Novel Data-Processing Techniques for Signal Extraction in Project 8

by

Theodora C. Kunicki

Submitted to the Department of Physics
in partial fulfillment of the requirements for the degree of
Bachelor of Science in Physics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2016

© Theodora C. Kunicki, MMXVI. All rights reserved.

The author hereby grants to MIT permission to reproduce and to distribute publicly paper and electronic copies of this thesis document in whole or in part in any medium now known or hereafter created.

Author.

Department of Physics
May 26, 2016

Certified by...

Joseph Formaggio
Associate Professor of Physics
Thesis Supervisor

Accepted by...

Nergis Mavalvala
Associate Head
Novel Data-Processing Techniques for Signal Extraction in
Project 8
by
Theodora C. Kunicki

Submitted to the Department of Physics
on May 26, 2016, in partial fulfillment of the
requirements for the degree of
Bachelor of Science in Physics

Abstract

Project 8 presents a new modality of electron spectroscopy with the potential to exceed the resolution of the most precise electron spectrometers in operation today, potentially at much lower cost. Project 8, being a novel method, has different computational demands from existing experiments. This thesis explores the use of the Hough Transform as a tool in data processing in Project 8 and discusses its utility and function generally.

Thesis Supervisor: Joseph Formaggio
Title: Associate Professor of Physics
Acknowledgments

There is not space enough to thank all who need thanking, and I’m sure I’ll forget a few.

I am thankful, foremost, to my parents, for their constant love and support. Thank you so much, Prof. Formaggio, for your patience and understanding, and for teaching me so much during my time here. I also wish to extend thanks to the Department of Physics, and in particular to Cathy Modica, for helping me along as I crawled these last few feet to the finish line.

Thank you Dan and Xiaolu, for always listening. Thank you James for always being in my corner. And finally, a sincere thanks to my dear friends Abby, Charles, and Lucy. I would not be here but for your pep-talks, rah-rahs, and 加油’s.
Contents

1 Background and Motivation .................................. 9
  1.1 Overview ................................................. 9
  1.2 Pauli's Letter and Fermi's Paper ......................... 9
  1.3 A Brief Discussion of $\beta$-decay ..................... 11
  1.4 The Solar Neutrino Problem and the Necessity of Neutrino Mass 12
    1.4.1 The Homestake experiment and the solar neutrino problem 13
    1.4.2 Flavor mixing requires a massive neutrino ............ 13
    1.4.3 Neutrino oscillation is experimentally confirmed ....... 16
  1.5 A return to $\beta$-decay spectroscopy .................. 16

2 CRES technique .............................................. 19
  2.1 Overview ................................................ 19
  2.2 Relativistic Cyclotron Radiation ........................ 19
  2.3 The Plausibility of CRES for Tritium Spectroscopy ......... 22
    2.3.1 detectability of the signal; signal-to-noise ratio .... 24
    2.3.2 Resolution ............................................. 24
  2.4 A Brief Description of the Project 8 Hardware ............ 25
    2.4.1 Currently using $^{83m}$Kr .......................... 25
    2.4.2 magnet ................................................ 26
    2.4.3 sample cell and waveguide ........................... 26
    2.4.4 detection and signal processing ....................... 27
  2.5 Signal Characteristics ................................... 27
### 3 Hough Transform

3.1 Overview ........................................ 29
3.2 Hough Transform motivation .................. 30
3.3 How a Hough Transform works ................. 32

### 4 The Data Processing Chain in Project 8

4.1 Overview ........................................ 35
4.2 The Expected Characteristics of a Project 8 Signal .......................... 35
4.3 Data Collection ................................. 36
4.4 Data Processing in Katydid .................... 37
   4.4.1 A spectrogram is made and refined. .................. 37
   4.4.2 Tracks are located using DBSCAN ................... 38
   4.4.3 Tracks are characterized by a Hough transform, and stitched together into complete events. .................. 38

### 5 Conclusion

5.1 Results, Current Capabilities .................. 39
5.2 Future Prospects ............................... 39

### A appendix: code

41
Chapter 1

Background and Motivation

1.1 Overview

Early twentieth century observations of the $\beta$-decay spectrum provided the first hints to the neutrino's existence; initially introduced as a solution to the continuous energy spectrum observed in $\beta$-decay electrons, neutrinos were long assumed to be exactly massless. The fairly recent observation of neutrino flavor oscillation, however, has confirmed that neutrinos possess some small degree of mass.

The most straightforward method for experimentally determining neutrino mass is a return to $\beta$-decay spectroscopy. The $\beta$-decay spectrum, whose gross features first alerted us to the neutrino's presence, has been understood since Fermi's original work on the subject to be finely sensitive to neutrino mass. To observe this effect on shape of the $\beta$-decay spectrum, however, we require higher-resolution spectroscopy than has yet been available. This is the goal of Project 8, which proposes the use of a new modality of electron spectroscopy for high-resolution measurement of the tritium $\beta$-decay spectrum, with the goal of thereby measuring neutrino mass.

1.2 Pauli's Letter and Fermi's Paper

In 1914, James Chadwick made the first precise measurement of the spectrum of $\beta$ rays [3]. At the time, there was a general agreement that the energy states of atomic
nuclei should be quantized, and so any radiation produced in nuclear transitions was expected to be discrete. Indeed, α- and γ-decay produced monoenergetic spectra, but the β spectrum, per Chadwick’s experiment, was continuous. β-decay, then, could not be exactly analogous to the other two other types of radiation, though it was unclear how, at first.

One appealing possibility was that Chadwick’s experiment had been poorly designed [17]. Perhaps β particles had been allowed to scatter before their detection, so that the β rays had only been measured once they had lost some of their energy to their environment. In this manner, β rays could all be emitted at a single energy line just like α and γ rays, but due to their already well-known propensity to interact with intervening matter, still have produced the broad spectrum that Chadwick had observed. This explanation was disproved by Ellis and Wooster in 1927, who performed precision calorimetry of a β source of known activity, and showed that the average energy per β particle emitted by the source was only a fraction of their highest observed energy [8]. If the particles had in fact been produced at a single characteristic energy and then subsequently lost some of that energy to the environment, then we should expect the total heat given off by the sample to correspond to an average energy per particle equaling the highest observed energy in the β spectrum, but this was starkly not the case.

Another proposed explanation for the continuous β spectrum was that energy simply wasn’t conserved in individual nuclear decays, being instead a statistical law [13]. This explanation had clear aesthetic problems, but at the frontier of physics in the early 20th century, with sub-atomic structure and the physics of such scales under examination for the first time, it was worth considering.

In 1930, Wolfgang Pauli wrote his famous “radioactive letter,” postulating the existence of what he called a “neutron,” an electrically neutral spin 1/2 particle which he proposed was emitted in β-decay alongside the electron [20]. This particle could conceivably carry away the energy which seemed to be “missing” in β-decay, but it would be difficult to observe due to its lack of electromagnetic interaction. In addition to solving the problem of energy conservation in β-decay, the proposal also solved a
problem with the observed spin of certain atomic nuclei.

