

# 将棋用ニューラルネットワークへの顕著性抽出手法の適用

中屋敷 太一<sup>1,a)</sup> 金子 知適<sup>2,3,b)</sup>

**概要:** 近年コンピュータ将棋プログラムは対局で人間のプロ棋士に勝利し、棋力は人間のプロ棋士を超えたと言われている。一方で、将棋プログラムによる将棋の局面の解説は、局面の勝率予測や次の指し手の予想ではある程度成功しているものの、それが何を根拠に決定されたかという内容を説明する手法はまだ確立されていない。もし将棋プログラムの判断の根拠を、プログラムが自動で、プログラムの利用者に説明することができれば、人間の棋力向上のサポートなどに用いることができ有用である。本稿では、画像分類を行うニューラルネットワークに対して考案された顕著性抽出手法を、将棋プログラムに用いられているニューラルネットワークに適用した。また、顕著性抽出手法は元の画像のどの部分に注目して判断したのかを可視化する技術であるが、可視化する際にいくつかの手法が考えられるため、本稿ではそれらの手法の差異を検討した。

## Application of Saliency Extraction Methods to Neural Networks in Shogi

TAICHI NAKAYASHIKI<sup>1,a)</sup> TOMOYUKI KANEKO<sup>2,3,b)</sup>

**Abstract:** Computer shogi programs defeated human experts and it has been said that they outperform human experts. On the other hand, although prediction of the best move and winning percentage was successful to a certain degree, the method is not established yet to explain the reason why they are predicted so by computer shogi programs themselves. If computer shogi programs were able to explain the reason to users by itself, they could be used to support human players and be very useful. In this paper, we applied saliency extraction methods which were devised in image classification to neural networks in a computer shogi program. In order to compare variations in its implementation, we compared the difference between visualization methods of the result of saliency extraction methods, which visualize pixels where the neural network paid attention, because some methods can be conceivable.

### 1. はじめに

近年コンピュータ将棋プログラムは対局で人間のプロ棋士に勝利し<sup>\*1</sup>、棋力は人間のプロ棋士を超えたと言われて

いる。一方で、プロの対局の観戦中などでの将棋プログラムによる将棋の局面の解説は、局面の勝率予測や次の指し手の予想ではある程度成功しているものの [1], [2], それが何を根拠に決定されたかという内容を説明する手法はまだ確立されていない。もし将棋プログラムの判断の根拠を、プログラムが自動で、プログラムの利用者に説明することができれば、人間の棋力向上のサポートなどに用いることができ有用である。

### 2. 先行研究

#### 2.1 将棋プログラムによる局面の解説

これまでの研究では、プロの対局の観戦中などでの将棋プログラムによる将棋の局面の解説は、局面の勝率予測や、

<sup>1</sup> 東京大学大学院総合文化研究科  
Graduate School of Arts and Sciences, The University of Tokyo

<sup>2</sup> 東京大学大学院情報学環  
Interfaculty Initiative in Information Studies, the University of Tokyo

<sup>3</sup> 国立研究開発法人科学技術振興機構 さきがけ  
JST, PRESTO

a) [tnakayashiki@g.ecc.u-tokyo.ac.jp](mailto:tnakayashiki@g.ecc.u-tokyo.ac.jp)

b) [kaneko@acm.org](mailto:kaneko@acm.org)

\*1 将棋電王戦 HUMAN VS COMPUTER <http://denou.jp/> (Accessed: 2018-10-16)

次の指し手の予想をする点では、十分強い将棋プログラムを用いることである程度達成しているものの [1], [2], その判断の根拠を説明する手法はまだ確立されていない。

## 2.2 ニューラルネットワークと将棋プログラム

ニューラルネットワークを用いた将棋プログラムである AlphaZero は、対戦時に 4 個の TPU を使用し、第 27 回世界コンピュータ将棋選手権の優勝プログラムである elmo に対して 90 勝 8 敗 2 分と大きく勝ち越している [3]。なお AlphaZero の学習の際には、自己対戦のために第 1 世代の TPU 5000 個が、訓練のために第 2 世代の TPU 64 個が用いられた [3]。

AlphaZero と elmo では、AlphaZero の方が探索している局面数は少ないため [3]、将棋プログラムにニューラルネットワークを用いることで、従来の評価関数より局面を正確に評価していると考えられる。このようなニューラルネットワークを解析することができれば、人間が局面を評価する際の参考になり、人間の棋力向上のサポートなどに用いることができると考えられる。さらに現在の将棋プログラムでは、特徴量をプログラマが指定した、駒の関係を利用した評価関数が主流だが [4]、AlphaZero ではニューラルネットワークを用いることでこのような特徴量をプログラマが直接設定することなく学習に成功している。また自己対戦のみで強くなることに成功しており、人間のプロの棋譜などを用いていない [3]。そのため、ニューラルネットワークを解析することで、人間が将棋を指す上で、新たに判断に用いるべき指標がわかる可能性がある。

