情報処理学会論文誌　数理モデル化と応用　Vol.12　No.2　20–33 (July 2019)

Deep Neural Networkのモデル逆解析による識別根拠可視化技術

柿下 容弓[1,a)]　服部 英春[1]

受付日 2018年8月21日，再受付日 2018年12月7日 / 2019年2月20日，
採録日 2019年3月5日

概要：近年，Deep Neural Network（DNN）を用いたメディア処理技術がさかんに研究されている．DNNはConvolutional Neural Network（CNN）や，Full Connection層等，複数種類の層を多層化することで複雑な関数表現を実現している．その反面，識別根拠や識別理由を人間が理解することが難しいという課題があり，誤識別原因の調査や識別精度の向上に多くの労力を要している．この課題を解決するために，DNNの可視化に関する技術が提案されているが，識別器の構成が制限される，識別根拠の解像度が低いといった課題がある．本論文では各層において出力に対する入力の寄与率を算出（モデル逆解析）することで，識別器構成に依存せず，高解像度の識別根拠を可視化する手法を提案する．

キーワード：DNN，Deep Learning，CNN，可視化，説明可能性

Classification Reasons Visualization of Deep Neural Network Using Model Inverse Analysis

Yasuki Kakishita[1,a)]　Hideharu Hattori[1]

Received: August 21, 2018, Revised: December 7, 2018/February 20, 2019,
Accepted: March 5, 2019

Abstract: Recently, media processing technologies using deep neural network are actively studied. Deep neural network realizes complicated function by multilayering several kinds of layers such as convolutional neural network, full connection layer, etc. On the other hand, deep neural network has a problem that is difficult to understand classification reasons by human. To solve this problem, visualization methods of deep neural network has been proposed. However, in these methods, the structure of classifier is limited or visualization of classification reasons is shown with low resolution. In this paper, we propose a visualization method of identification reasons with high resolution and without restriction on classifier structure by calculating the contribution ratio of inputs to output (model inverse analysis) in each layer.

Keywords: DNN, Deep Learning, CNN, visualization, explainable

1. 緒言

近年，Deep Neural Network（DNN）を用いたメディア処理技術がさかんに研究されている．たとえばコンピュータビジョンの分野においては，特にConvolutional Neural Network（CNN）[1]を用いた手法が画像認識や物体検出，セグメンテーション等，様々なタスクで従来手法を大きく上回る成績を収めている．DNNによる識別器は，CNNやFull connect層等の複数種類の層を多層化することで複雑な識別関数を表現し，高い識別性能を実現している．しかしながら，識別器の構造が非常に複雑であり，識別結果に対する根拠を人間が理解することが難しいという課題がある．そのため，誤識別原因の特定や，識別精度向上のための対策検討に多大な労力を要する．そこで我々は，入力画像内のどの領域が，識別結果の根拠となったのかを可視化するために，寄与率解析による識別根拠可視化手法

[1] 株式会社日立製作所研究開発グループ
Research & Development Group, Hitachi Ltd., Kokubunji, Tokyo 185–8601, Japan
[a)] yasuki.kakishita.rw@hitachi.com

(Contribution Analysis Visualization．以下 CAV と呼ぶ) を提案する．従来も DNN の識別根拠可視化に関する研究はあったが，識別器の構造が制限される，識別根拠を高解像度かつ正確に表現できないといった課題がある．提案手法は DNN の各層において，入力が出力に対してどの程度寄与したのかを最終層から遡って解析することで，構造上の制限が少なく，高解像度かつ正確な識別根拠を可視化する．

本論文では 2 章で従来手法について述べた後，3 章で提案手法を説明する．その後，4 章で学習済みの画像識別器を用いた実験結果を報告し，5 章で提案手法について考察する．

2. 従来手法

Neural Network の可視化に関する従来手法について述べる．CNN による識別において重要度の高いピクセルを可視化する手法の一例として Guided Backpropagation [2] や Deconvolution [3], [4] を用いた手法がある．これらの手法は特定のユニットの活性を逆演算により画像化し，可視化する手法であるが，識別根拠可視化のために用いた場合には，異なるクラス間での差が表れにくく，識別根拠を示すことが難しいと報告されている [5].