Nuclei, at the time, were thought to be composed of protons bound to “nuclear electrons,” electrons and protons being the only sub-atomic particles yet observed [17]. As an example, the nucleus of nitrogen-14 in this scheme would comprise 14 protons and 7 electrons, an odd number of spin 1/2 particles, and thus would be expected to have a half-integer spin. However, the nucleus of nitrogen-14 and other analogous nuclei had already been measured to have integer spin. Pauli’s addition of another spin 1/2 particle per electron in the nucleus did solve this nuclear spin problem, although further issues with this nuclear model remained until the discovery of the particle today called the neutron by none other than James Chadwick in 1932 [4].

Enrico Fermi ran with Pauli’s idea, and in 1934 published a famous paper outlining his theory of $\beta$-decay, which accurately predicted the shape of the continuous $\beta$-decay spectrum [9]. Since the name “neutron” had been taken in the intervening four years by Chadwick’s particle, Fermi playfully added an Italian diminutive to the word, giving the neutrino its modern name. Fermi’s paper was a triumph of early quantum field theory, and laid the groundwork for the ensuing decades of progress in particle physics theory.

1.3 A Brief Discussion of $\beta$-decay

Fermi’s paper was a treatment of $\beta$-decay, the process by which a neutron decays into a proton, an electron (also known in this context as a $\beta^-$ particle), and an electron antineutrino.\footnote{Fermi had no reason to expect the existence of more than one type of neutrino, and called this particle simply $\nu$.}

$$ n \rightarrow p^+ + e^- + \bar{\nu}_e $$

Free neutrons do undergo $\beta$-decay, but it’s easier to observe the context of atomic nuclei. For instance, tritium $^3\text{H}$ decays into the more stable nucleus $^3\text{He}$:

$$ ^3\text{H} \rightarrow ^3\text{He} + e^- + \bar{\nu}_e $$

\footnote{Fermi had no reason to expect the existence of more than one type of neutrino, and called this particle simply $\nu$.}
When a nucleus undergoes β-decay, it expels an electron and an electron antineutrino, and increases in atomic number while its atomic mass remains constant; equivalently, we can think of one of its neutrons as being transformed to a proton as it expels an electron and an electron antineutrino.

In his paper, Fermi notes that according to his theory, the shape of the endpoint of the β-decay spectrum should be sensitive to the mass of the neutrino. Fermi, observing that all measurements of the β-decay spectrum were consistent with a massless neutrino, took $m_\nu = 0$ in his paper.

The formula describing the shape of β-decay spectrum is

$$\frac{dN}{dE} \propto F(Z, E) \cdot p \cdot E \cdot (E_0 - E) \sum_{i=1}^{3} |U_{ei}|^2 \sqrt{(E_0 - E)^2 - m_i^2} \cdot \Theta(E_0 - E - m_i) \quad (1.3)$$

Where $N$ is the number of electrons emitted at a particular energy $E$, $p$ is the electrons' momentum, and $E_0$ is the total difference in energy between the initial and final states of the atomic nucleus. $\Theta$ is the unit step function, and $F(Z, E)$ is the Fermi function in terms of the daughter nucleus's atomic number and the electron's total energy, roughly accounting for the effect of electromagnetic attraction between the daughter nucleus and the emitted electron. We sum over the mass eigenstates of neutrinos, of which there are three.$^2$

If neutrinos are massive, we expect that the shape of the β-decay spectrum will be affected. Electron spectroscopy has improved since Fermi's day, but all studies of β-decay spectra since his time have remained consistent with a massless neutrino.

### 1.4 The Solar Neutrino Problem and the Necessity of Neutrino Mass

In the absence of experimental evidence to the contrary, neutrinos were assumed to be massless for most of the 20th century. This changed when free neutrinos were confirmed to oscillate between flavors, a phenomenon which is only possible for massive

$^2$See subsection 1.4.2
1.4.1 The Homestake experiment and the solar neutrino problem

Neutrino oscillation provided the solution to the longstanding solar neutrino problem: the persistent mismatch between the predicted and observed number of solar neutrinos detected at the surface of the Earth. The sun's main fusion processes produce a steady output of electron neutrinos, but the observed flux of these neutrinos on Earth is only a fraction of what would be expected, between one third and one half of what is predicted by solar models [5]. This deficit in solar neutrinos was first observed at the Homestake mine experiment in the late 1960s; the result was published by Davis et al. in 1968 [6]. The possibility that neutrinos change flavor as they travel from the sun to the Earth was one of many proposed solutions to the solar neutrino problem.

Neutrino observatories like Homestake count neutrinos via inverse $\beta$-decay and are only sensitive to electron neutrinos. The lower than expected neutrino flux observed at Homestake would be explained if solar neutrinos were no longer purely electron neutrinos when they intercepted the detector [12]. The main argument against the idea was that neutrino oscillation requires that neutrinos possess mass, and there was no other experimental evidence to suggest neutrinos were anything other than massless.

1.4.2 Flavor mixing requires a massive neutrino

We have so far only alluded to the existence of three neutrino mass states $\{\nu_1, \nu_2, \nu_3\}$ and three flavor states $\{\nu_e, \nu_\mu, \nu_\tau\}$. It is vital to note that these states do not directly correspond to one-another, a fact which directly leads to the phenomenon of free neutrinos changing in flavor as they travel.

Flavor states are eigenstates of the weak interaction: the only processes which can create or destroy neutrinos. Neutrinos are always produced in pure flavor eigenstates. Likewise, each of the mass eigenstates describes a particle of definite mass: eigenstates
of the free particle Hamiltonian. Both sets of states form orthogonal bases for the space of possible neutrinos\(^3\), related to each other by a unitary mixing matrix\(^4\).

\[
\begin{bmatrix}
\nu_e \\
\nu_\mu \\
\nu_\tau
\end{bmatrix} =
\begin{bmatrix}
U_{e1} & U_{e2} & U_{e3} \\
U_{\mu1} & U_{\mu2} & U_{\mu3} \\
U_{\tau1} & U_{\tau2} & U_{\tau3}
\end{bmatrix}
\begin{bmatrix}
\nu_1 \\
\nu_2 \\
\nu_3
\end{bmatrix}
\tag{1.4}
\]

The mass states are eigenfunctions of the free particle Hamiltonian, and so develop independently of each other according to the Schrödinger equation as the particle propagates through the vacuum. The independent development of each mass state causes the particle’s wavefunction to “slosh” between the three flavor states, its probability of being measured as any given flavor varying sinusoidally as time passes.