## 2.3 ニューラルネットワークの顕著性抽出手法

ニューラルネットワークの顕著性抽出手法には、勾配を用いて生成する Saliency Map [5] や、その改良手法 SmoothGrad [6] などがある。これらの手法は、画像認識を行うニューラルネットワークに対して考案された手法で、出力に対する各チャンネル、各ピクセルの影響力を調べる手法である。具体的には、出力ラベルを固定し入力ベクトルの勾配を計算することで、出力への影響力を調べる。

### 2.3.1 Saliency Map

画像認識を行うニューラルネットワークが、画像のどの部分に注目して認識しているのかということ解析する研究が行われている [5], [6]。

Saliency Map とは、ニューラルネットワークに与えられた画像での勾配を計算したものである [5]。クラス  $c \in C$  に画像分類された画像  $x_0$  に対して、 $S_c(x_0)$  をニューラルネットワークの出力として得られる、画像が  $c$  である確率とする。ここで 1 次のテイラー展開を用いることで、 $x_0$  で、 $S_c(x_0)$  を次のように近似する  $w, b$  を求めることができる。

$$S_c(x_0) \approx w_c^T x + b$$

ここで  $w$  は  $S_c$  の  $x_0$  での勾配である。

$$w = \left. \frac{\partial S_c}{\partial x} \right|_{x_0}$$

得られる  $w$  は、画像がグレースケールの場合にはピクセルと一対一対応するが、画像が RGB 画像などのマルチチャンネルの場合には、 $w$  とピクセルは一対一対応しない。勾配  $w$  をニューラルネットワークに与えられた元画像  $x$  と比較して見るために、論文 [5] の著者らは、 $w$  のそれぞれのピクセルでの最大値を取ることで元画像と同じ大きさのサイズにした。

Saliency Map を用いてニューラルネットワークを可視化する手法は、この手法を適用するための特別なラベルなどを訓練時に必要としないこと、また勾配の計算も 1 度 back-propagation を行えばよいため高速に行うことができることが利点である。

### 2.3.2 SmoothGrad

純粋な勾配を用いる Saliency Map では、人間が物事を分類する際には注目しないような点にも大きな値がつくことがあり、人間にはノイズが入っているように見える結果が得られることがある [6]。

一般にある画像にわずかなノイズを加えても人間には殆ど判別不可能である。しかし勾配は、画像にノイズが加えられることで、ニューラルネットワークによって出力される画像分類のラベルも変わらない状況であっても、大きく変わることがある [6]。

SmoothGrad という手法ではこの問題に注目し、ニューラルネットワークに与えられる入力に対してガウシアンノイズを加えることで、この問題の解決を目指した。具体的には、ニューラルネットワークに与えられる画像を  $x$ 、ニューラルネットワークの出力である画像がクラス  $c$  である確率を  $S_c(x)$  とし、 $M_c(x) := \partial S_c(x) / \partial x$  としたとき、

$$\hat{M}_c(x) := \frac{1}{n} \sum_1^n M_c(x + \mathcal{N}(0, \sigma^2))$$

を計算し、可視化の際にこれを用いた。ここで、 $\mathcal{N}(0, \sigma^2)$  は平均 0、分散  $\sigma$  のガウス分布に従ってサンプリングされる値である。

このようにすることで、入力画像近傍で平滑化された値が得られるため、人間が見た際に Saliency Map よりもノイズが少ない結果が得られると期待できる [6]。

## 3. 将棋プログラムへのニューラルネットワークの顕著性抽出手法の適用

画像認識を行うニューラルネットワークに対して提案された、純粋な勾配を用いる Saliency Map や SmoothGrad は、学習時に特別な条件を要求しない。そのため、将棋プ

表 1: ニューラルネットワークの入力に掛ける定数

**Table 1** Constant values multiplied by the neural network input

持ち駒の種類	定数値	持ちうる最大枚数
歩	0.055	18
香	0.25	4
桂	0.25	4
銀	0.25	4
角	0.5	2
飛	0.5	2
金	0.25	4