また，ネットワーク内の特定のユニット出力を最大化する入力パターンをシミュレートする手法がある [6], [7]．これらの手法は，特定の識別結果を得るための入力パターンを生成できるが，実際の入力画像のどの領域を識別の根拠としたかを可視化するものではない．

提案手法に最も関連深い従来手法として，CAM [8] や Grad-CAM [5] がある．CAM は CNN の最終層が出力した各特徴量が，どの程度最終出力に影響するかを可視化するものであり，提案手法の目的に近い．ただし，CAM は CNN の最終層に対して Global Average Pooling [9] を適用しなくてはいけないという，識別器構造上の制約がある．Grad-CAM はこれを一般化した手法であり，Global Average Pooling を適用しないネットワークに対しても，中間層の特徴量がどの程度最終出力に影響するかを可視化できる．Grad-CAM は原理的には任意の中間層における特徴量を用いた可視化が可能だが，最終層の特徴量を用いた可視化が推奨されている．これは入力に近い層の特徴量を用いて可視化を行うと，識別根拠の局在性が低下し，対象物とは無関係の領域に誤反応する等，正確に識別根拠を表現できない場合があるためである [5]．しかし，識別根拠の解像度は，特徴量マップの解像度に依存する．一般に，CNN を用いた識別器は Pooling 等のダウンサンプリング処理を複数回適用するため，CNN 最終層の特徴量は低解像度である場合が多い．識別器構成によっては CNN 最終層の解像度は 1×1 になる場合もある．このような場合，Grad-CAM では高解像度かつ正確な識別根拠を得ることは難しい．

提案手法の目的は，これらの課題を解決し，識別器の構造上の制約がなく，高い解像度で特定のクラスに関する識別根拠を正確に可視化することである．

以降，各従来手法について詳細に説明する．

2.1 DeconvNet, SaliNet, DeSaliNet

Deconvolution 処理を用いた代表的な従来手法として DecoonvNet [3] がある．また，DeconvNet に類する手法として SaliNet や DeSaliNet がある [4]．これらは各層において逆方向の演算を定義して，あるユニットの出力を入力層の信号として復元する手法である．

たとえば Full connection 層の場合，順方向の演算を式 (1) とすると，逆方向の演算は式 (2) で定義される．ここで，y および x は順方向演算の出力および入力，W は重み，b はバイアス，W^T は重みの転置行列，$\hat{y}$ は後層からの逆方向演算結果，$\hat{x}$ は Full connection 層が出力する逆方向演算結果を表す．Convolution 層の場合も，W が畳み込み演算子となる以外は，同様に順方向および逆方向の演算を行う．

$$y = Wx + b \tag{1}$$

$$\hat{x} = W^T \hat{y} \tag{2}$$

Max Pooling 層の場合は順方向演算の際に最大値を出力した箇所を記憶して，逆方向の演算時に最大値を出力した箇所にのみ，信号を伝播する．

Activation 層については ReLU を演算対象とする．DeconvNet，SaliNet，DeSaliNet の違いは ReLU の逆方向演算方法であり，DeconvNet は式 (3)，SaliNet は式 (4)，DeSaliNet は式 (5) に基づいて逆演算を行う．

$$\hat{x} = ReLU(\hat{y}) \tag{3}$$

$$\hat{x} = \hat{y} \times ReLU(x) \tag{4}$$

$$\hat{x} = \hat{y} \times ReLU(x) \times ReLU(\hat{y}) \tag{5}$$

これらの手法はネットワーク内のあるユニットを活性化する入力画像のパターンを可視化可能だが，識別根拠可視化のために用いた場合には，異なるクラス間での識別根拠の差が表れにくいことが報告されている [5]．我々は，この原因が Full connection 層および Convolution 層の逆方向演算にあると考える．

DeconvNet 等を用いて識別結果に対する特定のクラスの逆演算結果を得る場合を考える．最終層に入力する逆方向演算の入力 $\hat{y}$ は，各クラスの確率を格納した配列であり，たとえば特定のクラスに対する反応を可視化するためには，特定のクラス以外の確率を 0 にした配列を最終層に入力し，入力層まで逆方向の演算を行う方法が考えられる．

しかし，式 (2) から分かるとおり，Full connection 層や Convolution 層における逆方向の演算結果 $\hat{x}$ は，重み W と

出力 $\hat{y}$ のみに依存しており，順方向演算における入力 x，すなわち特徴量の分布は加味されない．たとえば入力画像 Ia と Ib という異なる 2 つの画像を入力した結果，最終層に入力される特徴量 x の分布には違いがある．しかし，特定のクラスに対する確率が同じ値であったとすると，最終層の逆方向演算結果 $\hat{x}$ はまったく同じ分布となる．