As an illustration, let us simplify to a case where there are two lepton flavors and two neutrino mass states. \( U \) is a unitary matrix, so we may call it a rotation by an angle \( \theta \). In two dimensions, there is a single mixing angle. In reality, with a three-dimensional Hilbert space, there must be three angles to describe a rotation, hence our simplification to two dimensions.

\[
U = \begin{bmatrix}
\cos \theta & -\sin \theta \\
\sin \theta & \cos \theta
\end{bmatrix}
\tag{1.5}
\]

Consider a neutrino \( |\nu\rangle \) which is created at \( t = 0 \) as a pure electron neutrino

\[
\langle \nu_e | \nu(t = 0) \rangle = 1
\tag{1.6}
\]

This state can be expressed in the mass basis as

\[
|\nu(t = 0)\rangle = U_{e1} |\nu_1\rangle + U_{e2} |\nu_2\rangle
= \cos \theta |\nu_1\rangle + \sin \theta |\nu_2\rangle
\tag{1.7}
\]

Each mass state \( |\nu_j\rangle \) is an eigenstate of the free particle Hamiltonian, and evolves in

---

\(^3\)barring sterile neutrino states

\(^4\)the Pontecorvo-Maki-Nakagawa-Sakata matrix, which made an appearance in equation (1.3)
time as
\begin{equation}
|\nu_j(t)\rangle = e^{-iE_j t/\hbar} |\nu_j\rangle \tag{1.8}
\end{equation}

Taken together, this means our particle will develop in time as
\begin{equation}
|\nu(t)\rangle = e^{-iE_1 t/\hbar} \cos \theta |\nu_1\rangle - e^{-iE_2 t/\hbar} \sin \theta |\nu_2\rangle \tag{1.9}
\end{equation}

Using \( \langle \nu_\mu \rangle = \langle \nu_j \rangle U^\dagger = -\sin \theta \langle \nu_1 \rangle + \cos \theta \langle \nu_2 \rangle \), we can calculate the particle’s probability of being measured as a \( \mu \) neutrino:
\begin{equation}
|\langle \nu_\mu | \nu(t) \rangle|^2 = (\sin \theta \cos \theta)^2 (e^{-iE_2 t/\hbar} - e^{-iE_1 t/\hbar})(e^{iE_2 t/\hbar} - e^{iE_1 t/\hbar}) \tag{1.10}
\end{equation}

Which simplifies to:
\begin{equation}
|\langle \nu_\mu | \nu(t) \rangle|^2 = \left[ \sin 2\theta \sin \left( \frac{E_2 - E_1 t}{\hbar} \right) \right]^2 \tag{1.11}
\end{equation}

So a particle which was created as an electron neutrino may interact instead as a \( \mu \) neutrino, with a probability which depends sinusoidally on time and the difference in the energies between mass states.

These energies \( E_i \) are \( \gamma m_i c^2 \), so the rate of neutrino oscillation is sensitive to the difference in masses between the states, and neutrino oscillation cannot occur without this difference being nonzero.

If we take the neutrinos to be extremely relativistic, our particle’s probability of interacting as a \( \mu \) neutrino is
\begin{equation}
|\langle \nu_\mu | \nu(t) \rangle|^2 = \left[ \sin 2\theta \sin \left( \frac{(m_2^2 - m_1^2)c^3}{4\hbar E} t \right) \right]^2 \tag{1.12}
\end{equation}

Where \( E \) is the particle’s energy.
1.4.3 Neutrino oscillation is experimentally confirmed

In 2001 and 2002, results from Super-Kamiokande and the Sudbury Neutrino Observatory experimentally confirmed that solar neutrinos undergo oscillation [11][1]. Both of these experiments were sensitive to all three flavors of neutrinos, but with differing cross sections, so the total neutrino flux could be measured and compared to the electron neutrino flux. When all three flavors were accounted for, the total neutrino flux agreed with what was expected from solar models, finally solving the solar neutrino problem, and confirming that neutrinos must be massive [14]. Measurements of individual mixing angles are ongoing, and our numerical representation of $U$ is getting more accurate by the day.

Massive neutrinos solved the solar neutrino problem, but pose a problem of their own. In order to determine the overall effect of massive neutrinos, we must know their absolute mass, but observations of oscillation are only sensitive to the magnitude of mass differences. Neutrinos possess the smallest masses of any known massive particle, and an accurate measurement would patch up one of the last holes in the Standard Model.

1.5 A return to $\beta$-decay spectroscopy

Observations of neutrino oscillation confirm that three distinct mass states exist, but observations of flavor mixing cannot determine the absolute masses of these states, only the difference in the squares of their masses.

Since $\beta$-decay spectra are known to be sensitive to neutrino mass, one way to solve this problem is to increase the resolution of $\beta$-decay spectroscopy until the effects of neutrino mass are manifest. By comparing the observed spectrum with the theoretical $m = 0$ spectrum, the expected value of the electron neutrino’s mass can be inferred.

Currently existing neutrino mass experiments are hoping to push the resolution of the neutrino mass measurement down to a few tenths of an eV, but current lower limits are an order of magnitude lower [2]. Project 8 represents a new technique for performing $\beta$-decay spectroscopy which could potentially match those limits in
resolution, but with an entirely different modality utilizing the observation of relativistic cyclotron radiation. The spectrometer promises to be much smaller than existing technology, and is currently in a prototype stage. The remainder of this thesis treats Project 8 and computational techniques in performing high-resolution electron spectroscopy using the detection of relativistic cyclotron radiation.
Chapter 2

CRES technique

2.1 Overview

This chapter will introduce Cyclotron Radiation Emission Spectroscopy as a technique for high-precision measurement of the $\beta$-decay spectrum. Pioneered by the Project 8 collaboration and also known by the acronym CRES, this technique uses the fact that cyclotron radiation emitted by a relativistic particle is dependent on its total kinetic energy. By capturing $\beta$-decay electrons in a strong magnetic field and observing their cyclotron radiation, we can, in principle, non-destructively measure their individual kinetic energies. The spectrum may then be reconstructed from these individual particle measurements.

After discussing the theory behind CRES, we will establish the technique's plausibility in the case of tritium $\beta$-decay, and briefly discuss the hardware of the Project 8 prototype spectrometer, currently operating at the University of Washington. Finally, there will be a brief discussion of the characteristics of an electron's signal, the detection of which is central to the following chapter.

2.2 Relativistic Cyclotron Radiation

In the presence of a transverse magnetic field, the trajectory of a charged particle is curved by the resulting magnetic force. In a sufficiently strong, uniform field, the
particle's motion in the plane perpendicular to that field becomes circular, resulting in
cyclotron motion. As would be the case with any accelerating trajectory, this charge
will emit radiation. The classical treatment of cyclotron radiation follows, and yields
the result that the frequency of cyclotron radiation is independent of the particle's
kinetic energy, depending only on its charge, mass, and the strength of the magnetic
field involved.

An electron (charge $-e$, mass $m_e$) moving with velocity $\vec{v}$ in a magnetic field $\vec{B}$
is subject to a magnetic force of

$$\vec{F}_B = -e\vec{v} \times \vec{B}$$  \hspace{1cm} (2.1)

Without loss of generality, we can define a cylindrical coordinate system $(r, \theta, z)$ where
$\vec{B} = B\hat{z}$ and $\vec{v} = v_\perp \hat{\theta} + v_\parallel \hat{z}$, so that equation (2.1) reduces to

$$\vec{F}_B = -ev_\perp B\hat{r}$$  \hspace{1cm} (2.2)

This kind of radial force produces circular motion:

$$\frac{m_e v_\perp^2}{r} = ev_\perp B$$  \hspace{1cm} (2.3)

Which we can solve for the angular frequency of nonrelativistic cyclotron motion

$$\omega_c = \frac{v_\perp}{r} = \frac{eB}{m_e}$$  \hspace{1cm} (2.4)

as long as $v_\perp \neq 0$. Equivalently, the classical frequency in cycles per second is

$$f_c = \frac{eB}{2\pi m_e}$$  \hspace{1cm} (2.5)

Notably, $f_c$ is independent of the particle's velocity; even as the electron loses energy
through radiation, its classical cyclotron frequency remains constant. The frequency
of cyclotron radiation, being identical to the frequency of circular motion, must also
be independent of the particle's energy.
While this is a beautiful result of classical electromagnetism, it's brought us no closer to our aim as stated in chapter 1. Our goal is to perform spectroscopy of tritium $\beta$-decay electrons, but since $f_c$ is independent of its kinetic energy, observing cyclotron radiation in the classical domain would be useless. As a consequence of relativity, however, the cyclotron radiation emitted by relativistic electrons does reflect their kinetic energy.