ログラムに用いられているニューラルネットワークにも簡単に適用することができる。本稿ではそれらの手法を将棋プログラムに用いた結果を示す。

### 3.1 本稿で用いたニューラルネットワーク

本稿では、論文 [7] で用いられたニューラルネットワークと殆ど同じものを用いた。違いは、各層のフィルターの数が 192 であり、Residual Block の最後に Pooling 層があることである。具体的な構造を以下に記述する。

論文 [7] で用いられているニューラルネットワークは、Generative Adversarial Networks (GAN) [8] の発想を用いたニューラルネットワークである。本稿では、今回の実験に関係する部分のみを説明する。ニューラルネットワークの構造は基本的に AlphaZero [3] に従っている。ニューラルネットワークの入力には、43 チャンネル、9×9 ピクセルが与えられる。このうち 28 チャンネルが盤上の駒、14 チャンネルが持ち駒、1 チャンネルが手番を表す。盤上の駒を表すチャンネルでは、それぞれの駒の種類で、駒がある場所のピクセルが 1、それ以外の場所が 0 と設定される。持ち駒を表すチャンネルでは、持ち駒の枚数と定数を掛けた値が、9×9 全てのピクセルに設定される。この定数は、それぞれの持ち駒を最大枚数を持った場合に、入力に設定される値が 1 となるように調整する定数であり、具体的な値は表 1 である。この入力には、はじめに 16 個の Residual Block [9] で処理される。それぞれの Residual Block は、次のような構成である。

- (1) 192 フィルターの畳み込み層
- (2) ReLU
- (3) 192 フィルターの畳み込み層
- (4) ReLU
- (5) 入力との足し合わせ

16 個の Residual Block で処理された値は、次に 2×2 の Global Average Pooling [10] で処理され、その後 192 ノードの全結合層により 1 つのスカラール値を出力する。このスカラール値は現在の手番から見た局面の評価値を表す。本稿では今後、この値を Value と呼ぶ。

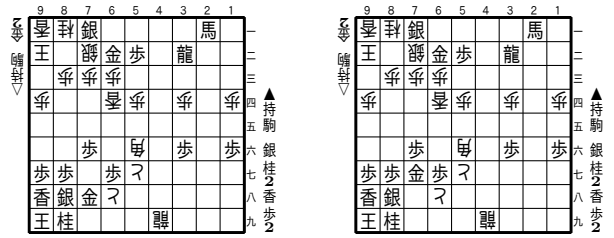


図 1: 先手番。少しの違いが対局結果に影響する例。左の局面と右の局面の違いは、7八の金と7七の金。左の局面の Apery の評価値は -2703。右の局面の Apery の評価値は 2612。思考時間は 1 分。

**Fig. 1** First player to move. An example where game result is affected by a little difference in the position. The difference between the left position and the right position is the gold piece at 7h and at 7g. The evaluated value by Apery of the left position was -2703. The evaluated value by Apery of the right position was 2612. Apery used one minute to evaluate these positions.

このニューラルネットワークは将棋の次の一問題集である文献 [11] において 216 問中 65 問正解した。このニューラルネットワークで次の一手を選択する際には、現在の局面で選ぶことができるすべての合法手の中で、最も評価値が良いものを選んだ。正答率を比較するため、第 28 回世界コンピュータ選手権のバージョンの Apery\*2 の探索深さを固定したものを用いた。Apery は探索深さ 1 で 57 問、探索深さ 2 で 74 問正解した。そのため、今回実験に用いたニューラルネットワークは、Apery の depth 2 より少々弱い程度の強さであると考えられる。

### 3.2 顕著性抽出手法の適用

将棋プログラムに用いられているニューラルネットワークにも、特別な処理を必要とせずに、勾配を計算して Saliency Map を作成したり、SmoothGrad を適用することができる。

SmoothGrad は入力画像にノイズを加え顕著性抽出を行う手法である。そのため将棋に適用すると、ニューラルネットワークに例えばある升に歩が 0.2 枚あるような局面が与えられる。画像では、元の画像にわずかにノイズを加えた画像でも、少なくとも人間には同じ画像に見えることが多い。一方で将棋では、歩が 0.2 枚追加されたような局面では、二歩のような局面になる可能性もある。また、歩を置くことで大駒の利きを遮ってしまう可能性もある。さらに駒の位置が少し違うことで対局結果が変わることもある。例えば図 1 はその例である。

画像認識では、SmoothGrad は純粋な勾配を用いる Saliency Map の改良手法であったが、将棋においては上記のような問題があるため、本稿では純粋な勾配を用いた Saliency Map と SmoothGrad のどちらが良いかを実験