Max Pooling 層や Activation 層においては入力信号の分布情報が反映されるため，全体としては入力画像が反映されているように見えるが，Full connection 層や Convolution 層において上述のとおり，入力信号の分布情報が反映されないため，正確な識別根拠の可視化が難しいと考える．

2.2 Grad-CAM

Grad-CAM [5] は，式 (6) により算出する L^c を識別根拠として可視化する．ここで A^k は Convolution 層の特徴量マップ，a_k^c は式 (7) で算出する A^k の係数，i，j は特徴量マップの水平および垂直方向のインデックス，Z は特徴量マップの水平および垂直方向のサイズを乗算した値，k は特徴量のインデックス，y^c はクラス c の識別スコアを表す．

$$L^c = ReLU\left(\sum_k a_k^c A^k\right) \tag{6}$$

$$a_k^c = \frac{1}{Z}\sum_i\sum_j \frac{\partial y^c}{\partial A_{ij}^k} \tag{7}$$

Grad-CAM ではクラス c の識別スコア y^c の特徴量 A^k に関する勾配が大きい程，y^c に対する A^k の影響が大きいと仮定して可視化を行う．L^c は任意の層の特徴量 A^k を用いて算出可能であるが，低次特徴量を用いた場合，識別根拠の局在性を表現することが難しくなり，対象物と無関係の箇所に識別根拠が表れるといった課題があると報告されている [5]．

我々はこの課題について，Grad-CAM にて低次特徴量を用いた可視化を行う際，高次特徴量における低次特徴量の影響を正確に表現できないことが原因であると考える．特徴量マップ A^k に対する係数 a_k^c はクラス c と特徴量インデックス k にのみ依存する．たとえば，特定の方向のエッジを表す特徴量群を有する低次特徴量を用いて可視化する場合を考えると，あるエッジが対象物に含まれているか，背景に含まれているかにかかわらず，特徴量インデックスが共通であれば識別結果に対する影響度は等しいと評価される．実際にはそのエッジが対象物の一部であるか背景の一部であるかによって，識別スコアに対する影響度は異なるはずである．そのエッジが対象物の一部であるか否かを反映した識別根拠を得るためには高次特徴量の中で低次特徴量がどのように働いたかを解析する必要があると考える．

本論文で提案する CAV では高次特徴量に対する低次特徴量の寄与を伝播していくことにより，低次特徴量を用いた場合でも識別根拠の局在性を維持しつつ，正確に表現することをめざした．

3. 提案手法

3.1 基本的なアイデア

CAV の基本的なアイデアは，Neural Network の各層における出力に対して，各入力値がどの程度の割合で寄与したのかを求めることである．以降，これを寄与率と呼ぶ．説明を簡単にするために，**図 1** に示す，入力ユニット数 5，出力ユニット数 1 の Full connection 層を考える．x_i は入力値，w_i は重み，p は識別結果である．ここで入力ユニット x_4 はつねに 1.0 が入力されるユニットであり，x_4 に対する重み w_4 はバイアス係数の働きをする．また，活性化関数はこの段階では考慮しない．入力値 x_i と出力 p の関係を式 (8)，(9) に示す．

$$\alpha_i = x_i w_i \tag{8}$$

$$p = \sum_i \alpha_i \tag{9}$$

ここで，出力 p の値を母数として，各入力値がどの程度の割合で寄与しているかを式 (10) により算出する．これを出力 p に対する入力 x_i の寄与率 $\beta_{x_i}^p$ とする．出力 $p = 0$ の場合，基本的には寄与率 $\beta_{x_i}^p = 0$ とするが，例外として出力 p がネットワークの最終出力である場合は $p = 1$ として式 (10) を計算する．理由は後述する．

$$\beta_{x_i}^p = \alpha_i / p \tag{10}$$

寄与率 $\beta_{x_i}^p$ が大きい入力値であるほど出力 p に大きく寄与した入力値であり，出力 p が最終出力，すなわち識別スコアである場合は識別根拠となる特徴量を示している．また，寄与率 $\beta_{x_i}^p$ は負の値もとり得，寄与率 $\beta_{x_i}^p$ が負の値の入力値は，出力 p を抑制する特徴量であることを示している．出力 p の値が低い場合は，出力 p を増大する特徴量の不足，または，出力 p を抑制する特徴量の存在が原因として考えられるが，提案手法は寄与率の算出により，その両方の原因を確認可能である．寄与率 $\beta_{x_i}^p$ は入力 x の分布を反映しており，仮に出力 p が同値となる異なる 2 種類の入力に対しても，入力の分布に応じて出力 p に対する各入力値の寄与を算出する．

