimatter?

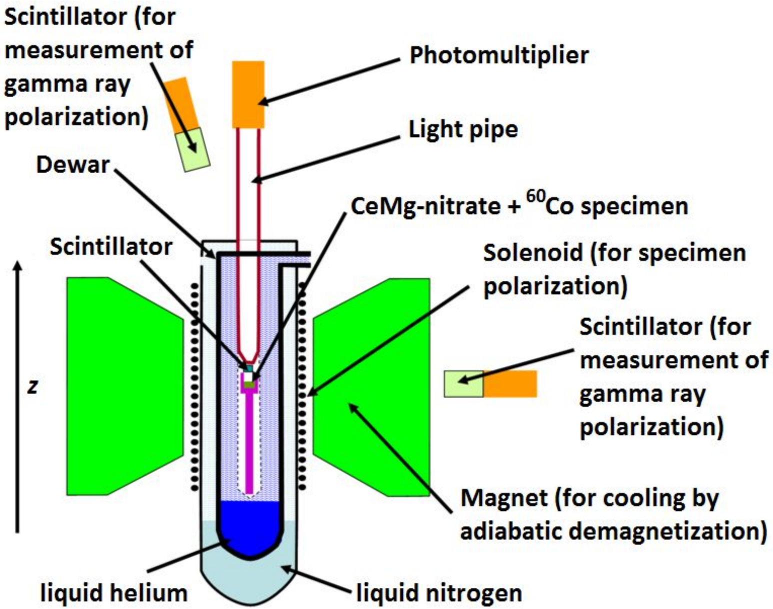
What is dark matter?

Hidden symmetries?

It is up to us to test the limits of this theory - see where it breaks

Nuclear Beta Decay - a history of breaking physics

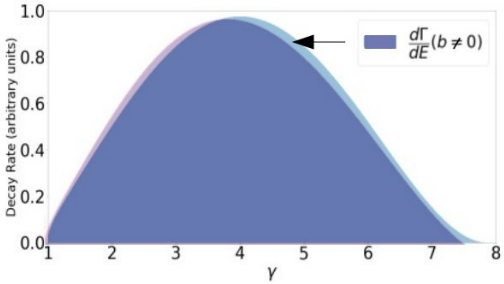
C.S. Wu (commonly referred to as Madame Wu)



Going Beyond the Standard Model with He6-CRES

- Modern iteration of studying beta decay for new physics
- Goal is to make world-leading precision measurement of the shape of the beta spectrum
- New limits on existence of extra terms in SM prediction can be made
    - No a priori reason for term circled in red to be 0 - unexplained symmetry

$$\frac{d\Gamma}{dE_e} = \frac{G_F^2 |V_{ud}|^2}{(2\pi)^5} p_e E_e (E_o - E_e)^2 \xi \left[1 + b\frac{m_e}{E_e} \right]$$



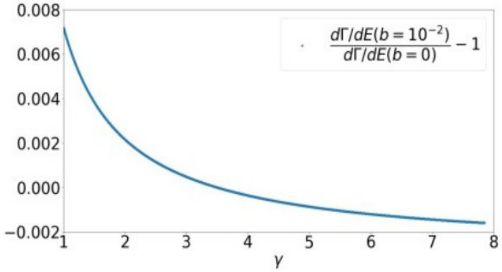
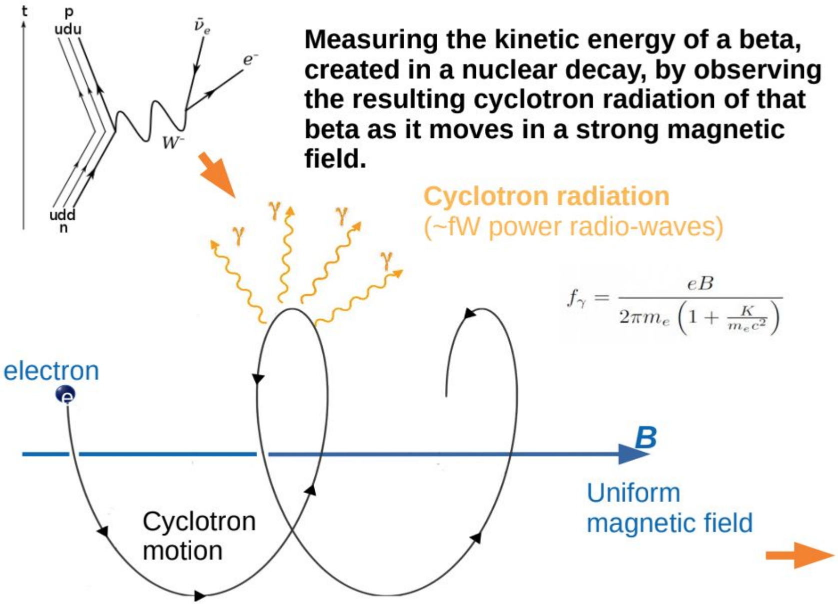