\*2 <https://github.com/HiraokaTakuya/apery>

した。

次に、得られた勾配の可視化をどのように行うかという問題について議論する。論文 [5] ではマルチチャンネルの画像の勾配を、それぞれのピクセルでの絶対値の最大値の値を使用して可視化が行われた。このようにすることで、元の画像と同じ大きさにすることができ、得られた勾配を元の画像と照らし合わせることができる。しかし将棋では、盤上の駒と持ち駒は両方ともニューラルネットワークでは  $9 \times 9$  の画像として与えられているため、得られる勾配をどのように表示するかは簡単ではない。そこで本稿では、盤上の駒と持ち駒の顕著性の可視化を分けて行った。また盤上の駒の顕著性の可視化を行う際にも、いくつかの手法が考えられるため、それらの手法を比較した。

### 3.3 他の手法との比較

ニューラルネットワークを用いた顕著性抽出手法では、勾配を1度計算するだけで良いため、計算に時間がかからないことが利点として挙げられる。論文 [1] では将棋プログラムの探索結果を用いて形勢判断を行っていたため、より正確な判断を得るために深い探索が必要であることが指摘されている。

将棋において、他の駒の重要度を可視化する手法として、それぞれの駒を取り除いた際の評価値の際を用いるといった手法が考えられる。この手法によっても、与えられた局面での重要な駒を判定することはできると考えられるが、現時点では、そのような手法と本手法の特性は明らかでなく、両方を研究する価値がある。

現在主流である三駒関係を利用した評価関数にも顕著性抽出手法を適用することは可能であるが、得られる結果が三駒関係の勾配であり、結果の可視化は単純ではない。またそれらの評価関数は探索の末端で利用されているため、探索の結果を用いずに勾配のみを利用して将棋プログラムの判断の根拠を説明するのは難しいと考えられる。

顕著性抽出手法では駒のない升に対しても勾配が計算されるため、盤上の駒のみに対する情報より多くの情報が得られると考えられる。一方で、与えられた局面での勾配を計算しているため、多くの局面を評価する探索を有効に利用するのは難しいと考えられる。

## 4. 実験

### 4.1 顕著性抽出の例

純粋な勾配を用いた Saliency Map や、SmoothGrad を将棋プログラムに用いられているニューラルネットワークに適用することで得られた結果を報告する。SmoothGrad を適用する際には、パラメータとして  $n = 120, \sigma = 0.15$  を用いた。

図 2, 図 3 は盤上の駒に関して Saliency Map 及び SmoothGrad を適用した例である。図 2 の局面では、純粋

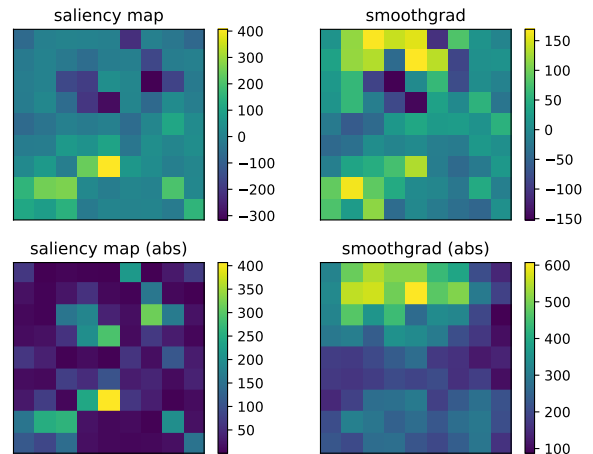
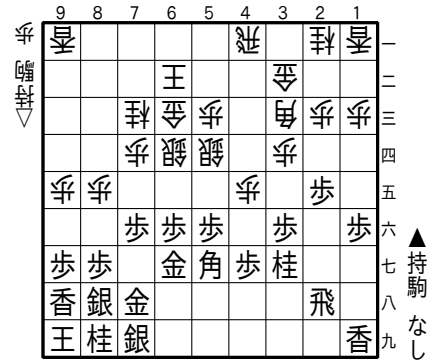


図 2: Saliency Map 及び SmoothGrad の適用例

文献 [12] の第 1 局から引用。先手番。左の列は純粋な勾配を用いた Saliency Map の結果、右の列は SmoothGrad を用いた結果である。上の行は各ピクセルで、各チャンネルの勾配を足し合わせた結果、下の行は各ピクセルで、各チャンネルの勾配の絶対値を足し合わせた結果である。

**Fig. 2** An example of Saliency Map and SmoothGrad. The left column is the result of Saliency Map, and the right column is the result of SmoothGrad. The upper row is the result of summation of gradients at each pixel. The lower row is the result of summation of absolute gradients at each pixel.

