

# New analysis method of TPC data using neural network

Takanobu Doi<sup>1</sup>, Takahiro KAWABATA<sup>2</sup>, Tatsuya FURUNO<sup>3</sup>, Yuki FUJIKAWA<sup>1</sup>, Kento INABA<sup>1</sup>,  
Motoki MURATA<sup>3</sup>, Shintaro OKAMOTO<sup>1</sup> and Akane SAKAUE<sup>1</sup>

<sup>1</sup>Department of Physics, Kyoto University, Kyoto, Kyoto 606-8502, Japan

<sup>2</sup>Department of Physics, Osaka University, Toyonaka, Osaka 540-0043, Japan

<sup>3</sup>Research Center for Nuclear Physics, Osaka University, Ibaraki, Osaka 567-0047, Japan

E-mail: doi.takanobu.68x@st.kyoto-u.ac.jp

(Received 2019/8/4)

TPC を用いた実験では荷電粒子の飛跡を測定することができる。TPC では 3 次元的な荷電粒子の飛跡を 2 次元平面に射影した画像として得られるので、データの解析には画像解析を行う必要がある。従来は Hough 変換を用いてデータの解析を行って来たが、この方法を用いた解析には多くの労力を要する。近年、画像データの認識にはニューラルネットワークが注目されている。本研究では、新しくニューラルネットワークを用いた解析方法の開発を行った。新しく開発した手法を用いることで、従来の方法と比較して高速に解析できるようになった。

**KEYWORDS:** neural networks, convolutional neural networks, TPC, active target, MAIKo TPC

## 1. Introduction

近年、荷電粒子の飛跡を測定する検出器として Time Projection Chamber (TPC) が広く用いられている。我々は不安定核実験に用いるために Micro Pixel Chamber ( $\mu$ -PIC) [1] (Fig. 2) で読み出しを行う、Mu-Pic based Active target for Inverse Kinematics . (MAIKo) TPC [2] (Fig. 1) を開発した。TPC は検出器にガスを充填しておき、荷電粒子がガス中を通過するとき発生する電子をドリフト電場により読み出し面にドリフトさせる (Fig. 1 中では下向きに移動させる) ことで飛跡を検出する。 $\mu$ -PIC は 2 つの直交する読み出しストリップを持つので、電子が到達した  $x, z$  座標を決定できる。また、電子が到達するまでのドリフト時間により、飛跡から読み出し面までの距離 ( $y$  座標) を決定することが出来る。 $x, y, z$  の座標が決定できるので、飛跡を 3 次元的に再構成することが出来る。

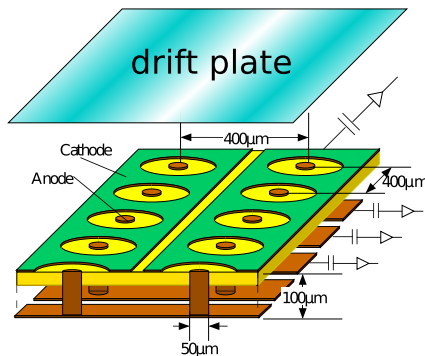


Fig. 1.  $\mu$ -PIC の構造

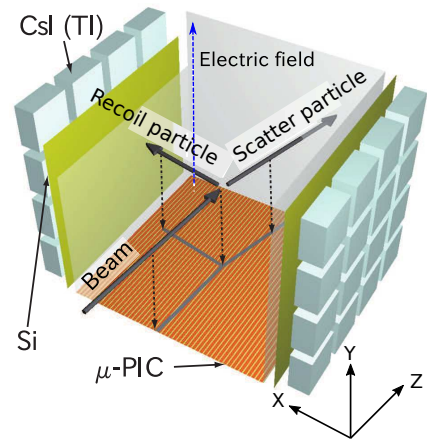


Fig. 2. MAIKo TPC の外観。

この検出器では検出ガスを実験の標的として用いる (アクティブ標的)。この手法を用いることで、散乱が検出器内部で起こるため、大立体角で低エネルギー粒子の測定を行うことが出来る。広く標的ガスとして用いられる He や  $H_2$  は放電耐性が低いので、通常は放電耐性の高い  $CO_2$  やイソブタンをクエンチングガスとして混入させて TPC を運用する。このとき、クエンチングガスからの散乱は背景事象となる。近年、大阪大学核物理研究センター (RCNP) において、MAIKo TPC を用いた  $^{10}C$  と  $^4He$  の非弾性散乱の測定が初めて行われた。この実験では標的ガスとして He、クエンチングガスとして  $CO_2$  を用いた。