Image taken from Heather Harrington’s General Exam presentation

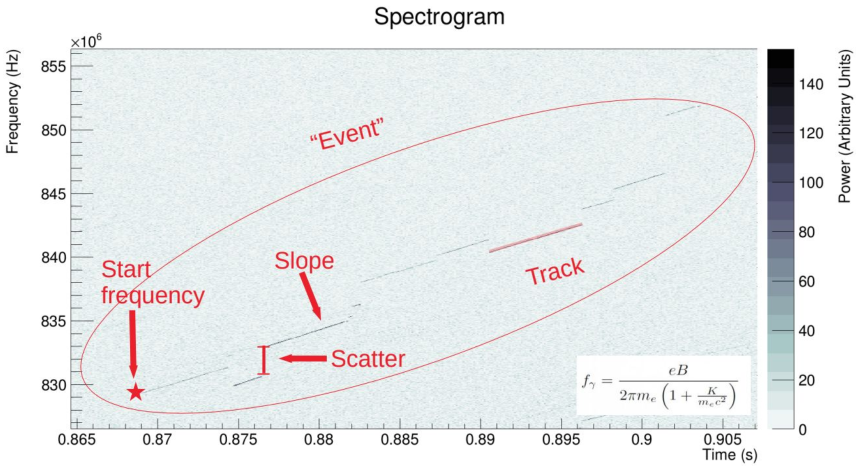
CRES Overview

- CRES = Cyclotron Radiation Emission Spectroscopy
- Novel technique - first developed by the Project-8 collaboration, we are second group ever to use it
- Fancy words for “spinning electron emits light and we measure its frequency/energy”
- Radioactive nuclei (for example 6He) put inside of two magnets: one for trapping, one for making it spin
- Emitted electron spinning/shining light the moment it is born - can measure energy at moment it was created



CRES Signal

- The very low power light digitized an ADC, and FPGA performs FFT before writing to disk
- So we don't write voltage, but look at a spectrogram of power put into frequency bins as a function of time
- Event structure can be quite rich, complicating reconstruction
- Enter the need for excellent object detection algorithms!

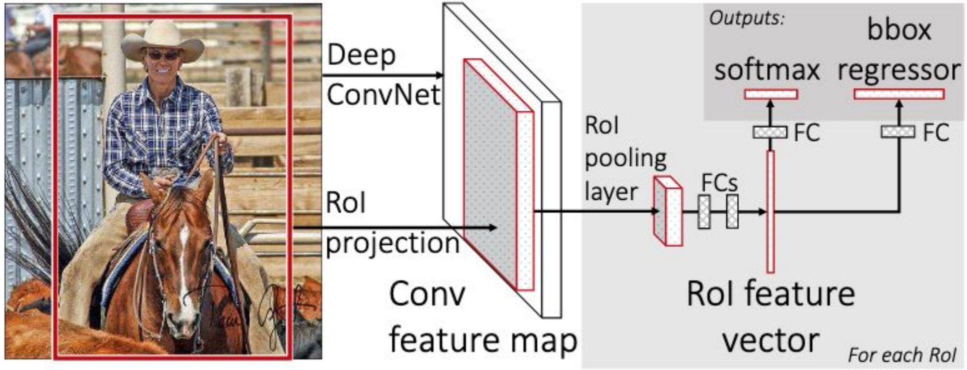


Region-based Convolutional Neural Networks (R-CNN) and Region Proposal Networks (RPN): Faster R-CNN

https://arxiv.org/abs/1506.01497

Fast R-CNN

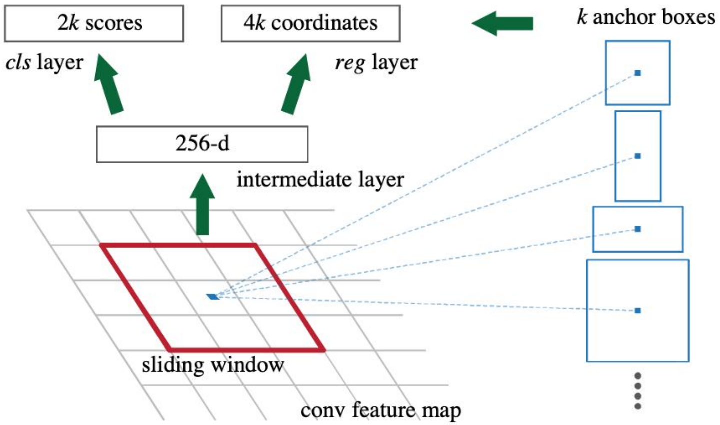
- In general, R-CNNs perform regression and classification on proposal regions
- Computationally expensive if done in several training iterations, or if proposal regions not chosen well
- Fast R-CNN solves the first problem with its single stage training step
- **Does not solve the second problem, in general**



[Fast R-CNN paper]

Region Proposal Network

- After applying convolutional layers to the input image a sliding window of anchor boxes is applied to the feature map
  - Each anchor is subset of image in sliding window
  - Regression applied for the corners of the output box, classification for the 'objectness'
- **Faster** than previous methods of region proposal
- **Translationally invariant**
- **Any input shape allowed**



Faster R-CNN

- Combines the RPN method (novel to the paper referenced before) with the Fast R-CNN method
- Convolutional layers shared between RPN and Fast R-CNN networks
- Object proposals from RPN and feature maps from shared conv. layers are input to Fast R-CNN
- **Improved accuracy and training time** compared to Fast R-CNN
- Output is a **boundary box** (bbox) that (hopefully) contains the object you are trying to detect