な勾配を用いた Saliency Map では 5 七の角や 8 八の銀、3 三の角や 5 四の銀に注目していることがわかる。一方で SmoothGrad では、特定の駒ではなくある程度の領域に注目していることがわかる。

SmoothGrad では、入力にノイズを加えるため、将棋においては、例えばある升に歩が 0.2 枚ある、というような入力がニューラルネットワークに対して与えられる可能性がある。そのような状況では大駒の利きなどを遮るなどで、本来考えるべき元局面と大きく異なる局面についての入力を処理している可能性がある。また、既に歩がある列に歩を加えた二歩の局面など、合法でない局面を評価している可能性もある。一方で、SmoothGrad を用いることで、少

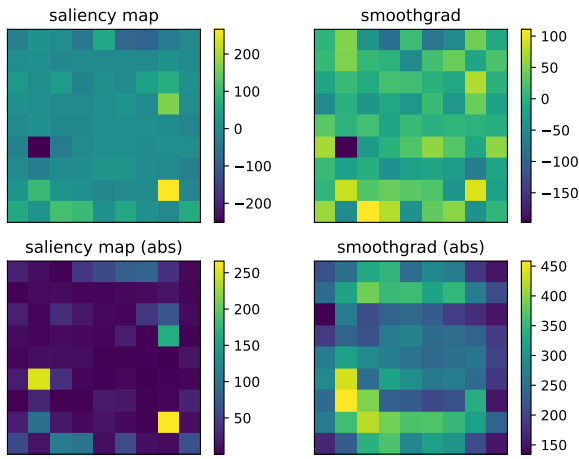
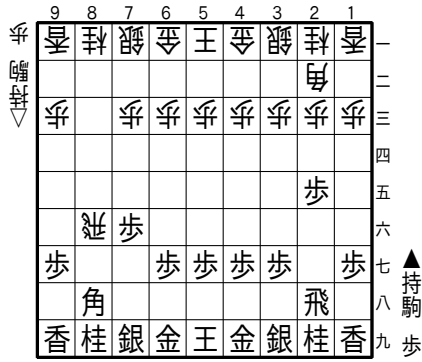


図 3: Saliency Map 及び SmoothGrad の適用例  
文献 [11] の第 1 問目から引用。先手番。▲ 2 四歩 △ 同歩 ▲ 2 三歩  
が予想される進行。2 五の歩と 2 八の飛があることでこの進行ができる  
ため、その 2 つの駒が大事であると考えられる。Saliency Map では  
この 2 つの駒に注目していることがわかる。加えて、8 六の飛にも  
注目していることがわかる。

Fig. 3 An example of Saliency Map and SmoothGrad. This position was taken from book [11]. First player to move. The expected progression is P2e-2d, P2c-2d, P\*2c. This progression is supported by the fact that there are a pawn at 2e and a rook at 2h, so these pieces are important. Saliency Map highlighted these pieces and also highlighted a rook at 8f.

し変更が加えられた局面でも、まだ重要である駒を指摘できている可能性がある。

この 2 つの手法の違いを比較するため、それぞれの手法が出力する最も重要な駒を取り除いたときの、ニューラルネットワークの出力の変化を計測した。

#### 4.2 駒を取り除いた際の Value の変化

対象を盤上の駒に限り、純粋な勾配を用いた Saliency Map と SmoothGrad のどちらが重要な駒を指摘できるかどうかを比較した。実験は、2 つの手法で得られる顕著性抽出手法によって指摘される最も重要な駒を取り除いたと

表 2: それぞれの手法で、最も重要と指摘された駒を取り除いたときの Value の変化の平均

Table 2 The mean value of the difference obtained by removing the most important piece pointed out with each method

手法	評価値の変化の平均
Saliency Map	83.6
SmoothGrad	66.7

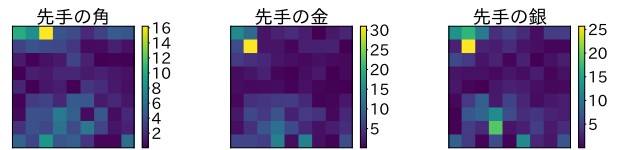
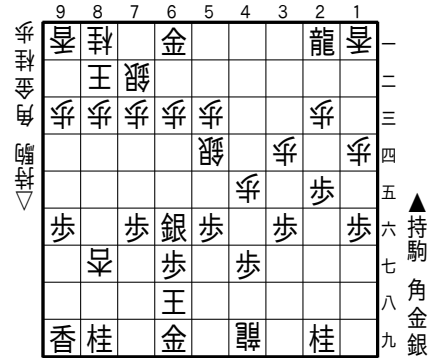


図 4: 持ち駒を打ち、相手玉を詰ませる局面の例

Fig. 4 An example position of checkmate with dropping pieces.