At relativistic energies, a particle with rest mass $m$ has a relativistic mass in the lab frame of $\gamma m$, where $\gamma$ is the Lorentz factor, $\gamma = [1 - (\frac{v}{c})^2]^{-1}$. Indeed, cyclotron motion of a relativistic electron is analogous to the classical case apart from the shift in the particle's apparent mass, so we can repeat the above derivation substituting $\gamma m_e$ for each instance of $m_e$ and arrive at a result in which frequency is a function of $\gamma$, and thus kinetic energy.

Since cyclotron frequency is inversely proportional to mass, it follows that $f_c$ will be multiplied by $1/\gamma$ to yield the correct relativistic cyclotron frequency. The frequency of a relativistic electron’s cyclotron radiation is

$$f_{\text{rel}} = \frac{eB}{2\pi\gamma m_e} \quad (2.6)$$

The Lorentz factor $\gamma$ is a function of particle velocity alone, so the cyclotron frequency, utterly independent of particle energy in the classical regime, gains an energy dependence in the relativistic case. This means that if we have a relativistic charged particle undergoing cyclotron motion, we can measure its energy by observing its cyclotron radiation alone.

A free particle’s kinetic energy $K$ is all of its energy which is not contained in its rest mass, so the kinetic energy of a free relativistic electron is

$$K = (\gamma - 1)m_e c^2 \quad (2.7)$$

thus,

$$\gamma = 1 + \frac{K}{m_e c^2} \quad (2.8)$$
Substituting into equation (2.6), we have,

\[ f_{\text{rel}} = \frac{eB}{2\pi mc} \frac{1}{\gamma} \]

\[ = \frac{f_c}{K/m_e c^2 + 1} \quad (2.10) \]

By solving for K, we can now determine a relativistic electron’s kinetic energy solely by observing its cyclotron frequency,

\[ K = \left( \frac{f_c}{f_{\text{rel}}} - 1 \right) m_e c^2 \quad (2.11) \]

Vitally, \( f_{\text{rel}} \), like \( f_c \), is independent of the direction of the electron’s motion. The only element of \( f_{\text{rel}} \) which is sensitive to the particle’s energy (i.e., \( \gamma \)) is agnostic to the direction of the particles velocity, depending only on its magnitude as measured relative to the lab frame. Any electron of velocity \( \vec{v} \) for which \( \vec{v} \cdot \vec{B} \neq 0 \) will emit radiation at the same frequency \( f_{\text{rel}} \).

In practical terms, this means that if we are able to observe the cyclotron radiation emitted by a single relativistic electron, we could make an entirely passive measurement of that individual particle’s energy. By measuring enough individual particle energies, we can reconstruct the spectrum of whatever was emitting those electrons in the first place, for instance, tritium. This is the backbone of the technique used by Project 8, with the goal of eventually measuring the tritium \( \beta \) spectrum with a high enough resolution to infer neutrino mass.

2.3 The Plausibility of CRES for Tritium Spectroscopy

This method for performing spectroscopy of relativistic electrons using cyclotron radiation is referred to as Cyclotron Radiation Emission Spectroscopy, or its acronym CRES. The question of whether CRES would be a viable technique for using \( \beta \)-decay spectroscopy to determine neutrino mass remains. We can imagine a few sticking points which could derail our use of CRES:
• perhaps the electrons emitted in tritium $\beta$-decay are not sufficiently relativistic for their shift in cyclotron frequency to be measurable, or

• the cyclotron radiation from an isolated electron is too faint to be observable above the level of noise, or

• maybe it is not possible to measure the frequencies at a high enough precision to sufficiently improve on current methods of high-energy electron spectroscopy.

Regarding the first point, we are at the mercy of Nature; in this case, she is generous. Those electrons with the highest kinetic energies will be most susceptible to relativistic effects, and the most energetic electrons emitted in the $\beta$-decay of tritium have an energy of around 18.6 keV, corresponding to $\gamma \approx 1.0364$ [2]. As noted in chapter 1, only the highest-energy region of the tritium spectrum is of interest if we want to see the effects of neutrino mass, so the sensitivity of our spectroscopy technique overlaps nicely with our object of study.

In a 1 Tesla magnetic field, an electron's classical cyclotron frequency $f_c$ is very close to 28 GHz, and the radiation emitted by an 18.6 keV electron will be shifted down to around 27 GHz by relativistic effects. These frequencies lie in the microwave K band and are detectable (and quite distinguishable) with commercially available equipment [17].

The second point is more a question of engineering and the feasibility of building a sensitive enough detector. We will show that a magnetic field of around 1T is viable, so long as the apparatus is built to minimize thermal noise. This is expounded in section 2.3.1.

Finally, we must consider the resolution of this technique. Projects such as KA-TRIN, using well-established spectroscopy techniques, are already working towards measuring the tritium $\beta$-decay spectrum on a scale of around 0.2 eV [18]. If CRES would not represent an improvement upon these techniques, then there is not much value in pursuing it as an alternative to them. In this regard, also, CRES theoretically succeeds, as shown in section 2.3.2.
2.3.1 detectability of the signal; signal-to-noise ratio

An electron moving through a magnetic field emits radiation at a characteristic frequency, and for sufficiently energetic electrons, this frequency will be shifted as a function of kinetic energy. For this to be useful in spectroscopy, this radiation must be detected. Said another way, in order for CRES to be effective, we must be able to detect the cyclotron radiation of individual electrons, an unprecedented feat [2].

Superconducting electromagnets of the sort used for NMR produce fields of a few Teslas in strength. We will show that a 1T magnet is more than sufficient for the production of detectable single-electron cyclotron radiation, as long as noise is carefully managed.

As alluded to above, a trapped electron moving in cyclotron motion will radiate away energy, the rate of which is given by the Larmor formula.

\[ P = \frac{2}{3} \frac{e^2 a^2}{\pi \epsilon_0 c^3} \]  

(2.12)

It can be shown that in the case of a relativistic electron in a magnetic field, this amounts to

\[ P(\gamma, \theta) = \frac{2}{3} \frac{e^4}{\pi \epsilon_0 m_e c^3} B^2 (\gamma^2 - 1) \sin^2 \theta \]  

(2.13)

where \( \theta \) is the electron’s angle of inclination relative to the direction of the magnetic field [2].

For a typical tritium \( \beta \)-decay electron around the endpoint energy of 18.6 keV, created in a 1 T field the power radiated will be on the order of 1 fW, which, though small, is eminently detectable with conventional RF equipment [17].

2.3.2 Resolution

From equations (2.11) and (2.6), we can see that our accuracy in measuring \( K \) will be dependent on the precision with which we know the strength of the magnetic field\(^1\) and our accuracy in measuring \( f_{rel} \).

