画像認識で使用する画像は適切か、オートエンコーダーによる評価

EVALUATIONS WITH AUTOENCODER WHETHER THE IMAGE USED FOR IMAGE RECOGNITION IS APPROPRIATE

野村昌弘 *A)、沖田英史 A)、島田太平 A)、田村文彦 A)、山本昌亘 A)、
古澤将司 B)、杉山泰之 B)、長谷川豪志 B)、原圭吾 B)、大森千広 B)、吉井正人 B)
Masahiro Nomura* A), Hidefumi Okita A), Taihei Shimada A), Fumihiko Tamura A), Masanobu Yamamoto A),
Furusawa Masashi B), Yasuyuki Sugiyama B), Katsushi Hasegawa B), Keigo Hara B), Chihiro Ohmori B), Masahito Yoshii B)
A) Japan Atomic Energy Agency, JAEA
B) High Energy Accelerator Research Organization, KEK

Abstract

By using image recognition technology in J-PARC RCS, values such as momentum offset can be obtained from mountain plot images. In the future, when considering the use of the obtained value for control, it will be important whether the obtained value is reliable. This is because, in the image recognition technology, some value is returned even from the image in which the data acquisition fails. An image that returns a reliable value is an image similar to the trained images. Therefore, we applied the method of abnormality diagnosis by Autoencoder and tried to find an index showing the difference from the training images, that is, an index showing whether the value is reliable. As a result, it was confirmed that this index obtained by Autoencoder indicates whether the value obtained from the image is reliable.

1． 序

畳み込みニューラルネットワーク (Convolutional Neural Network:CNN) による画像認識の技術は、幅広い分野で用いられ、優れた成果を残している。J-PARC RCS ではこの画像認識技術を適用し、シミュレーションで作成したマウンテンプロットの画像を CNN に学習させることにより、実測したマウンテンプロットの画像から調整時に必要な運動量やタイミングオフセットが得られるようになった [1]。

今後、マウンテンプロットの画像から得られた値を制御に使用することを考えた場合、得られた値が信頼できるかが重要となってくる。なぜなら、CNN では学習した画像とは全く違う、データ取得に失敗した画像からでも何らかの値を返してくるからである。そこで、本研究では画像から得られた値が信頼できるかどうかを示す指標を求めてみた。

2． 指標を与えるニューラルネットワーク

得られた値が信頼できるかどうかは当然その画像で決まる。先ずは、信頼できる値を返す画像とはどのような画像かを考えてみる。答えは明白で、値を得るために学習した画像と類似の画像である。学習画像と類似の画像からは信頼できる値が得られるが、学習画像と全く違う画像からは信頼できる値は得られない。そこで、何かのニューラルネットワークに学習画像を理解させ、与えられた画像との比較ができれば、その画像から得られた値が信頼できるかどうかの指標が求まるはずである。また、どの様な画像が与えれるかは予想できず、機械学習の分類としては教師無し学習となる。つまり、求められるニューラルネットワークとしては、教師無しの機械学習で、学習画像を理解し、新たに与えられた画像との差異を判断できるニューラルネットワークと言える。見方を変えれば、その様な用途に用いられているニューラルネットワークは既に存在している。それは、異常診断におけるオートエンコーダーである。

3． オートエンコーダーによる異常診断

オートエンコーダーは、機械学習におけるニューラルネットワークの一種で、Encoder と Decoder から構成されている。入力層と出力層の次元は同じに設定され、中間層はそれよりも少ない次元となっている。学習では出力を入力に近づけるように行われる。

オートエンコーダーに限らず機械学習における異常診断を考えた場合、一般に正常なデータは多くあるが、異常なデータは数少ない場合がほとんどである。教師データとして正常と異常の両方のデータを十分に揃えることはできない。当然、教師無し学習となる。オートエンコーダーによる異常診断では、正常な状態とは違う状態を異常としている。

次に、オートエンコーダーによる具体的な異常診断の方法を述べる。オートエンコーダーには正常な状態のデータのみを与えて、出力を入力に近づけるように学習させる。これにより、正常な状態のデータが入力された場合は、出力データと入力データにほとんど差は生じないが、正常とは違った状態のデータが入力された場合には、出力データと入力データには差が生じてしまう。異常診断はこの入力データと出力データとの差を指標として行われている。この種の異常診断の例は [2, 3] 等を参照されたい。

このオートエンコーダーによる異常診断と今回のテーマである”信頼できる値を返す画像かどうか”は本質的には同じと考えられる。オートエンコーダー

* masahiro.nomura@j-parc.jp

による正常な状態を表すデータを今回の学習画像と考えると、信頼できる値を返さない画像は学習画像とはなんらかの違いがあるので、オートエンコーダーによる異常データに対応すると考えられる。つまり、このオートエンコーダーによる異常診断の手法を適用すれば、信頼できる値を返す画像かどうかの指標が得られるはずである。