寄与率 $\beta_{x_i}^p$ はネットワークの最終出力である識別スコアに対する各入力値の寄与率であるため，図 1 のように識別器が Full connection 層 1 層で構成されている場合，寄与率 $\beta_{x_i}^p$ が識別根拠を表している．しかし，多層化した場合，最終層の出力である識別スコア p に対する寄与率を，入力層側に遡って算出する必要がある．次にその方法を説明する．

図 2 に Full connection 層 2 層の例を示す．ここで x_i は Layer 0 への入力値，y_j は Layer 0 の出力値かつ Layer 1 への入力値，p は識別スコア，w_{ji}^l は Layer l の出力ユニッ

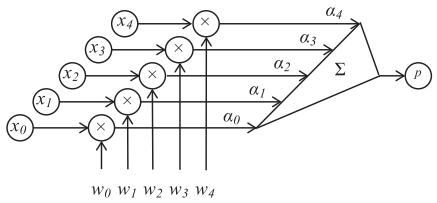

図 1　Full connection 層 1 層の例
Fig. 1　Example of a fully connected layer.

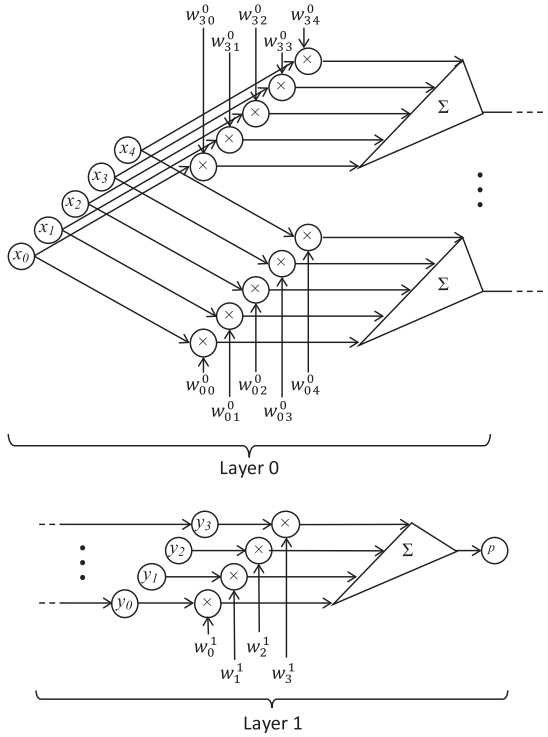

図 2　Full connection 層 2 層の例
Fig. 2　Example of two fully connected layers.

ト j，入力ユニット i に対する重みである．また，図には記載していないが，$\beta^{y_j}_{x_i}$ は Layer 0 の出力 y_j に対する入力 x_i の寄与率，$\beta^{p}_{y_j}$ は識別結果 p に対する Layer 1 の入力 y_j の寄与率を表す．

図 2 中の Layer 1 は図 1 と同様であるため，先述の方法で寄与率 $\beta^{p}_{y_j}$ を算出できる．また，図 2 中の Layer 0 は図 1 の出力ユニットを複数設置した状態と考えることができるため，前述の方法で寄与率 $\beta^{y_j}_{x_i}$ を算出できる．

目的は識別スコア p に対する Layer 0 の入力 x_i の寄与率 $\beta^{p}_{x_i}$ を算出することである．ここで寄与率 $\beta^{y_j}_{x_i}$ は出力 y_j に対する入力 x_i の寄与率を表しており，識別スコア p に対する出力 y_j の寄与率は $\beta^{p}_{y_j}$ であると分かっている．そこで我々は寄与率 $\beta^{y_j}_{x_i}$ に寄与率 $\beta^{p}_{y_j}$ を乗算して，出力 y_j に関して総和をとることで，寄与率 $\beta^{p}_{x_i}$ を算出した．寄与率 $\beta^{y_j}_{x_i}$ の算出式を式 (11)，寄与率 $\beta^{p}_{y_j}$ の算出式を式 (12)，$\beta^{p}_{x_i}$ の算出式を式 (13) に示す．先述したが，式 (11) で示した中間層における出力 y_j が 0 の場合は寄与率 $\beta^{y_j}_{x_i} = 0$ とする．これは中間層における出力が 0 の場合には，後層における寄与率も 0 であると考えるためである．一方，式 (12) においてネットワークの最終出力である識別スコア $p = 0$ の場合には $p = 1$ として寄与率を計算する．これは識別スコア p が 0 の場合，寄与率 $\beta^{p}_{y_j}$ がすべて 0 である場合と，寄与率 $\beta^{p}_{y_j}$ 内の正負の寄与が均衡している場合があり，どちらの状況であるのかや，後者の場合はどのような特徴量がどのような寄与率を有するのかを確認することが重要であると考えており，$p = 1$ とすることで寄与率 $\beta^{p}_{y_j}$ の分布を確認可能となるためである．