きに、ニューラルネットワークの出力である Value がどの程度変化するかを調べた。ここで、最も重要な駒とは、勾配の中でもっとも絶対値が大きな（駒の種類、升の場所）の組とした。変化量として、駒を取り除く前の評価値と、駒を取り除いた後の評価値の差の絶対値を使用した。変化量が大きい駒は、その駒が取り除かれたことによって評価値が大きく変わる駒であるため、重要な駒であると考えられる。それぞれの手法によって指摘される升に駒がない場合には、Value の変化は 0 であるとして扱った。実験には、floodgate の 2016 年の棋譜から無作為に選んだ 100 棋譜用い、局面の重複ありでその 100 棋譜に現れるすべての局面を対象とした。結果を表 2 に示す。この結果から、純粋な勾配を用いた Saliency Map の方が Value の変化の平均が大きかったため、より重要な駒を指摘していると考えられる。

#### 4.3 持ち駒に関する検証

ニューラルネットワークの入力には、持ち駒も与えられる。この持ち駒の入力に対してどのような勾配が得られるか調べるため、人為的に作成した詰みのある局面を用いて

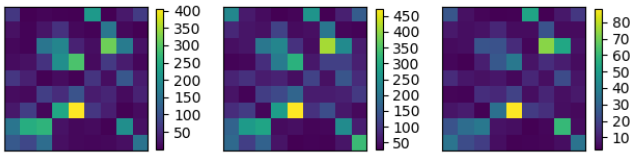


図 5: 可視化手法の比較. 左はそれぞれの升での勾配の値の合計の絶対値, 中央はそれぞれの升での勾配の絶対値の合計, 右はそれぞれの升での勾配の絶対値の最大値である.

**Fig. 5** A comparison for visualization methods. The left one is drawn with summation of gradients at each pixel, the middle one is drawn with summation of absolute gradients at each pixel, and the right one is drawn with max value of gradients at each pixel.

表 3: それぞれの手法によって得られる結果間の L2 距離

**Table 3** The L2 distance for results of each method.

比較する手法		L2 距離
和の絶対値	絶対値の和	0.844
和の絶対値	最大値	0.782
絶対値の和	最大値	0.763
和の絶対値	全て 0	2.046
絶対値の和	全て 0	2.322
最大値	全て 0	2.032

勾配を調べた. 結果を図 4 に示す.

図 4 の局面は先手が▲7一角と打つことで後手玉を詰ませることができる. 持ち駒は, ニューラルネットワークの入力として与えられるときに, それぞれの駒の種類で  $9 \times 9$  のピクセル全てが同じ値の入力として与えられるが, 図 4 の結果からどこに打つべきかが出力層まで伝播していると考えられる.

また, 勾配の値に関して, 角と銀の勾配の値よりも金の勾配の値の方が大きいのは, 図 4 の局面では角か銀どちらか一方があれば相手玉を詰ませることができるが, 金にはそのような代わりの駒がないことが原因と考えられる.

#### 4.4 可視化手法についての検討

顕著性抽出手法によって得られる勾配は, 入力と同じサイズであるため, 人間がニューラルネットワークの判断根拠を理解するために直接その値を参考にするには難しい. 具体的には, 本稿で用いたニューラルネットワークの場合,  $43 \times 9 \times 9$  個のスカラーが得られる.

論文 [5] では, RGB の画像に対して, 顕著性抽出手法の結果を可視化するために, それぞれのピクセルの絶対値の最大値を用い, 元の画像と同じ大きさの画像として可視化を行った. SmoothGrad でも絶対値を用いたほうがよりはっきりした画像が得られると報告されている [6]. これは例えば, ボールを分類するようなタスクで, 背景が明るくボールが暗いと勾配は負となるが, ボールが明るく背景が暗いと勾配は正になる. しかし, どちらの画像でも注目すべき部分は同じであるはずのため, 絶対値を用いた方が良いと考察されている [6].

将棋においては, 得られた勾配が負であるということは, その入力の値が現在の値より小さいと, 評価値が大きくなると考えられる. 逆に正であれば, その入力の値が現在の値より大きいと, 評価値が大きくなると考えられる. そのため, ある升での勾配を足し合わせることで, 先手にとって良い駒と, 後手にとって良い駒などが打ち消し合う, または強め合うことが考えられる. 一方で, 両方の手番にとって, 良い悪いともに影響があるということは, 重要な駒であるとも考えられる.