荷電粒子の飛跡はビーム軸に平行な面に射影された画像 (anode track) とビーム軸に垂直な面に射影された画像 (cathode track) の 2 つの画像として取得される。 $\mu$ -PIC のストリップは  $400\ \mu m$  間隔で 256 本あり、時間方向は 100 MHz で 1024 sample 測定されるので、取得される画像の解像度は  $256 \times 1024$  pixels となる。Fig. 3, 4 は RCNP で行われた実験で取得されたデータの一例である。Fig. 3 は He ガスと散乱した事象、Fig. 4 はクエンチングガスと散乱した事象である。

散乱実験では、散乱粒子のエネルギーと放出角度を決定する必要がある。散乱粒子のエネルギーは、散乱粒子がガス中で停止するまでの距離から決定することが出来る。飛跡情報の抽出を行うためには

- 標的との散乱事象と背景事象との選別
- 飛跡の長さや方向などの物理的情報の抽出

の大きくわけて 2 つの解析が必要になる。

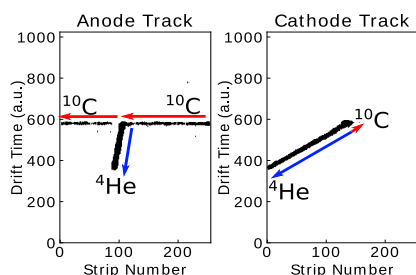


Fig. 3.  $^{10}C + ^4He$  の散乱事象

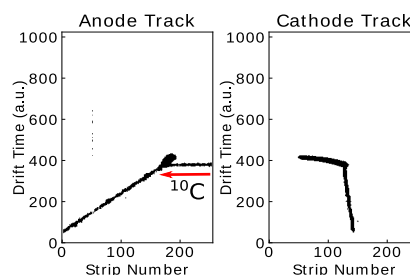


Fig. 4.  $^{10}C$  とクエンチガスとの散乱事象

Fig. 3, 4 のような画像データに対して、事象の選別や飛跡情報の抽出には多くの労力が必要となる。そこで、我々は近年画像認識において多くの成果を出しているニューラルネットワークを用いることで、より高速な解析を行うことが出来る解析方法の開発を行った。

## 2. Conventional analysis method

従来は、得られたデータから背景事象の除去を行うのに、画像中の直線を抽出する手法の 1 つである Hough 変換を用いたアルゴリズムによって行ってきた。Hough 変換を行うことで画像に含まれる軌跡上の点は、原点からの距離 ( $r$ ) と傾き ( $\theta$ ) の 2 つの軸を持つパラメータ空間における曲線へ変換される。元の画像における直線状の点はすべて同じ  $r$  と  $\theta$  を共有するので、このパラメータ空間において複数の曲線が交わる点を求めれば  $r$  と  $\theta$  が一意に決定され、元の画像における直線を抽出することができる。抽出した直線に沿って端点を探すことで、画像に含まれる直線の方法や長さなどを決定することができる。決定された直線の長さや角度、本数、配置に対して多くの条件を課すことによって、He との散乱事象を選別することが出来る。この条件を設定するには多くのパラメータを必要とし、分岐も複雑なものとなる。そのため、パラメータの最適化や画像識別に多くの時間を必要とする。さらに、選別されたデータ中の He の飛跡の長さや角度を求めることで、散乱角とエネルギーを決定することが出来る。