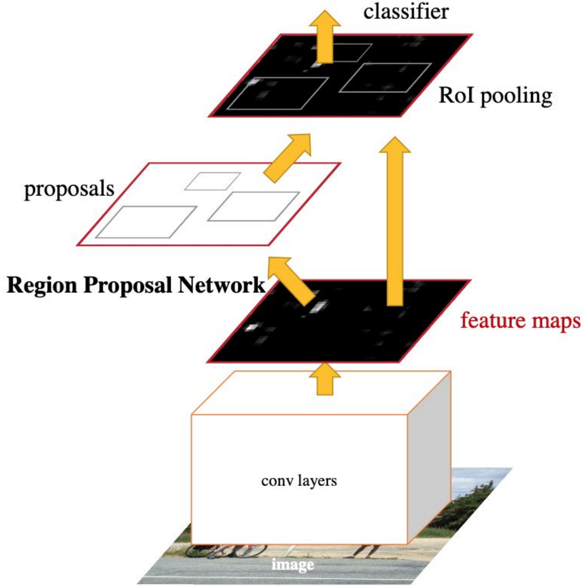
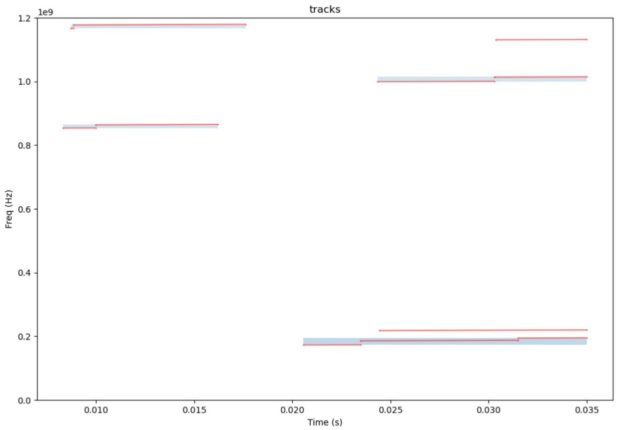


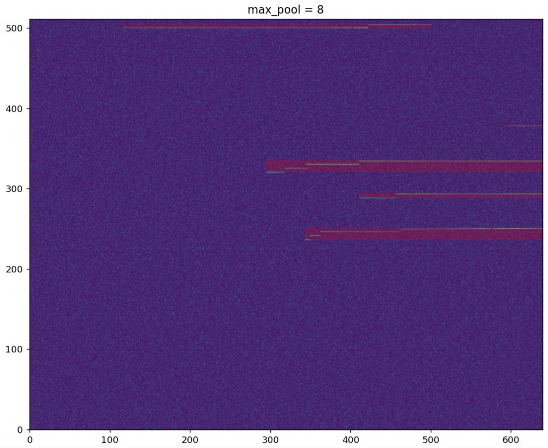
Figure 2: Faster R-CNN is a single, unified network for object detection. The RPN module serves as the 'attention' of this unified network.

Can a Faster R-CNN model be used to identify CRES events?

Simulated Dataset

- Based on the simulation framework created by Drew Byron at UW
- Edited to the task of generating ground-truth boundary boxes for simulated events
- Track information converted into a spectrogram image
- Total set comprised of 1000 spectrograms (~limit of free tier AWS size) of .035s in length with an average of 3 events per spectrogram





PyTorch-Lightning Implementation

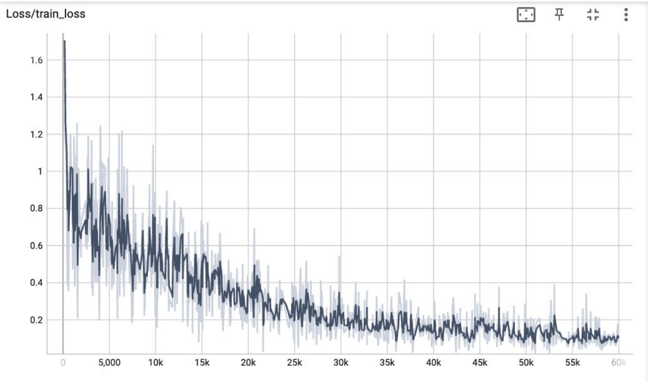
Workflow:

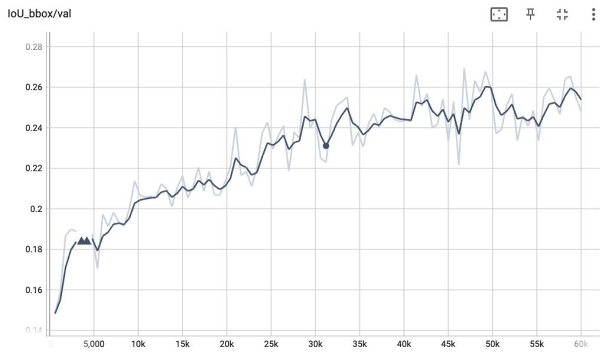
Dataset class loads the data and transforms it to form expected by the built-in Faster R-CNN model

LightningDataModule class calls the Dataset and handles the train/val/test splits, batch collation, and shuffling of data

LightningModule class configures the model and handles the training/validation steps (used a pretrained model to build off of)

Training object takes in the latter two for training of the model and logging of parameters of choice as it goes



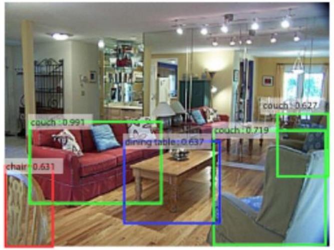
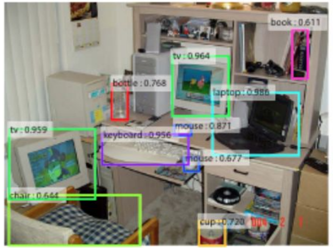
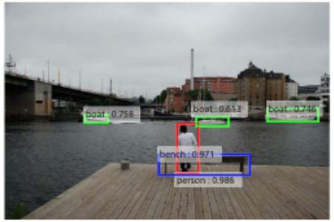


It converged!