これらの可視化のための手法の違いを調べるため, 勾配をそれぞれの升での和の絶対値を用いる手法, それぞれの升での絶対値の和を用いる手法, そして論文 [5] で用いられたそれぞれの升での絶対値の最大値を用いる手法を, 図 2 の局面で比較した. なお, ここでは盤上の駒のみを対象とした. 結果を図 5 に示す. この結果から, 少なくとも図 2 の局面では殆ど同じ結果が得られることがわかった.

様々な局面でも手法による違いが小さいのか調べるために, 先述した無作為に選んだ 2016 年の floodgate の棋譜 100 局に現れる局面で, それぞれの手法を比較した. それぞれの手法によって得られる  $9 \times 9$  の値のスケールを  $[0, 1]$  に正規化し, L2 距離を測定した. 正規化には, 正規化前のベクトルを  $\mathbf{x}$  としたとき,  $x_i = (x_i - \min \mathbf{x}) / (\max \mathbf{x} - \min \mathbf{x})$  を用いた. 比較のため, すべて 0 であるベクトルとの距離も測定した. 結果を表 3 に示す. この結果から, どの手法も比較的似た結果が得られ, また絶対値の和を取る手法と最大値を用いる手法は特に似た結果が得られると考えられる.

#### 4.5 一局を通した顕著性マップの遷移

次に, 一局を通して, 純粋な勾配を用いた Saliency Map を適用した. 対象とした棋譜は 1978 年 11 月. 東和男 対 谷川浩司の対局である. また, Saliency Map は 16 手目,



21 手目, 27 手目, 35 手目に適用した. 結果を図 6 に示す. 可視化の際には, それぞれの升での勾配の絶対値の和を用いた.

この結果から局面によって勾配の最大値が大きく変わることがわかる. 勾配が大きいほど, ニューラルネットワークの出力への影響が大きいと考えられる. 図 6 で, 4 行目の 35 手目の局面で, 7 六の馬に大きな勾配がついているのは, 現局面が△4 九馬 ▲同玉 △4 八金 ▲同玉 △3 八飛 ▲5 九玉 △5 八銀の詰めろになっているためと考えられる.

### 5. まとめ

本稿では画像認識を行うニューラルネットワークに対して考案された顕著性抽出手法を, 将棋プログラムに用いられているニューラルネットワークに適用した.

画像認識を行うニューラルネットワークに対する顕著性抽出手法として, 純粋な勾配を用いる Saliency Map と, その改良手法である SmoothGrad がある. 本稿ではそれらの手法を将棋に用いて, ニューラルネットワークの出力である Value の変動を指標として, どちらの手法がより重要な駒を指摘出来ているかを比較した. その結果, 純粋な勾配を用いた Saliency Map のほうが, この指標のもとでは重要な駒を指摘出来ていることがわかった.

得られた勾配を, どのように人間にわかりやすく表示するかも難しい問題である. 本稿では, 3 つの手法 (それぞれの升での勾配の和の絶対値を用いる手法, それぞれの升で勾配での絶対値の和を用いる手法, それぞれの升での勾配の絶対値の最大値を用いる手法) を比較し, 勾配の絶対値の和を用いる手法と, 勾配の絶対値の最大値を用いる手法で特に似た結果が得られることを示した.

本稿では, 得られた勾配の値の大きさを有効に利用できていない. 勾配の値の大きさを有効に利用し, ニューラルネットワークの可視化をより人間にわかり易くすることが今後の課題である.

### 謝辞

本稿を執筆するにあたりご協力して下さった万 山川氏, 朱 捍華氏に感謝いたします.

この研究の一部は, JSPS 科研費 16H02927 と JST さきがけの支援を受けています.

### 参考文献

[1] 金子知適: コンピュータ将棋を用いた棋譜の自動解説と評価, 情報処理学会論文誌, Vol. 53, No. 11, pp. 2525-2532 (2012).  
 [2] 亀甲博貴, 森 信介, 鶴岡慶雅: 実現確率に基づく解説すべき指し手の推定, GPW2016 論文集, pp. 28-35 (2016).  
 [3] Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Ku-