パラメータの最適化には 100 CPU を使用して約 1 日かかり、その後の識別には 1 event あたり 1 秒かかった。この手法による識別能力の評価には、画像データを人間が目で見えて散乱事象もしくは背景事象と判断した 3000 イベントを評価用データとして用いて行った。全事象のうち、散乱事象または背景事象の区別を正しく判断することが出来た事象数の割合を正解率としたとき、この従来型の解析手法では 89% の正解率であった。

### 3. New analysis method

従来手法では識別に複雑な分岐条件や多くの計算機パワーが必要になるという問題点があった。そこで、我々は近年注目されているニューラルネットワークを用いることで、これらの問題点を克服する新しい解析方法の開発を試みた。ニューラルネットワークを用いることで、複雑な条件分岐を導入することなく、直線の位置、角度などの従来の解析手法ではまとめて扱うことの難しい多くの特徴量を同時に考慮した画像識別が可能になると期待される。また、ニューラルネットワークは一度構築するとその後は短時間で画像識別を行うことができる。このようなニューラルネットワークの特徴を活かすことで、従来のアルゴリズムでは実現が難しかった高い精度と短い識別時間を実現することが可能である。

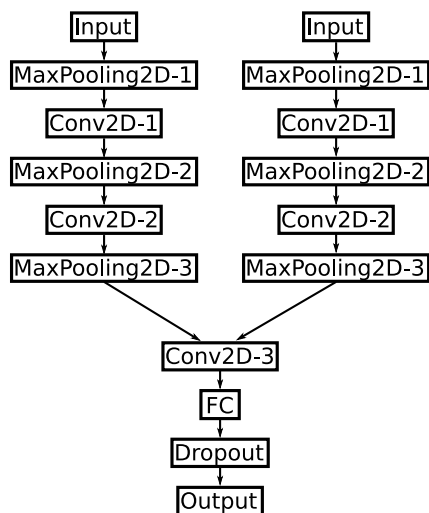


Fig. 5. データの選別を行うためのニューラルネットワーク

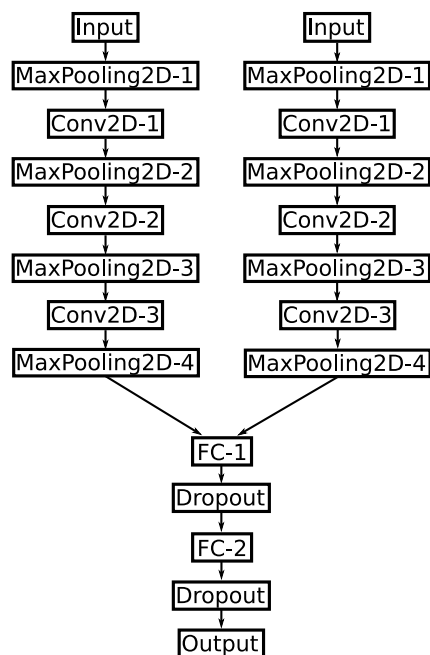


Fig. 6. 画像から情報を抽出するためのニューラルネットワーク

MAIKo TPC から得られるデータが画像であるため、画像認識に有用とされる Convolutional Neural Network (CNN) を用いた。CNN とはネットワーク内に畳み込み層を有するネットワーク構造である。解析には事象の選別と飛跡情報の抽出の 2 つの段階があるため、2 種類のニューラルネットワークを用いて解析を行った。事象の選別を行うニューラルネットワークと画像から飛跡情報を抽出するニューラルネットワークの構造をそれぞれ Fig. 5 と Fig. 6 に示す。

事象選別のためのニューラルネットワークは、MAIKo TPC から得られる 2 つの画像を入力し、その事象が標的と散乱した事象である確率を出力する。出力された確率がある値以上である場合にその事象を標的と散乱した事象であると判断する。側面に射影した飛跡とビーム方向に射影した飛跡とは画像の持つ意味が異なるため、2 つの入力が別れた構造を持つネット

ワークを構築した。人間が判断したデータを用いて、学習および評価を行った。学習には 2,700 events、評価には 300 events を使用した。

飛跡情報を抽出するためのネットワークは、MAIKo TPC から得られる 2 つの平面に射影された飛跡画像を入力し、散乱が起こった座標と反跳した  $^4\text{He}$  が停止した座標を出力する。教師データには従来手法で決定した座標を用いた。学習には 3,012 events、評価には 1,554 events を使用した。