\(^1\)bearing in mind our initial assumption that \( \bar{B} \) perfectly homogeneous
Assuming for the moment that our magnetic field strength is known exactly, we can derive the uncertainty of the measured kinetic energy $\sigma_K$ as a function of our uncertainty in the frequency measurement $\sigma_f$ [17]:

$$\sigma_K = \gamma m_e c^2 \frac{\sigma_f}{f_{rel}}$$

(2.14)

For an uncertainty of less that 1 eV in our measurement of $K$, this necessitates $\sigma_f/f_{rel} \approx 2 \times 10^{-6}$, at a frequency of 26 MHz, this corresponds to a resolution of around 50 kHz. Again, this is easily within the capabilities of modern RF equipment [17].

2.4 A Brief Description of the Project 8 Hardware

In order to perform CRES on an electron source, we need a way to confine that electron source in a strong uniform magnetic field, a way to detect cyclotron radiation, and a technique to process it. The Project 8 setup is not much more complex than this.

2.4.1 Currently using $^{83m}$Kr

Although the eventual goal of Project 8 is to use CRES to perform high-precision spectroscopy of the tritium $\beta$ spectrum, it is currently in a proof-of-concept phase, using a prototype detector at the University of Washington to measure the electron emission spectrum of $^{83m}$Kr, a metastable isotope of Krypton. $^{83m}$Kr is useful for testing because of its well-documented and characteristic electron lines at energies comparable to the energy of the endpoint of the tritium $\beta$ spectrum. The $^{83m}$Kr gas is, much like tritium gas, transparent to the cyclotron radiation of interest, making CRES possible. The remainder of this chapter will focus on the current configuration of the Project 8 prototype, which is focused on $^{83m}$Kr spectroscopy as a proof-of-concept for eventual tritium $\beta$-decay spectroscopy using CRES.
2.4.2 magnet

The limits on resolution as presented sections 2.3.2 and 2.3.1 is promising, there is a snag: in order to truly observe a signal of frequency $f$, we will need to observe it for a time comparable to $1/f$. The electrons under study are by necessity mildly relativistic, meaning that without some additional means of keeping the electrons confined, they will escape too quickly along the $z$-axis of the experiment. For instance, without an electron trap of some kind, an electron born with a kinetic energy of 18 keV, with a trajectory only $1^\circ$ inclined from the horizontal, will still have a $z$-direction velocity on the order of $10^6$ m/s, far too fast to be properly observed before escaping.

Thus, a magnetic trap must be added to the main 1T magnetic field. Although a magnetic trap will not affect an electron's total energy, it produces an inhomogeneity which complicates the equations of motion for a captured electron. This trap will not be discussed in depth in this thesis, although tweaks to the design of this component have the potential to greatly increase the resolution of the Project 8 experiment.

In addition to the primary uniform magnetic field, we must also have a way to trap the electrons in the high-field area long enough to observe them—without a mechanism to trap electrons inside the device, they would escape too quickly to be observed. The trap cannot utilize electric fields, since electric fields, unlike magnetic fields, do work on the particle, and would confuse the task of spectroscopy. For further details about the magnet, see chapter 3 of [17].

2.4.3 sample cell and waveguide

The sample of radioactive gas is confined to a small ($\approx 3$ cm$^3$) chamber [17]. Adjoined to the sample cell is a length of waveguide, whose propagating mode can transmit any microwave radiation from directly adjacent to the cell to the first round of amplifiers. This assembly is placed at the center of a superconducting magnet, which provides the primary magnetic field for the experiment. The smaller trap field is provided by a coil would directly around the cell itself.

The cell, waveguide, and the first round of amplification are all kept at a cryogenic
temperature to reduce noise. The cell is also kept in a vacuum, with a pressure of less than $5 \times 10^{-7}$ Pa [17]. This is necessary to keep a captured electron's mean free path long enough for it to be observed sufficiently long before it is scattered.

2.4.4 detection and signal processing

The signal, which lies in the band between 25 and 27 GHz, is amplified, and then mixed down to a frequency band centered on 1.8 GHz, more manageable with commercially available RF components. Once it's been mixed down, the signal is taken from a bandwidth of around 2 GHz to a smaller window of 120 MHz, by mixing another, variable frequency oscillator and applying a fine-tuned band-pass filter. This signal, narrowed and amplified, is then digitized by an 8 bit digitizer running at 250 MHz, and written to disk [17].

Alternatively, before mixing down, the signal may be passed through a real-time spectral analyzer, which allows for triggering, discussed in chapter 4.

The latter technique has been used in all but the earliest runs, as it greatly simplifies data processing.

2.5 Signal Characteristics

Data analysis for Project 8 is a massive undertaking. In principle, our task is quite simple—once the RF signal has been digitized, we want to search through it for the telltale signal of the cyclotron radiation of individual electrons. However, because early runs had the digitizer continuously working for the length of the run, this can mean combing through terabytes of collected data searching through distinguishable, but truly random events.

More recent runs have made use of triggering by way of a real-time spectral analyzer [10]. Even if data recording is mostly localized to electron tracks, from each track we hope to automatically determine the electron's kinetic energy, along with other data.

Equation (2.12) shows that any electron undergoing relativistic cyclotron motion
will radiate away energy, and equation (2.11) demonstrates an inverse relationship between a relativistic electron's energy and the frequency of its cyclotron radiation. That is, as an electron sheds energy through cyclotron radiation, its cyclotron frequency (and so the frequency of its cyclotron radiation) increases. So the frequency of cyclotron radiation from relativistic electrons must be monotonically increasing in time. At the time-scales of Project 8, it is safe to treat this increase in frequency as linear.

The next chapter introduces the Hough transform as a tool for detecting straight lines in data like Project 8's spectrograms. In actual fact, the signal is a bit more complex: electrons have been found to inelastically scatter a few times before escaping the trap, so the radiation from a single particle discontinuously and discretely decreases in energy a few times as it scatters. Each time the electron scatters, it loses a discrete amount of energy, which produces a "staircase" effect in a spectrogram of the event [2].

The details of how these tracks are identified and characterized are contained in chapter 4.
Chapter 3

Hough Transform

3.1 Overview

This chapter will introduce the Hough transform as a tool for feature extraction in Project 8 data. Project 8 runs can produce huge volumes of data, out of which we are interested only in specific and brief events: the radiation emitted by captured electrons. A captured electron produces a characteristic chirp, which is easily identifiable to the naked eye in a spectrogram, but the sheer volume of data and the truly random nature of nuclear decay make it unreasonable for people to conduct such a search. Like so many tasks easy for the human eye and brain, this search is nontrivial to implement in software. The Hough transform provides a crucial step in automating the identification of these events.

The Hough transform is commonly used in modern computer vision, and it has been thoroughly adapted and expanded since it was first proposed in the 1960s. Our application, however, stays loyal to Paul Hough’s first implementation, not only in that we are using it to search for straight lines in a noisy image, but also in its direct application to particle physics.

The mathematics behind the Hough transform is fairly straightforward, involving the transformation of an image to a parameter space where each point has a one-to-one correspondence with a unique line in the original image. Each point in the input image likewise maps to a sinusoid in the parameter space, and points where these
curves overlap describe lines which multiple points lie along, allowing us to isolate lines even when when the collinear points are not adjacent, or when the image is noisy.