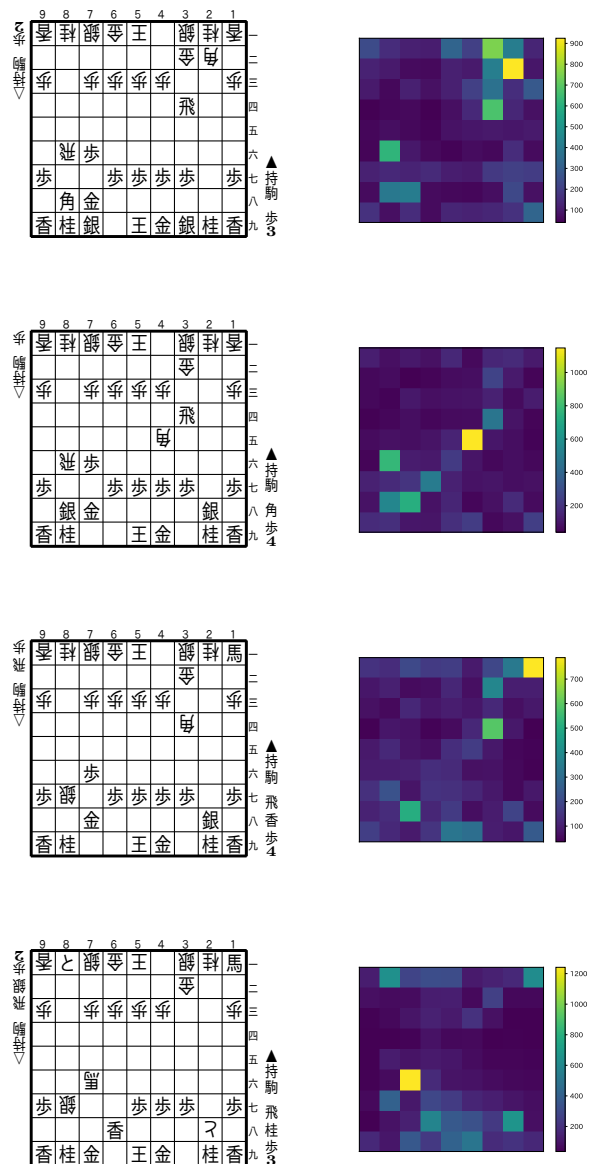


図 6: 1978 年 11 月. 東和男 対 谷川浩司の対局. 36 手で終局. 1 行目は 16 手目の局面. 2 行目は 21 手目の局面. 3 行目は 27 手目の局面. 4 行目は 35 手目の局面.

Fig. 6 A game played by Azuma Kazuo and Tanigawa Koji at November, 1978. The game ended in 36 plies. First row represents the position at 16th move. Second row represents the position at 21th move. Third row represents the position at 27th move. Fourth row represents the position at 35th move.

maran, D., Graepel, T., Lillicrap, T. P., Simonyan, K. and Hassabis, D.: Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm, *CoRR*, Vol. abs/1712.01815 (online), available from <http://arxiv.org/abs/1712.01815> (2017).

[4] 金子知適, 田中哲朗, 山口和紀, 川合 慧: 駒の関係性を利用した将棋の評価関数の学習, 情報処理学会論文誌, Vol. 48, No. 11, pp. 3438-3445 (オンライン), 入手先 <https://ci.nii.ac.jp/naid/110006530805/> (2007).  
 [5] Simonyan, K., Vedaldi, A. and Zisserman, A.:

- Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps, *CoRR*, Vol. abs/1312.6034 (online), available from <http://arxiv.org/abs/1312.6034> (2013).
- [6] Smilkov, D., Thorat, N., Kim, B., Viégas, F. B. and Wattenberg, M.: SmoothGrad: removing noise by adding noise, *CoRR*, Vol. abs/1706.03825 (online), available from <http://arxiv.org/abs/1706.03825> (2017).
- [7] Wan, S. and Kaneko, T.: Building Evaluation Functions for Chess and Shogi with Uniformity Regularization Networks, *IEEE Conference on Computational Intelligence and Games, CIG 2018* (2018).
- [8] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C. and Bengio, Y.: Generative Adversarial Networks, *CoRR*, Vol. abs/1406.2661 (online), available from <http://arxiv.org/abs/1406.2661> (2014).
- [9] He, K., Zhang, X., Ren, S. and Sun, J.: Deep Residual Learning for Image Recognition, *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, IEEE Computer Society, pp. 770–778 (online), DOI: 10.1109/CVPR.2016.90 (2016).
- [10] Lin, M., Chen, Q. and Yan, S.: Network In Network, *CoRR*, Vol. abs/1312.4400 (online), available from <http://arxiv.org/abs/1312.4400> (2013).
- [11] 日本将棋連盟書籍：ラクラク次の一手：基本手筋集，メキメキ将棋上達本，日本将棋連盟 (2002).
- [12] 佐藤康光：佐藤康光の戦いの絶対感覚，最強将棋塾 246，河出書房新社 (2000).