$$\beta^{y_j}_{x_i} = (x_i w^0_{ji})/y_j \tag{11}$$

$$\beta^{p}_{y_j} = (y_j w^1_j)/p \tag{12}$$

$$\beta^{p}_{x_i} = \sum_j \beta^{p}_{y_j} \beta^{y_j}_{x_i} \tag{13}$$

上記の手法により，最終層の寄与率と最終層の 1 つ前の層における寄与率を乗算することで，最終層の 1 つ前の層についても識別スコアに対する寄与率を計算することができる．もし，入力層側にさらに層を増やした場合は最終層の 1 つ前の寄与率を最終層の寄与率と置き換えることで，同様に識別スコアに対する寄与率を計算することが可能である．すなわち，各層ごとに寄与率を計算することで，識別スコアに対する任意の層の寄与率を求めることが可能である．

次章ではコンピュータビジョンの分野で一般的に使われる Activation 層，Pooling 層，Convolution 層のそれぞれについて寄与率算出手法を説明する．なお，Full connection 層における寄与率算出手法は上述のとおりである．先述の議論から，各層における寄与率算出手法を定義することで，これらの層をどのように組み合わせても識別根拠を可視化することが可能である．本論文では上記 4 種類の層についてのみ説明するが，提案手法はその他の層であっても寄与率算出方法を定義することで適用可能であり，識別器構造上の制約が少ない手法であると考える．

3.2　寄与率の算出方法

本章では主に画像処理や画像認識の分野で頻繁に使用する層について，寄与率の算出方法を説明する．全体を通して，x_i を層の入力，y_j を層の出力，$\beta^{y_j}_{x_i}$ を出力 y_j に対する入力 x_i の寄与率とする．また，層によっては入出力が 1 次元配列でなく，複数次元配列（たとえばチャンネル，垂直方向位置，水平方向位置の 3 次元配列）の場合もあるが，表記を簡単にするため 1 次元配列として表記する．

3.2.1　Activation 層

Activation 層は入力ユニットに対して ReLU や tanh，sigmoid 等の非線形関数を適用する層である．Activation 層では入力と出力が一対一の関係となるため，出力 y_i に対応する入力 x_i の寄与率 $\beta^{y_j}_{x_i}$ のみ 1.0 となる．式 (14) に寄与率 $\beta^{y_j}_{x_i}$ の算出式を示す．

$$\begin{cases} \beta_{x_i}^{y_j} = 1.0 & (j = i) \\ \beta_{x_i}^{y_j} = 0.0 & (j \neq i) \end{cases} \tag{14}$$

ReLU を用いた場合，入力信号が 0 以下のとき出力が 0 となり，後層における寄与率も 0 となる．よって，ReLU を用いた場合の Activation 層における寄与率算出方法は，SaliNet で用いる式 (4) に相当すると考える．

3.2.2　Pooling 層

Pooling 層は，入力の部分領域に対して最大値や平均値等の演算を行い，その演算結果を出力する層である．Pooling 層の場合は，用いる演算の種類によって寄与率の算出方法が異なる．

演算に最大値を用いる場合（Max Pooling 層），1 つの出力に寄与する入力は部分領域中の最大値のみである．よって出力 y_j に対する入力 x_i の寄与率 $\beta_{x_i}^{y_j}$ は式 (15)，(16) により算出する．ここで R_j は出力 y_j の演算対象となる入力 x_i の部分領域を表す．

$$max_i = argmax(x_i) \quad (i \in R_j) \tag{15}$$

$$\begin{cases} \beta_{x_i}^{y_j} = 1.0 & (i = max_i) \\ \beta_{x_i}^{y_j} = 0.0 & (i \neq max_i) \end{cases} \tag{16}$$