4. マウンテンプロットの画像への適用

4.1 使用したオートエンコーダー

今回、オートエンコーダーとしては、参考文献 [2,3] を参考にして、全結合のオートエンコーダーを採用した。使用したオートエンコーダーの構成を Fig. 1 に示す。このオートエンコーダーでは、先ず入力であるマウンテンプロットの画像 [400,120] を 1 次元 [48000] のデータに変換し、その後 [48000] から [5000]、[5000] から [1000]、さらに [500] と次元圧縮を行い、最後に Decoder により、元の画像の次元まで戻している。入力データと出力データとの違いは平均二乗誤差 (mean square error:mse) で評価した。

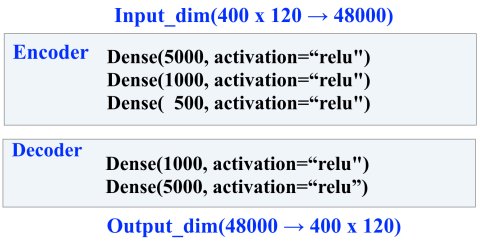

Figure 1: Configuration of the NN Autoencoder.

4.2 学習

正常データにあたる画像は、シミュレーションで 10000 枚作成し、そのうち 9000 枚を学習に、1000 枚を学習の検証用として使用した。シミュレーションでは、運動量とタイミングオフセット、運動量広がりと時間幅の 4 つを乱数で発生させ画像を作成している。画像は最大値を 1 に規格化し、白黒画像としている。学習画像の例を Fig. 2 に示す。

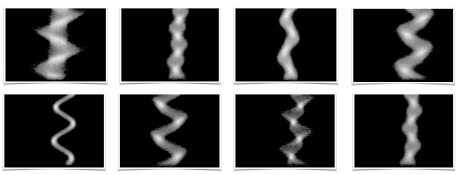

Figure 2: Mountainplot images for learning.

正常データである学習画像のみを使って、Epochs = 10 まで学習させた。学習による mse の減少と学習後の学習画像と検証画像の mse のヒストグラムを Fig. 3 に示す。ニューラルネットワークの学習では、常に過学習 (overfitting) になることを防がなければならない。今回は、異常の判断を mse で行うので、特に overfitting には注意しなければならない。なぜなら学習を進めすぎて overfitting になると、学習画像の mse は小さくなるが、正常画像である検証画像の mse は逆に大きくなる。つまり、与えられた画像が正常画像であっても、mse が大きくなり異常と判断されることになる。Figure 3 では、学習画像と検証画像の mse が同じ傾向で減少していること、さらに、両者の mse の分布が同じであることから、overfitting にはなっていないことが分かる。

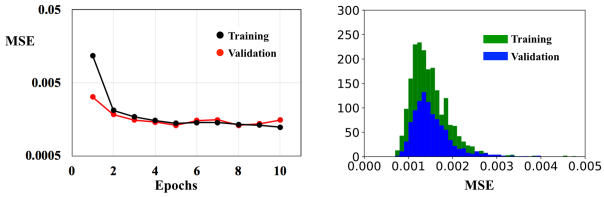

Figure 3: Learning curve and histogram of mse values.

4.3 学習範囲外の画像への適用

シミュレーションで学習画像を作成する際の運動量オフセットの値は −0.3% から +0.3% の範囲で乱数により決められている。ここでは学習範囲内と範囲外の画像の指標を調べるために、運動量オフセットの値を 0% から +0.1% づつ 0.7% まで増やした 8 枚の画像を準備した。それらの画像を Fig. 4 に示す。1 から 4 が学習範囲内、5 から 8 が学習範囲外の画像である。

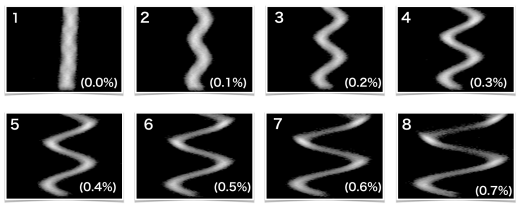

Figure 4: Mountainplot images.

画像を作った時の設定値と CNN により画像から得られた値の差、CNN による測定誤差と言える値を Fig. 5 に示す。学習範囲内の 1 から 4 の画像では、設定値と CNN により得られた値の差はほとんど無く、学習範囲外の 5 から 8 の画像では大きく違っていることが分かる。つまり、1 から 4 の画像が信頼できる値を返す画像で、5 から 8 の画像が信頼できる値を返さない画像である。

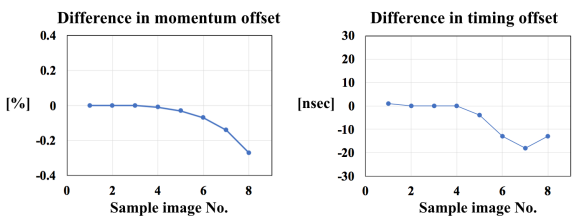

Figure 5: Difference between set value and CNN predicted value.