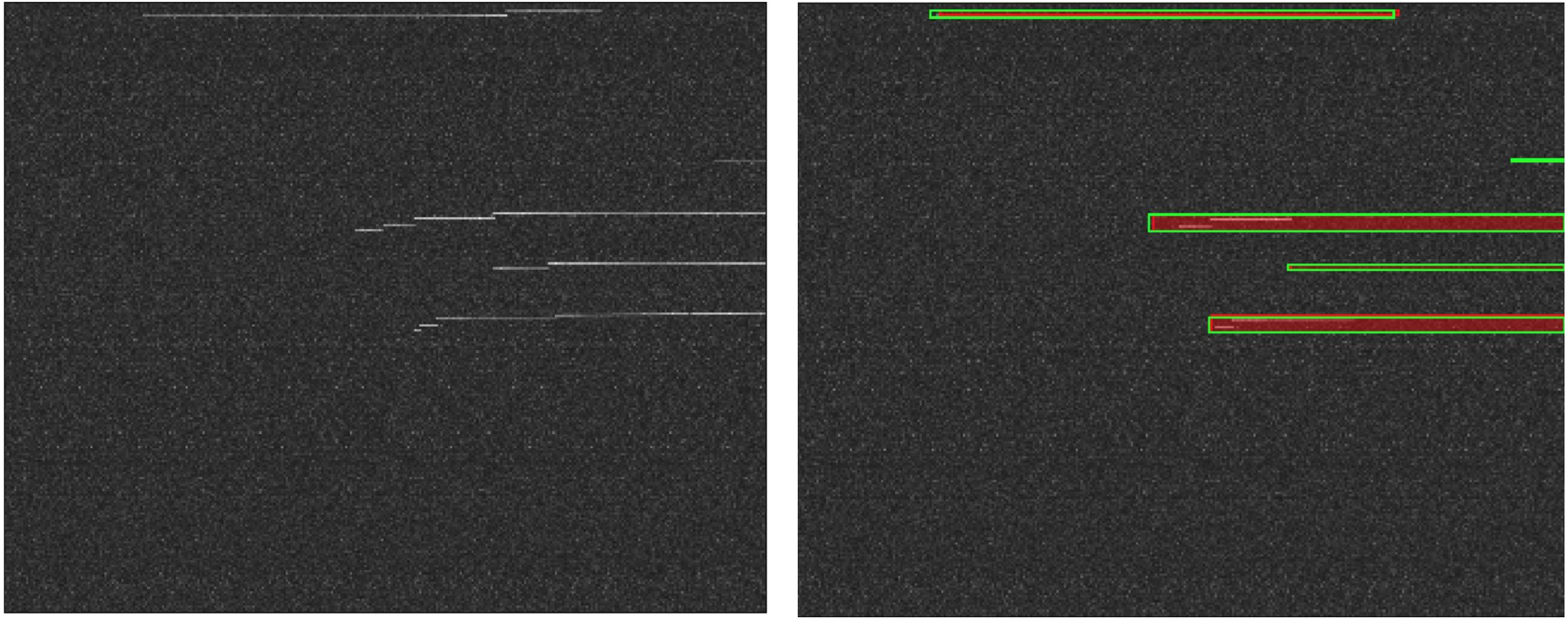
The model

- Built-in Faster R-CNN model with [ResNet50 Backbone]
- Began with pre-trained weights from the [COCO] dataset for fine-tuning and faster convergence
- Adam descent algorithm with learning rate = 1e-4
- 100 epoch training period
- Max-pool factor 16 applied