Max Pooling における寄与率算出方法は，DeconvNet や SaliNet 等における Max Pooling の逆演算と同等の処理であると考える．

演算に平均値を用いる場合（Mean Pooling 層），1 つの出力には R_j 内のすべての入力が均等に寄与する．これは，$|R_j|$ を部分領域 R_j に含まれる入力ユニットの数とした場合，各入力に $1/|R_j|$ の重みをかけた後に総和する処理と見なすことができる．よって，寄与率 $\beta_{x_i}^{y_j}$ は式 (17) により算出する．

$$\begin{cases} \beta_{x_i}^{y_j} = \left(x_i \dfrac{1}{|R_j|} \right) / y_j & (i \in R_j) \\ \beta_{x_i}^{y_j} = 0.0 & (\text{other}) \end{cases} \tag{17}$$

3.2.3　Convolution 層

Convolution 層は，入力値に対して重みを畳込み，バイアス項を加算したうえで出力する層である．**図 3** に 1 つの出力値に対する Convolution 処理を示す．図 3 では入力ユニットの部分領域 X を 1 次元に並べ替えて x_0 から x_n，重み W を 1 次元に並べ替えて w_0 から w_n としている．このように並べ替えることで Convolution 層の処理は部分的に Full connection 層と同様の処理を行っていると分かる．よって，出力ユニット y_i に対する入力ユニット x_i の寄与率 $\beta_{x_i}^{y_j}$ は Full connection 層と同様の計算方法で算出できる．寄与率 $\beta_{x_i}^{y_j}$ の算出方法を式 (18) に示す．ただし，出力 $y_j = 0$ の場合は寄与率 $\beta_{x_i}^{y_j} = 0$ とする．

$$\beta_{x_i}^{y_j} = (x_i w_{ji}) / y_j \tag{18}$$

提案手法では Full connection 層や Convolution 層における寄与率 $\beta_{x_i}^{y_j}$ が入力 x の分布を反映している．この点が，DeconvNet 等とは大きく異なる点である．また，出力に対する各入力の寄与を伝播することにより高次特徴量に対する低次特徴量の影響を算出する点が Grad-CAM との大きな違いと考える．

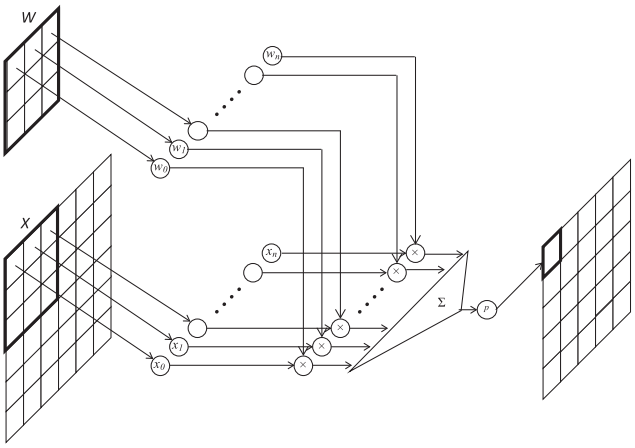

図 3　1 つの出力値に対する Convolution 処理
Fig. 3　Convolution process regarding an output value.

3.3　可視化方法

上記までで説明した手法により，任意の層における識別スコア p に対する寄与率，すなわち識別根拠を算出できる．ここで，画像識別において，入力画像上の領域ごとに識別根拠を可視化する手法を説明する．多くの場合，画像識別器は**図 4** に示すように Convolution 層と Pooling 層を組み合わせて特徴量を抽出した後に，Full connection 層を 1 層以上通過して出力を得る．Activation 層は全体を通して使用する．Convolution 層と Pooling 層により特徴量を抽出する層群を特徴抽出部と呼ぶことにする．

特徴量抽出部に属する層の入出力はチャンネル，垂直方向位置，水平方向位置の次元を持つ 3 次元配列である．ただし層によって垂直，水平方向の解像度に違いがあり，Pooling や Convolution 処理の影響で層が深くなるほど解像度が低くなる傾向がある．また，Full connection 層を通過すると水平垂直方向の識別根拠の情報が消失する．そのため垂直および水平方向の情報を持つのは特徴量抽出部に属する層までである．