学習済みのオートエンコーダーにより求めた、1 から 8 の画像の mse のヒストグラムを Fig. 6 に示す。青色は学習における検証画像つまり正常画像を表し、赤色は指標を求める画像のヒストグラムを表している。1 から 4 の画像は、青色のヒストグラム内にあり、正常、つまり信頼できる値を返す画像であることが示されている。一方、5 から 8 の画像は、設定値と CNN により得られた値の差が広がるに従って青色のヒストグラムから外れていき、異常、信頼できる値を返さない画像であることが示されている。mse が適切な指標となっていることが分かる。

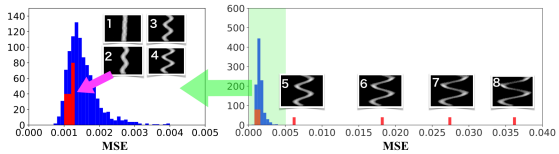

Figure 6: Histograms of mse values.

確認のために、オートエンコーダーにより、出力画像がどの様に再構築されたかを調べてみる。結果を Fig. 7 に示す。正常と判断された 1 と 2 の出力画像は入力画像との差は小さく、入力画像がほぼそのまま再現されている。一方、異常と判断された 7 と 8 の画像は、出力画像と入力画像の間に差が生じ、入力画像が再現されていない。これは、学習画像と違った画像は正確には再現されないと言う結果である。

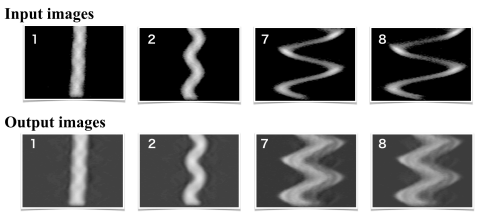

Figure 7: Input and output mountainplot images.

4.4 予想外の画像への適用

さらに、予想外の画像等を用いて mse が適切な指標となっているかを調べてみる。準備した画像は、正常と判断される画像としてシミュレーションでは無く、実測した画像を 4 枚、異常と判断される画像として、ビーム損失を模した画像、一部が欠けた画像、学習画像と全く違った画像、データ取得し失敗した画像の 4 枚、全部で 8 枚である。それらの画像を Fig. 8 に示す。

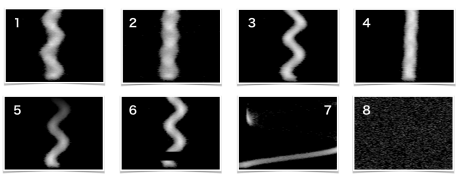

Figure 8: Mountainplot images.

学習済みのオートエンコーダーにより求めたそれらの画像の mse のヒストグラムを Fig. 9 に示す。実測画像は青色のヒストグラム内にあり正常に分類され、それ以外は異常に分類されている。さらに、学習画像と全く違った画像のほうが、少し違った画像よりも mse が大きいことも mse が適切な指標となっていることを示している。

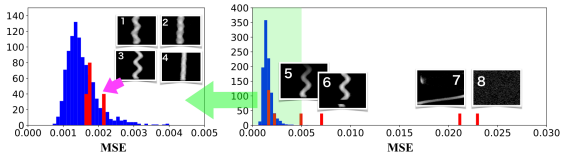

Figure 9: Histgrams of mse values.

確認のために、オートエンコーダーによりどの様に再構築されたかを調べてみる。結果を Fig. 10 に示す。画像 5 と 6 を見ると、出力画像は、ビーム損失を表した部分や欠損した部分が再現されておらず、入力画像を学習画像で表したらこうなるといった画像となっている。まさに、入力画像と出力画像の違いが、入力画像と学習画像との違いとなって現れている。予想どおりである。画像 7 と 8 は大きく学習画像と違っているため、全く再現できていないことが分かる。

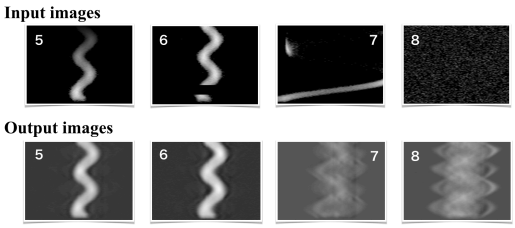

Figure 10: Input and output mountainplot images.

5. まとめ

今回、CNN により画像から得られた値が信頼できるかを示す指標をオートエンコーダーによる異常診断の手法を用いて求めてみた。その結果、与えられた入力画像とオートエンコーダーにより再構築された出力画像との差、mse がその指標となり得ることが示された。

参考文献

[1] M. Nomura *et al.*, Proceedings of 17th Annual Meeting of Particle Accelerator Society of Japan (2020) 64.
[2] DeepAutoencoder を利用した画像、時系列データの異常検知入門 (keras 編); `https://techblog.istyle.co.jp/archives/4318`
[3] Keras Autoencoder で異常検知をやってみる; `http://cedro3.com/ai/keras-Autoencoder-anomaly/`