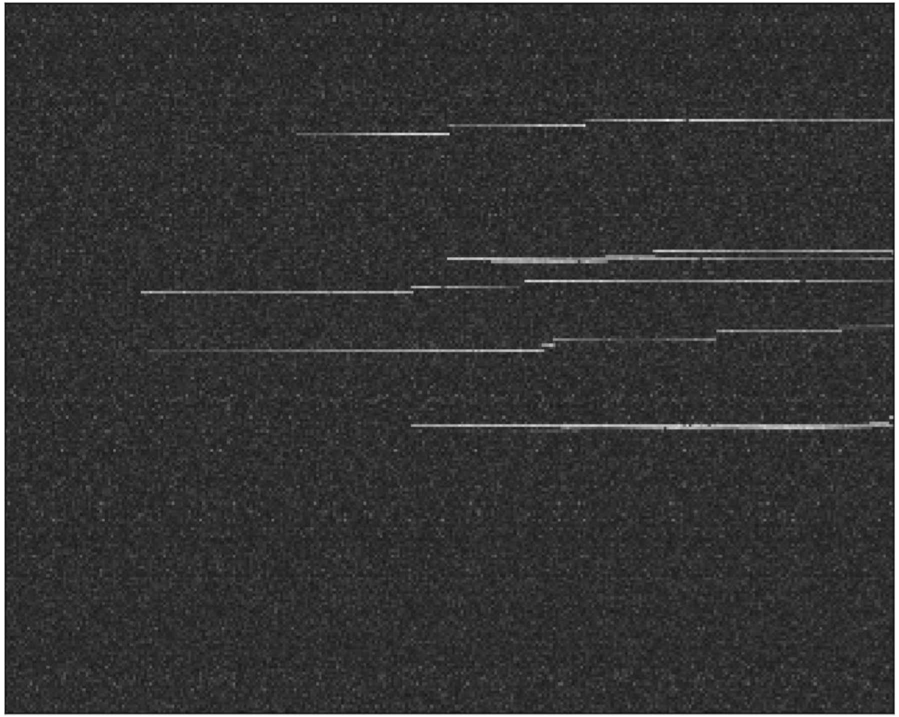





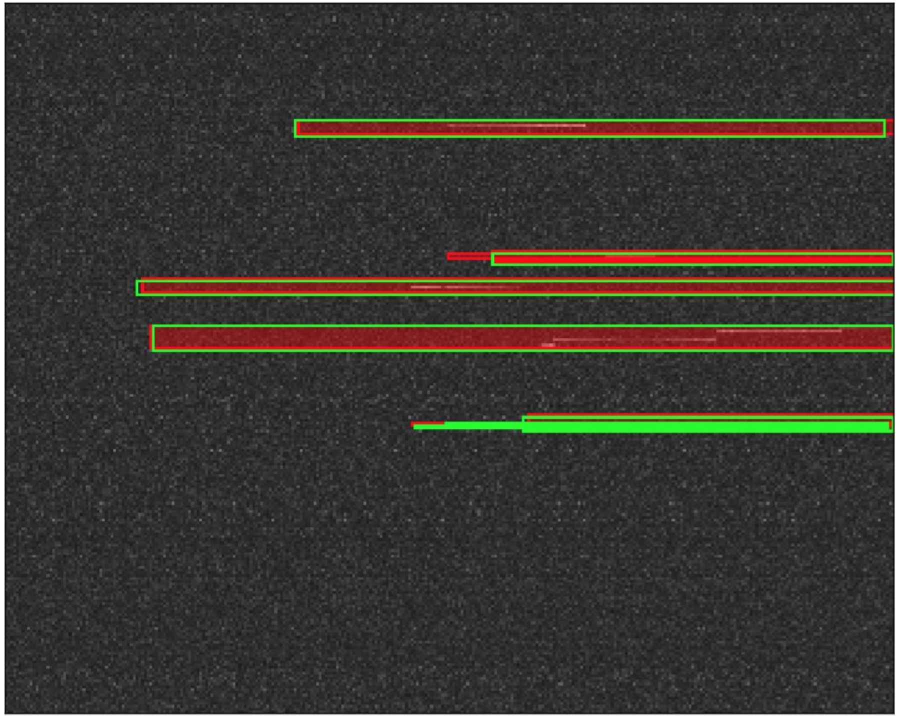
It’s working!