そこで，本手法では特徴量抽出部に属する層において垂直，水平領域ごとに上述した寄与率を表示することで，入力画像上に識別根拠を可視化する．特徴量抽出に属する層の入出力は垂直，水平位置以外にチャンネルの次元を有しているが，本論文では，チャンネル方向の寄与率を総和することで，画像上に識別根拠を可視化している．

識別根拠可視化の際，入力層から遠い層の特徴量を使う程，解像度は低いが大局的な傾向を示し，入力層から近い層の特徴量を使うほど，解像度は高いが識別根拠が局所に

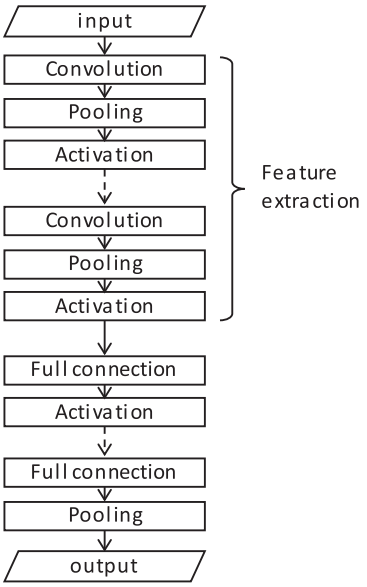

図 4　画像識別器の例

Fig. 4　Example of an image classifier.

分散してしまう傾向がある．そこで，後述する実験結果の章では，いくつかの層の特徴量単体による識別根拠可視化結果を示したうえで，すべての層における識別根拠算出結果の平均値による可視化結果をあわせて示す．すべての層の識別根拠を平均することで，高解像度かつ大局的および局所的な情報の両方を表現する識別根拠の可視化をめざした．

4.　実験結果

4.1　可視化結果

画像識別器として広く知られているモデルである VGG-16 モデル [10] を使用して，従来手法である DeconvNet，SaliNet，DeSaliNet および Grad-CAM と提案手法である CAV の可視化結果の比較を行った．以降，VGG-16 モデルを単に VGG モデルと呼ぶ．VGG モデルは 2014 年に ImageNet Large-Scale Visual Recognition Challenge にて発表されたモデルであり，入力画像を 1,000 種類のオブジェクトカテゴリに分類する識別器である．**図 5** に VGG モデルのネットワーク構成図を示す．

図 6 に実験に用いた入力画像と，VGG モデルによる識別結果を示す．図 6 (a) は原画像，図 6 (b) は原画像から猫の写っている領域をマスクした画像，図 6 (c) は原画像から犬の写っている領域をマスクした画像であり，各画像の右側には VGG モデルによる識別結果の Top 5 を示している．図 6 (b) と図 6 (c) の識別結果から，画像上側の犬を ‘boxer’，下側の猫を ‘tiger cat’ と識別していると考えられる．

図 7 に DeconvNet，SaliNet，DeSaliNet による識別根拠可視化結果を示す．識別根拠算出方法は 2.1 節で説明したように，識別スコアの内，可視化対象とするクラス以外

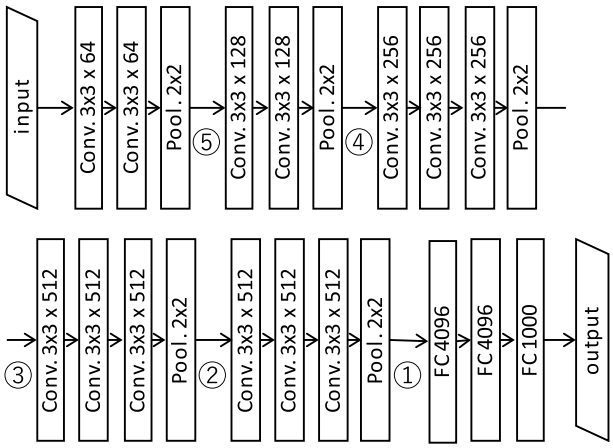

図 5　VGG モデルのネットワーク構成図．①から⑤は識別根拠可視化箇所

Fig. 5　Network structure of VGG model. ① to ⑤ indicate features that are used for classification reasons visualization.