A short example is provided at the end of the chapter to better illustrate the process, and to further highlight its utility for Project 8.

### 3.2 Hough Transform motivation

As described in chapter 2, an electron trapped in the Project 8 apparatus emits narrow-band cyclotron radiation whose frequency increases as energy is lost and the particle's relativistic mass $\gamma m_e$ decreases. In a spectrogram of this event, a trapped electron has a characteristic appearance as a series of short line segments, increasing in frequency as the electron loses energy to radiation, and punctuated by discontinuities corresponding to inelastic scattering.

![Figure 3-1: an electron signal, from [2]](image)

In order to reconstruct an accurate spectrum, we must tally the energies of huge numbers of these captured electrons. Depending on the run, this may be a few hundred events, or a few hundred-thousand [10]. For CRES to be plausible as a spectroscopic technique, this count should be automated, starting with the identification of an electron signature. A captured electron produces a characteristic signal
in Project 8 spectrograms: notably, it's composed of groups of high-intensity straight lines. This is where we can find utility in the Hough transform.

Applying a Hough transform to an image maps the original into a parameter space, the axes of which parametrize possible lines. If we apply a Hough transform to an image containing a distinct line, the point corresponding to that line in parameter space piles up with votes—if the parameter space is implemented as a histogram, a line in the original image will produce a peak in the output. All that remains to isolate the line is the fairly straightforward process of peak-finding in the output histogram.

The precise details of the implementation of the Hough transform in Project 8 will be discussed in the next chapter. Here, we will instead discuss the Hough transform more generally, focusing on the theory behind it, and a general description of an algorithm to perform it. So while we will eventually apply the technique two-dimensional histograms containing spectrograms of RF data, we will simplify the language here, speaking more generally of "images." This follows the convention set by Duda and Hart in their work on Hough transforms, at the time envisioned as a tool for feature extraction in real-world photographs [15].

\[1\] We can also parametrize other structures, like circles. Any case other than straight lines is considered a generalized Hough transform.
3.3 How a Hough Transform works

To facilitate the identification of straight lines in the input, we will transform the input spectrogram onto a parameter space. In order to map a line to a point, the axes of our parameter space must describe any possible line in the original image. Of course, there’s more than one way to parametrize a line. At first, we might think of having our axes represent a line’s y-intercept and slope, and this is equivalent to Hough’s original formulation of the transform in his 1962 patent, although he phrased it purely geometrically [16][15]. A point-slope parametrization is bothersome, however, because slope is unbounded. We can easily imagine a particularly steep line corresponding to a point in our parameter space which lies well outside of any reasonable bounds on the axes of our transformed image. Hough’s original work-around was to run the transform twice, at right angles, on a single image [15]. This is workable, but not ideal: it increases the complexity of eventual feature extraction and doubles the algorithm’s runtime.

A more elegant approach to parametrization was proposed by Duda and Hart [7]. In this formulation, a line is parametrized by the length, $\rho$, and angle of inclination, $\theta$, of a perpendicular intercept joining it to the origin. Equivalently, we can think of $(\rho, \theta)$ as the polar coordinate of the line’s closest approach to the origin.

![Figure 3-3: Duda and Hart’s $(\rho, \theta)$ parametrization](image)

If we restrict $\theta$ to $0 \leq \theta < \pi$, then every possible line in our spectrogram cor-
responds to a single point in Hough space. The range of $\rho$, on the other hand, will depend on the scale of our original image—clearly there will be no need to parametrize lines which lie outside of our original plot.

Each point in our image defines a family of lines which pass through it. A point $(x, y)$ in our image defines a family of lines $(\rho_i, \theta_i)$ which pass through it. It can be shown that all of these lines satisfy

$$\rho_i = x \cos \theta_i + y \sin \theta_i$$  \hspace{1cm} (3.1)

So in our parameter space, each image point becomes the curve

$$\rho(\theta) = x \cos \theta + y \sin \theta$$  \hspace{1cm} (3.2)

As we perform our transformation, we plot a series of sinusoids, one for each point in our input image.

In the context of computer vision, each “point” is a pixel in the input image; in Project 8, it is a bin in the two-dimensional histogram which stores the spectrogram data. In either case, the point has an $(x, y)$ position and a value (hue or bin-value respectively).

As sinusoids are drawn to the parameter space, they are weighted by the value of the input point. Whenever two curves overlap in the parameter space, that point will be given an even higher weight, as the weights of the two curves are added to each other at the point where they overlap.

Each point in the Hough transform parametrizes a line that may exist in the input image. For two arbitrary points in the input image, the line which connects them is represented in the parameter space by the intersection of the two sinusoids corresponding to those points. So, if the original spectrogram contains a set of highly weighted collinear points, we would expect their corresponding sinusoids to overlap at one point.

By finding the places with the most overlapping curves in the final Hough transform, we can quickly determine the best-fit parametrization of any line-like structures.
The input is a two-dimensional histogram, consisting of a number of bins of a certain resolution. The bin’s height corresponds to the resolution in frequency, and its width corresponds to time resolution.

Our algorithm must cycle through each of these bins, and consider the parametrization of each line which passes through it, plotting that point to an output Hough-space \((\rho, \theta)\) histogram.

For an input histogram of width \(n\) bins, and height \(m\) bins, a pseudocode version of the Hough transform looks like:

Step through bins along horizontal axis, \(i = 0\) to \(n\)
Step through bins along vertical axis, \(j = 0\) to \(m\)
Step through lines which intersect bin \((i, j)\)
for each of these lines, plot its parametrization to the output histogram, weighted to the same extent as the input bin \((i, j)\)

The algorithm fundamentally consists of three nested loops. The first two serve to sequentially visit each bin in the input histogram. The third is the real meat of the process, which actually performs the transformation from frequency-time space to \((\rho, \theta)\)-space by calculating the parameters of each of a family of lines and plotting them to the output Hough-space histogram with the same weight as the input bin in question. Computationally speaking, we cannot actually consider the uncountable family of lines passing through each bin. Instead, we define a finite but arbitrarily small step size, by which we increment the slope of the line we are considering.

Once each input bin has been processed, we are left with the final Hough transform of the image. The highest peaks in the Hough-space histogram will correspond to the best-fit lines of any linear structures in the input.

A C++ implementation of the Hough Transform is included in Appendix A.
Chapter 4

The Data Processing Chain in Project 8

4.1 Overview

The Hough transform forms a part of the signal processing chain in Project 8. From the digitized raw frequency data, the aim is to conduct an automated search for the characteristic series of chirps generated by a captured electron’s cyclotron radiation.

Project 8 data pass through several stages before it is possible to extract individual electron energies. The Hough Transform is utilized near the end of the signal-processing chain to assist in finding the best-fit of lines appearing in electron tracks.

This chapter will outline the processes by which the raw data of Project 8 are processed to identify electron tracks and quantify them for the purposes of CRES.

4.2 The Expected Characteristics of a Project 8 Signal

In Chapter 2, we discussed the characteristics of an electron signal as it would appear in a spectrogram of Project 8 data. In summary, we are in search of events comprising
one or more narrow-bandwidth upward-sloping line segments, separated by discrete jumps in frequency due to scattering. Once we have identified these events, we can find the onset frequency to determine the electron’s initial energy, the statistic most pertinent to spectroscopy.