Red = target, green = predicted

Mostly…

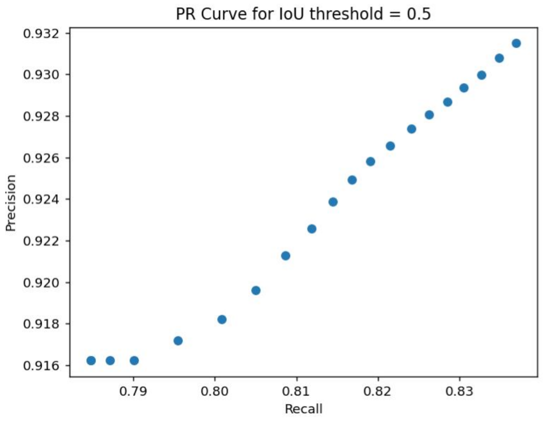


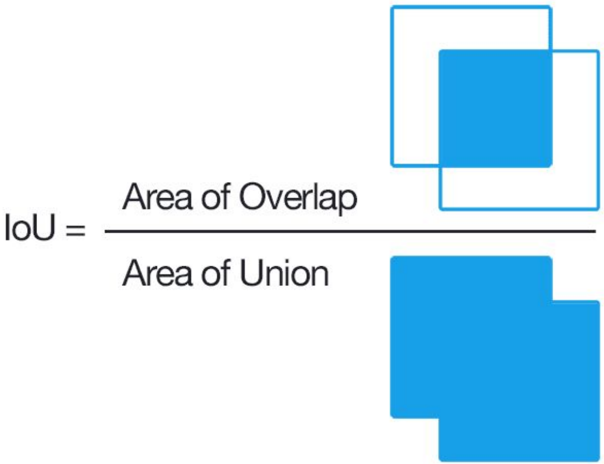


Thankfully, **event rate is a tunable parameter in the experiment**

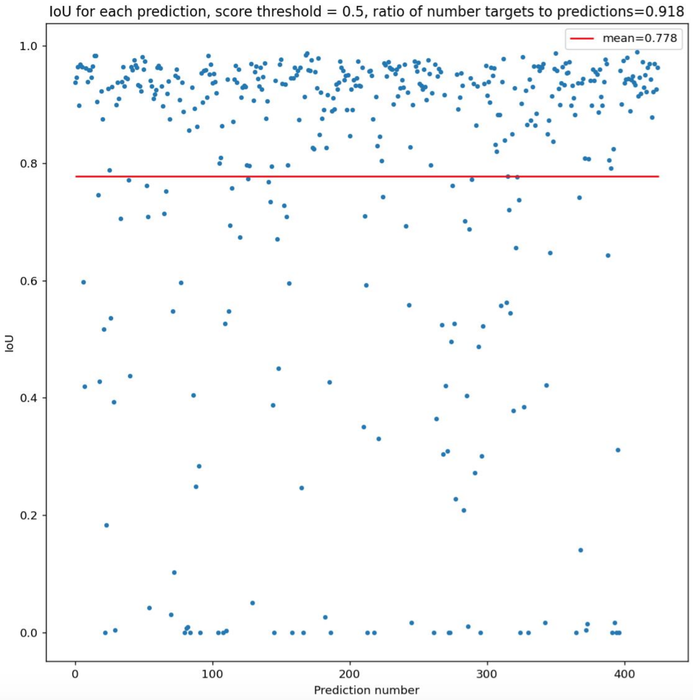
Evaluating Model Performance

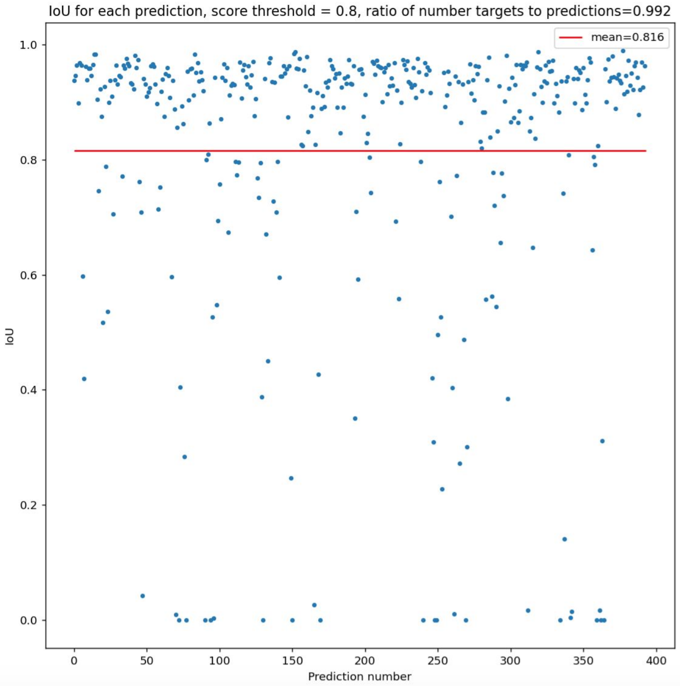
- Ideal metric for object detection models is mean average precision (mAP)
- Requires calculating area under Precision vs. Recall Curve
- Could not get full range in recall, possibly due to limited size of test dataset, possibly due to my own error
- Instead looked at Intersection over Union (IoU) and fraction of images with correct number of predictions





Evaluating Model Performance

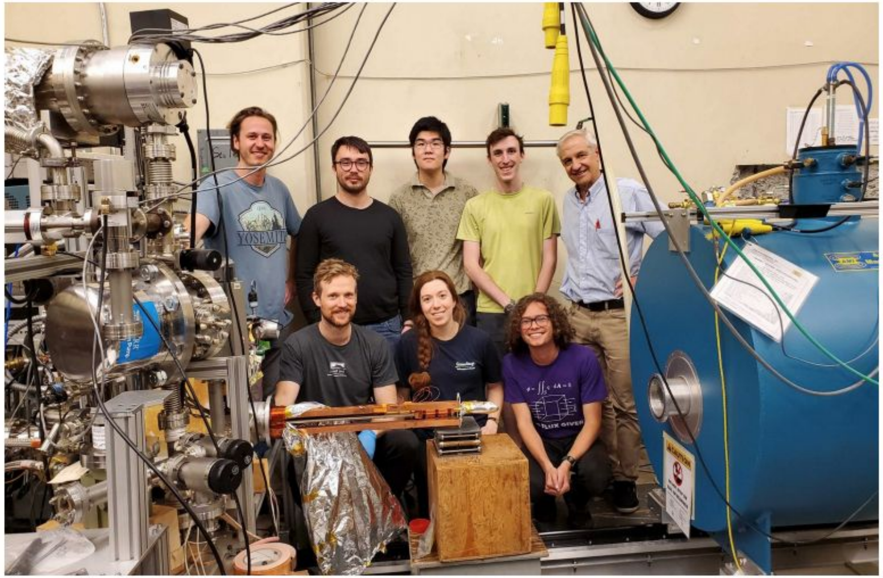




Summary and Conclusion

- Created a simulation and modeling pipeline for CRES event reconstruction with a Faster R-CNN model
- Initial performance shows that object detection models could become a valuable tool for CRES analysis
- Next steps would include:
  - Larger simulation set
  - Establish a better method of measuring model performance
  - Hyperparameter tuning
  - More physical simulation
  - Build beta spectrum from output of model
- **In the future can use this work as basis to build out a Mask R-CNN instance segmentation model**