学習環境は Intel Core i7、Nvidia GeForce GTX 1080Ti、Ubuntu 18.04 LTS、TensorFlow [3]+Keras [4] を用いた。

## 4. Result

事象選別のためのニューラルネットワークは、2,700 events に対して 200 回学習を行った。学習にはおよそ 26 分、その後の推測には 300 events に対しておよそ 1 秒かかった。従来手法の評価に用いた実験データのうち、300 イベントを用いて評価を行った結果、正解率は 96% であった。従来の手法と比較して高精度に事象選別を行うことが可能となった。また、パラメータチューニングおよび選別に要する時間も大きく短縮することができた。

座標の決定のためのニューラルネットワークは、3,012 events に対して 500 回学習を行った。学習にはおよそ 270 分、その後の推測には 1,554 events およそ 2 秒かかった。学習後にニューラルネットワークが予測した座標と従来手法によって決定した座標の比較を Fig. 7 に示す。赤色の点は散乱が起こった点、緑色は反跳粒子が停止した点を表す。丸点は従来型の手法を用いて決定した点 (学習に用いたデータ)、バツ点はニューラルネットワークを用いて決定した点を表す。従来手法とニューラルネットワークを用いた手法のそれぞれで決定した座標は、ほぼ一致している。つまり、従来手法の精度を維持しつつ、従来の手法と比較してパラメータチューニングおよび情報の抽出にかかる時間を短縮することができた。

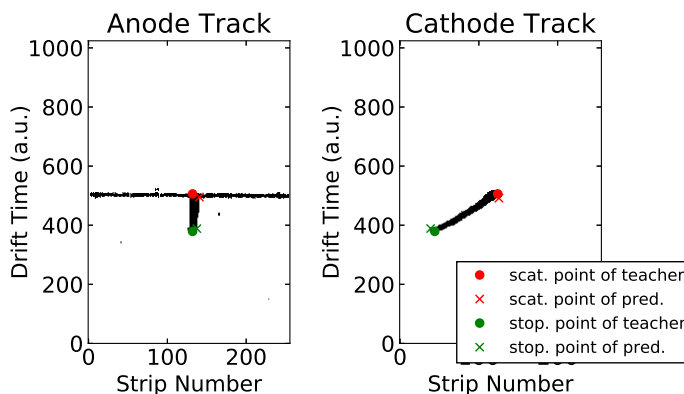


Fig. 7. ニューラルネットワークによって決定した座標と Hough 変換によって決定した座標との比較

## 5. Conclusion

従来の Hough 変換を用いた画像識別アルゴリズムに変わる、ニューラルネットワークを用いた新手法の開発を行った。新手法は従来のものよりも高速かつ高精度に事象の選別を行うことが可能となった。また、飛跡情報の抽出については従来の精度を保ちつつ、高速に行うことが可能となった。ニューラルネットワークを用いた解析手法は TPC の解析に有用であることがわかった。

## References

- [1] A. Ochi, T. Nagayoshi, T. Tanimori, T. Nagae, and M. Nakamura Nuclear Instruments and Methods in Physics Research A **471** 264 (2001)
- [2] T. Furuno, T. Kawabata, H. Ong, S. Adachi, Y. Ayyad, T. Baba, Y. Fujikawa, T. Hashimoto, K. Inaba, Y. Ishii, S. Kabuki, H. Kubo, Y. Matsuda, Y. Matsuoka, T. Mizumoto, T. Morimoto, M. Murata, T. Sawano, T. Suzuki, A. Takada, J. Tanaka, I. Tanihata, T. Tanimori, D. Tran, M. Tsumura, and H. Watanabe Nuclear Instruments and Methods in Physics Research A **908** 215 (2018)
- [3] A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefow-

icz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, (2015). <https://tensorflow.org>

[4] F. Chollet, et al. (2015). <https://keras.io>