Here, each short line segment is referred to as a “track,” and the sum of all of the tracks produced by an electron as it scatters and is re-captured is referred to as an “event”.

4.3 Data Collection

Electromagnetic waves are conveyed from the interior of the Project 8 spectrometer by a waveguide. Along with the intended signal, the waveguide will also inevitably transmit some thermal noise. The output is sent through a high-gain, low-noise amplification chain before it is digitized [2].

The amplified signal is digitized and written to disk as time-series data [2].

Recent runs have made use of a real-time spectrum analyzer (RSA) to trigger recording only when a sufficiently strong signal is detected. “Sufficiently strong” in this case means that the RSA has detected a signal which surpasses the system’s noise floor by a threshold generated by the instrument\(^1\) of around 6 dB [10]. Once triggered, the RSA records 5 ms of data, beginning 1 ms before the triggering event. Once these events have been recorded, software processing is used to determine whether the recordings truly represent instances of electron cyclotron radiation, and then to extract useful information from each such instance. Depending on the run, there may be as many as several hundred thousand events to sort through, some of which are not true electron tracks, making the automation of data-processing absolutely vital to the experiment’s success.

Furthermore, there was no triggering in the earliest experimental runs, and so data acquisition was continuous. The very first individual electrons successfully observed by Project 8 were fished out from these continuously recorded swaths of data. In

\(^1\)a Tektronix 5000 series model
these early runs, it was absolutely vital that we had a reliable computational means of identifying and characterizing events; otherwise, they would have been lost among the background.

### 4.4 Data Processing in Katydid

The software package used for data processing in Project 8 is called Katydid, an effort largely attributable to Noah Oblath. All processes described in this section are performed by different aspects of Katydid, a large and extensible modular framework built specifically for the demands of Project 8.

#### 4.4.1 A spectrogram is made and refined.

The initial input in the data processing chain is the digitized RF output from the Project 8 apparatus. This is digitized time series data with a sampling rate of 50 MHz. First, a spectrograph is created from the time series data: the time series data are "sliced" into 30 µs intervals, and a Fast Fourier Transform (FFT) is applied to each successively, so we are left with a power spectrum of each 30 µs slice of time. These slices are arranged in temporal order to form a spectrogram: a 2-dimensional histogram whose x-axis represents time, y-axis represents frequency, and bin-height represents power spectral density.

From here, we are only interested in the highest-power bins. These, barring a few due to noise, should be emissions from captured electrons. The next step, then, is to simplify the spectrograms by copying only the highest-power bins over to another histogram. This is done by applying a masking function which only selects bins which have a higher power than a set threshold (a few dB above the background level). We are left with a sparse spectrogram, which is empty except for the highest-power bins of the original.

---

2This section owes a great deal to [19]
4.4.2 Tracks are located using DBSCAN.

Next, we apply the clustering algorithm DBSCAN to the sparse spectrogram. DBSCAN identifies clusters of points by examining each point’s neighbors (defined as a point lying within a certain distance), and it’s neighbors’ neighbors. Briefly, points are considered to belong to a single cluster if they share at least a “neighbor’s neighbor” relationship with each-other\(^3\). DBSCAN returns a set of clusters. Each of these clusters which exceeds a set size is presumed to be an electron track, a single line segment resulting from a captured electron’s uninterrupted emission of cyclotron radiation before it scatters. In this manner, DBSCAN is used for feature identification as well as for vetting of the RSA’s preliminary candidate identification.

4.4.3 Tracks are characterized by a Hough transform, and stitched together into complete events.

Each of the electron tracks as identified by DBSCAN is subjected to a Hough transform, to find the parametrization of the line which best coincides with the track. A Hough transform is applied to the track, and then the peak in Hough space is taken as the most accurate best-fit line for that segment. This gives us more information about the track, including its exact position and its slope. The magnitude of the track’s slope can serve as a way to measure the electron’s power output.

Finally, multiple tracks may be stitched together into an event. The endpoints of tracks are located in time, and if the end-time of one is close enough to (without overlapping with) the beginning of another, the two tracks are bundled together as part of the same event. Once all tracks have been processed into events, the initial energy of the electron can be determined from the frequency of the first track’s onset.

\(^3\)for a more in-depth treatment of DBSCAN, see [17]
Chapter 5

Conclusion

5.1 Results, Current Capabilities

In 2014, Project 8 announced the successful observation of single-electron cyclotron radiation, and the reconstruction of the electron emission spectrum of $^{83m}$Kr using the CRES technique [2]. The result represents a proof-of-concept for CRES as an effective method for high-energy electron spectroscopy, although the prototype device did not demonstrate a resolution which would challenge conventional electron spectroscopy techniques. The Project 8 survey of the $^{83m}$Kr spectrum was in good agreement with previous results.

Current limitations on resolution in Project 8 are largely due to uncertainty in the strength and field inhomogeneities of the magnet [17].

5.2 Future Prospects

Several improvements to the design are being studied and implemented. Chief among these improvements is a magnetic trap which subjects a captured electron to less field variation for more of their trajectory. This will bring the cyclotron emission of these electrons closer to the theoretical case of zero field variation, making it possible to more accurately determine their energies.

As it stands, there are no theoretical barriers to sub-electronvolt resolution spec-
troscopy with Project 8. When the apparatus has been tuned sufficiently to perform single-electronvolt-resolution electron spectroscopy, the ability to measure absolute neutrino mass by using CRES to observe the $\beta$-decay spectrum of tritium will be within reach.
Appendix A

appendix: code

```cpp
// hough.cc : performs a hough transform on a histogram, and outputs another
// histogram to the same .root file

#include <TH2D.h>
#include <TFile.h>
#include "TSpectrum2.h"
#include <TMath.h>
#include <iostream>
#include <cstdlib>

Double_t thetaStep = 100000.0;

void hough(TH2D* input, TH2D* output, char flag = 0)
// the flag should be 'n' if we are producing an xform for normalization
{
    for(Int_t time = 1; time <= input->GetNbinsX(); time++) // step through x-axis (time)
    {
        for(Int_t freq = 1; freq <= input->GetNbinsY(); freq++) // step through y-axis (frequency)
        {
            for(Double_t theta = TMath::Pi() / 2.0;
```
theta < TMath::Pi();
theta += (TMath::Pi() / (2.0*thetaStep)) // step through
    // values of theta.
{
    // the vector (R, theta) is the closest approach to the
    // origin of the line we are fitting, i.e., it is
    // perpendicular to the line we are fitting, and
    // intersects the origin of our x-y space.