Acknowledgments

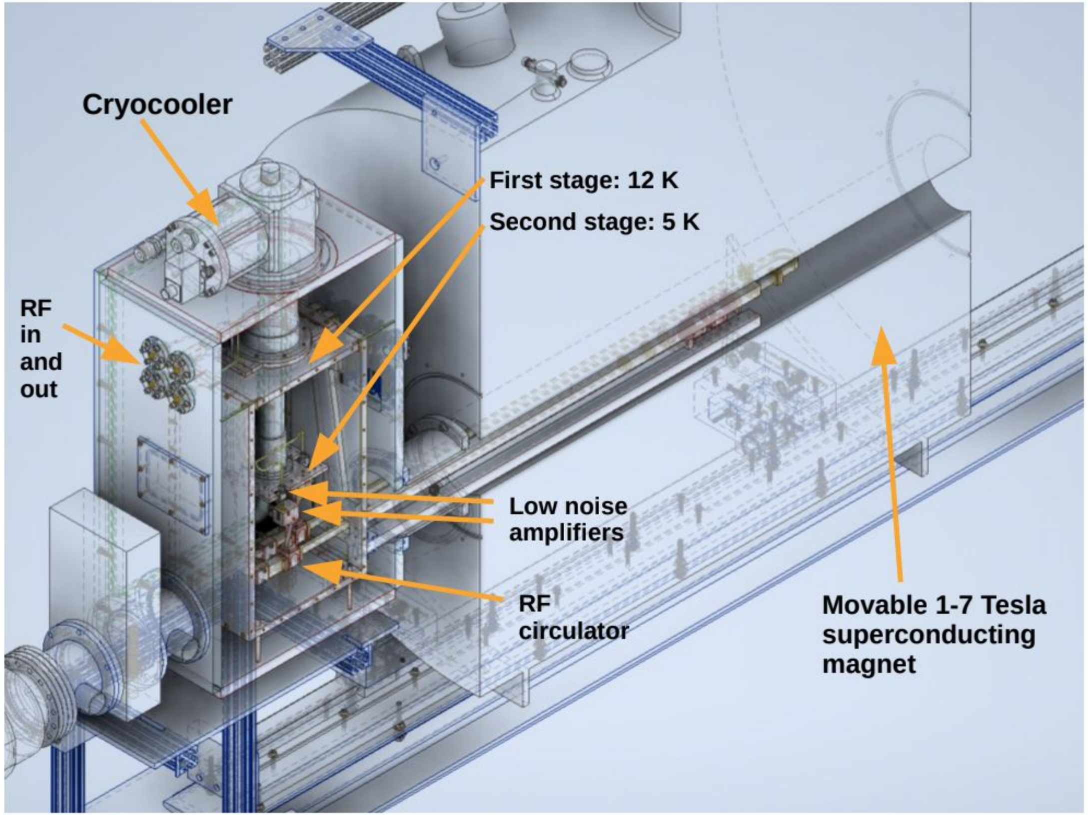


N. Buzinsky, W. Byron, W. DeGraw, B. Dodson, M. Fertl, A. Garcia, G. Garvey, B. Graner, M. Guigue, H. Harrington, L. Hayen, X. Huyan, K.S. Khaw, K. Knutsen, D. McClain, D. Melconian, P. Muller, E. Novitski, N. S. Oblath, R. G. H. Robertson, G. Rybka, G. Savard, E. Smith, D.D. Stancil, M. Sternberg, D. W. Storm, H. E. Swanson, R. J. Taylor, J. R. Tedeschi, B. A. VanDevender, F. E. Wietfeldt, A. R. Young, and X. Zhu.

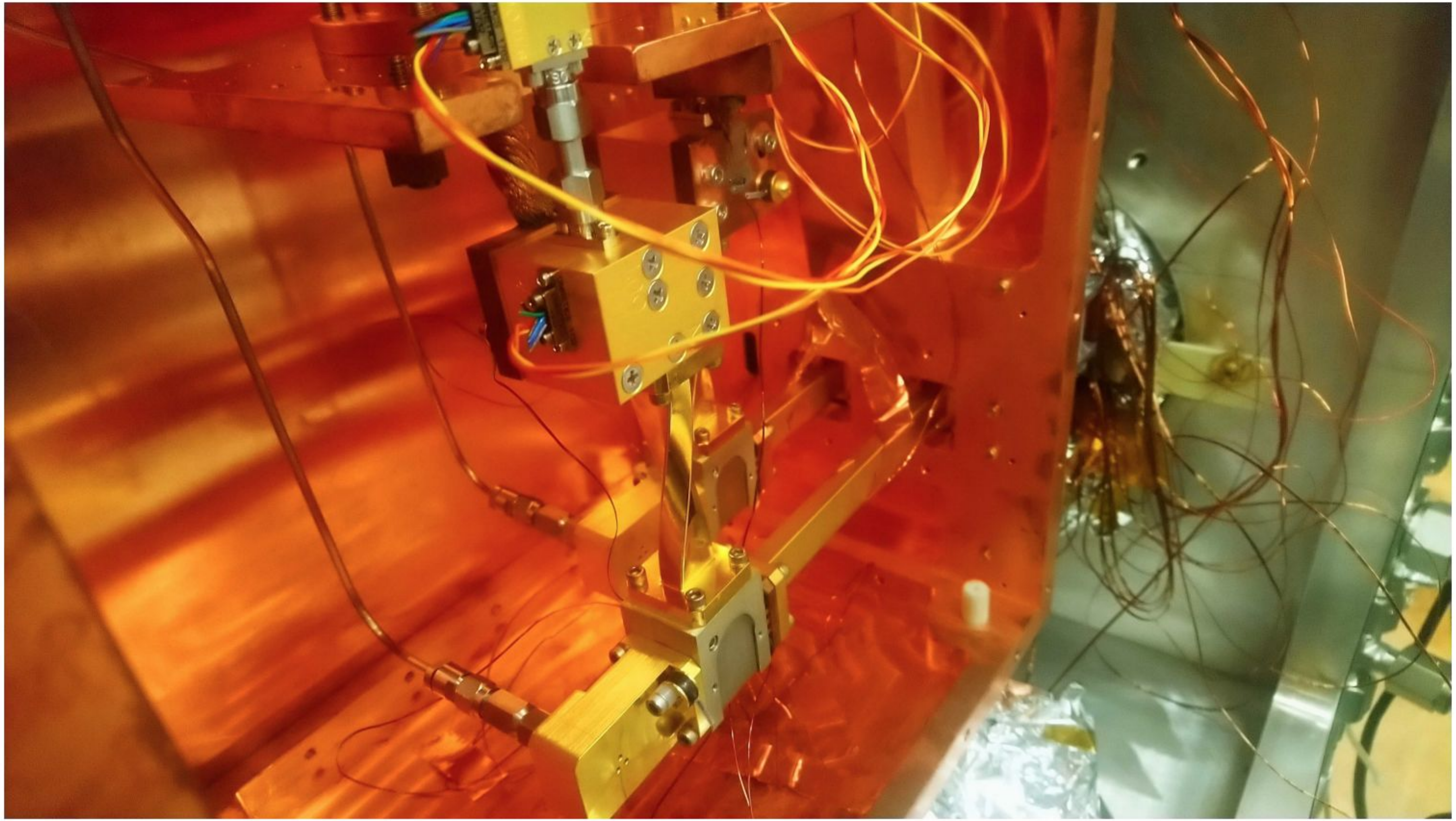
(The He6-CRES Collaboration)

Backup slides
(show and tell)

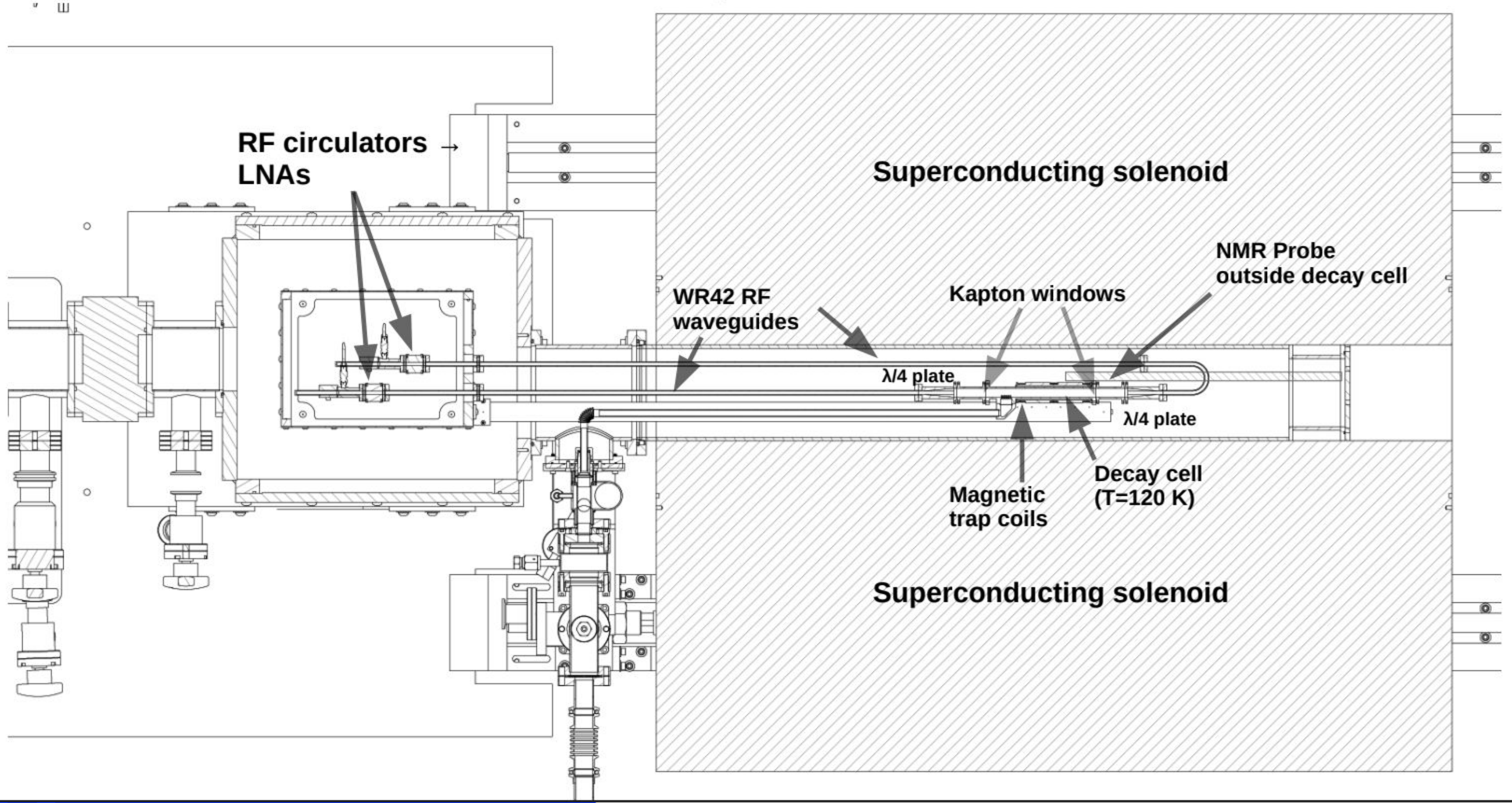
The ^{6}He-CRES experiment design



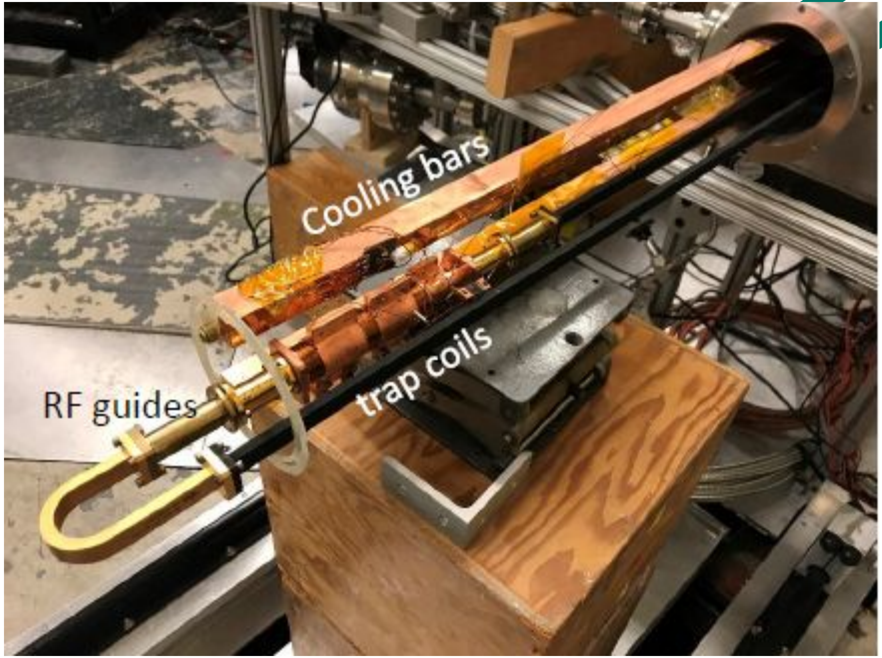
^{6}He CRES: Apparatus



RF System



RF circulators →
LNAs
Superconducting solenoid
NMR Probe
outside decay cell
WR42 RF
waveguides
Kapton windows
λ/4 plate
λ/4 plate
Decay cell
(T=120 K)
Magnetic
trap coils
Superconducting solenoid


Cooling bars
trap coils
RF guides

Katydid Analysis Flowchart