    Double_t R = 0.0;

    // find R given theta, x (time), and y (freq)
    // R = ((x^2 + y^2) - 0.5)*cos(theta - arctan(y/x))
    R = TMath::Sqrt(time*time + freq*freq)
       *TMath::Cos(theta - TMath::ATan2(freq , time));

    // if we're making an xform for normalization, we don't
    // want to weight by bin content...
    if(flag == 'n')
        output->Fill(R, theta);
    else // ...but usually, we *do* want to weight by bin content.
        output->Fill(R, theta, input->GetBinContent(time, freq));
}
return;

void normalize(TH2D* input, TH2D* output)
{
// This will be our "background" hough graph for normalization.
// It has the same dimensions as our output histogram.
TH2D* bg = new TH2D("normalization", "normalization",
output->GetNbinsX(),
output->GetXaxis()->GetXmin(),
output->GetXaxis()->GetXmax(),
output->GetNbinsY(),
output->GetYaxis()->GetYmin(),
output->GetYaxis()->GetYmax(),
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kTRUE,
TH2D::kTRUE,
TH2D::kTRUE,
TH2D::kTRUE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
TH2D::kFALSE,
output->GetXaxis() ->GetXmax(),
output->GetNbinsY(),
output->GetYaxis() ->GetXmin(),
output->GetYaxis() ->GetXmax();

// fill bg with an unweighted hough transform of our x–y space.
hough(input, bg, 'n');

// normalize output against bg by dividing all equivalent bins by
// the normalization bins.
for(Int_t x=1; x <= output->GetNbinsX(); x++)
{
    for(Int_t y=1; y <= output->GetNbinsY(); y++)
    {
        if(bg->GetBinContent(x,y) != 0)
            output->SetBinContent(x, y, output->GetBinContent(x,y)
                / bg->GetBinContent(x,y));
    }
}
delete bg;

TH2D* changeToMandB(TH2D* rtheta)
    // returns a histogram of the hough transform in (slope, intercept)
    // space
{
    TH2D* mb = new TH2D("mbhough",
        "Hough transform in slope–intercept space",
        500, -5, 5, 500, -200, 200);

    Double_t m = 0.;
    Double_t b = 0.;

    for(Int_t rBin=1; rBin <= rtheta->GetNbinsX(); rBin++)
Double_t $r = r\theta->GetXaxis()->GetBinLowEdge(rBin);$  
for(Int_t thetaBin=1; thetaBin <= r\theta->GetNbinsY(); thetaBin++)
{
  Double_t $\theta = r\theta->GetYaxis()->GetBinLowEdge(thetaBin);$  
m = -1/(TMath::Tan(\theta));  
b = $r / (TMath::Sin(\theta));$
  // if(r\theta->GetBinContent(rBin,thetaBin) != 0)
  // {
  //   std::cout << "r: "r << "\n" theta: "theta << std::endl;
  //   std::cout << "\n" untranscribing: "r\theta->GetBinContent(rBin, thetaBin)
  //   << "\n" << std::endl;
  //   std::cout << "m: "m << "\n" b: "b << "\n\n" << std::endl;
  // }
  mb->Fill(m,b,r\theta->GetBinContent(rBin,thetaBin));
}
return mb;
}

void findPeaks(TH2D* hist)
{
  TSpectrum2* analyzer = new TSpectrum2();
  // analyzer->Search(hist);
  Int_t i,j,nfound = 0;
  const Int_t nbinsx = hist->GetNbinsX();
  const Int_t nbinsy = hist->GetNbinsY();
  std::cout << "nbinsx: " << nbinsx << std::endl;
  std::cout << "nbinsy: " << nbinsy << std::endl;
  Double_t xmin = 0.0;
  Double_t xmax = (Double_t)nbinsx;
  Double_t ymin = 0.0;
Double_t ymax = (Double_t)nbsy;
// over here
Float_t *source[nbsy];
for(i=0; i<nbinsx ; i++)
{
    source[i] = new Float_t[nbinsx];
}

Float_t *dest[nbsy];
for(i=0; i<nbinsx ; i++)
{
    dest[i] = new Float_t[nbinsx];
}

for(i=0; i<nbinsx ; i++)
{
    for(j=0; j<nbinsy; j++)
    {
        source[i][j] = 0.0;
        dest[i][j] = 0.0;
    }
}

for(i=0 ; i<nbinsx ; i++)
{
    for(j=0 ; j<nbinsy ; j++)
    {
        source[i][j] = hist->GetBinContent(i+1,j+1);
    }
}

nfound = analyzer->SearchHighRes(source ,dest ,nbinsx,nbinsy,5.0,80.0,
kTRUE,10,kTRUE,3);
std::cout << "NPeaks is: " << nfound << std::endl;
Float_t* peaksX = analyzer->GetPositionX();
Float_t* peaksY = analyzer->GetPositionY();

for (i = 0; i < analyzer->GetNPeaks(); i++)
{
    Double_t r = hist->GetXaxis()->GetBinCenter((int)peaksX[i]);
    Double_t theta = hist->GetYaxis()->GetBinCenter((int)peaksY[i]);
    Double_t m = -1/(TMath::Tan(theta));
    Double_t b = r/(TMath::Sin(theta));
    std::cout << "peak #" << i+1 << " (" << peaksX[i] << ", " << peaksY[i] << "); "
    << "m = " << m << ", b = " << b << ", r = " << r << ", theta = "
    << theta << std::endl;
}

// analyzer->Search(hist, 5, "col", 0.5);

// nfound = analyzer->GetNPeaks();

// std::cout << "NPeaks is: " << nfound << std::endl;

// Float_t* peaksX = analyzer->GetPositionX();
// Float_t* peaksY = analyzer->GetPositionY();

// for (i = 0; i < analyzer->GetNPeaks(); i++)
// {
//     Double_t r = hist->GetXaxis()->GetBinCenter((int)peaksX[i]);
//     Double_t theta = hist->GetYaxis()->GetBinCenter((int)peaksY[i]);
//     Double_t m = -1/(TMath::Tan(theta));
//     Double_t b = r/(TMath::Sin(theta));
//     std::cout << "peak #" << i+1 << " (" << peaksX[i] << ", " << peaksY[i] << "); "
//     
// }
TH2D* threshold(TH2D* hist)
{
    // first chop off everything above some threshold, right now, let's call it 28, for no good reason.
    // then return the resulting histogram.
    TH2D* output = new TH2D("threshold", "Thresholded image",
        hist->GetNbinsX(), hist->GetXaxis()->GetXmin(), hist->GetXaxis()->GetXmax(),
        hist->GetNbinsY(), hist->GetYaxis()->GetXmin(), hist->GetYaxis()->GetXmax());

    for (int rBin = 1; rBin <= hist->GetNbinsX(); rBin++)
    {
        for (int thetaBin = 1; thetaBin <= hist->GetNbinsY(); thetaBin++)
        {
            if (hist->GetBinContent(rBin, thetaBin) > 28)
                output->SetBinContent(rBin, thetaBin);
        }
    }

    return output;
}

int main(int argc, char** argv)
{
    // this is the file which we made in 'generate'
    return 0;
}
TFile* file = TFile::Open("sample.root","UPDATE");

// retrieve the histogram with the straight lines in it
TH2D* input = 0;
file->GetObject("lines",input);

// this is where we will store our theta vs. R hough xform
TH2D* output = new TH2D("hough","Hough transform",
1000,-100,150,1000,1.5,3.2);

hough(input, output);
normalize(input, output);
findPeaks(output);
TH2D* outputThresh = threshold(output);

TH2D* outputMB = changeToMandB(output);

output->Write();
outputMB->Write();
outputThresh->Write();

// clean up
delete output;
delete input;
delete outputMB;
file->Close("R");
delete file;

return 0;
Bibliography