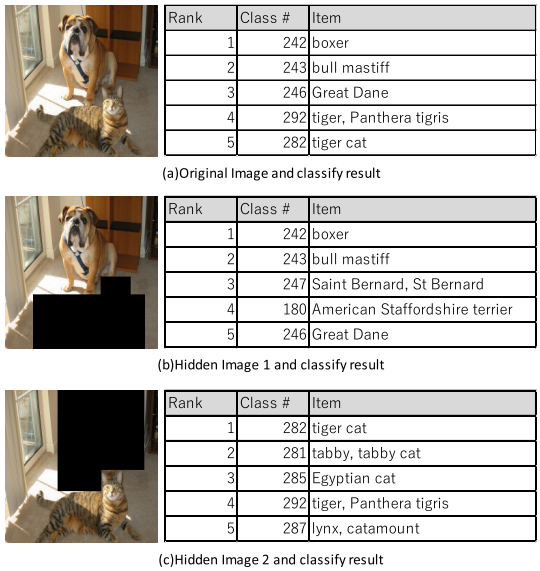

図 6　(a) 原画像 [11] と識別結果，(b) 猫が写っている領域を手動でマスクした画像と識別結果，(c) 犬が写っている領域を手動でマスクした画像と識別結果．各画像は 224 × 224 サイズにリサイズして識別器に入力した．識別結果は上位 5 クラスのみを記載

Fig. 6　(a) represents original image [11] and classification result, (b) represents image with masked cat region and classification result, (c) represents image with masked dog region and classification result. Each input image is resized to 224 × 224 before feeding into the classifier. In each classification results, the top 1 to top 5 accuracy scores are shown.

のスコアを 0 として，最終層から逆演算した．図 7 から，DeconvNet，SaliNet，DeSaliNet による識別根拠可視化結果には，‘boxer’ と ‘tiger cat’ の可視化結果に明確な差がなく，各クラスを区別するための特徴量を判別することは難しい．また，‘boxer’ あるいは ‘tiger cat’ クラスに対する

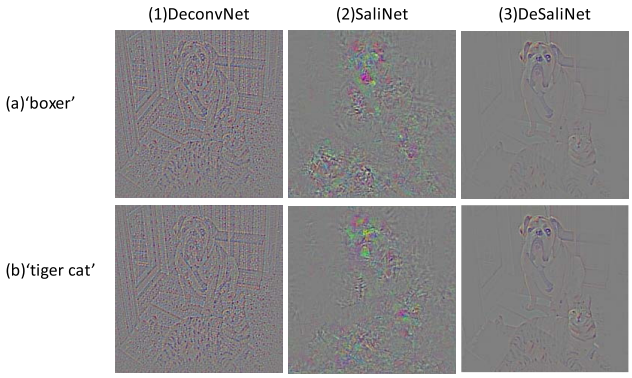

図 7　DeconvNet，SaliNet，DeSaliNet による識別根拠可視化結果．(a) 行は 'boxer'，(b) 行は 'tiger cat' に対する可視化結果を表す．(1) 列は DeconvNet，(2) 列は SaliNet，(3) 列は DeSaliNet による可視化結果を表す

Fig. 7　Result of classification reasons visualization using DeconvNet, SaliNet and DeSaliNet. Row (a) and (b) shows the visualization results of 'boxer' and 'tiger cat', column (1), (2) and (3) shows the visualization results by DeconvNet, SaliNet and DeSaliNet respectively.

識別根拠であるにもかかわらず，どちらも 'boxer' と 'tiger cat' の全体をハイライトする傾向があり，手法によっては背景内のエッジや点までハイライトしている．上記のような傾向から，DeconvNet，SaliNet，DeSaliNet を用いて各クラスの識別根拠を可視化することは難しいと考える．

図 8 に Grad-CAM [5] による可視化結果と提案手法である CAV による可視化結果を示す．図 8 (a) 行と図 8 (c) 行は Grad-CAM による 'boxer' クラスと 'tiger cat' クラスに対する可視化結果を示している．図 8 (b) 行と図 8 (d) 行は CAV による 'boxer' クラスと 'tiger cat' クラスに対する可視化結果であり，ここでは正の寄与率をヒートマップで描画している．図 8 の (1) 列から (5) 列は図 5 の①から⑤における特徴量を使用して識別根拠の可視化を行った結果を表している．(6) 列は全層における識別根拠可視化結果を平均した結果である．

まず，図 8 (1) 列に示した，①の特徴量を用いた可視化結果を比較すると，Grad-CAM および CAV の両方が同様の傾向を示しており，猫は下半身，犬は顔の部分に強い識別根拠を示している．しかし，(a)，(c) 行に示した Grad-CAM による可視化結果は，②から⑤へと進むにつれて，識別根拠の局在性が低下し，対象物以外の箇所への反応が多くなる傾向がある．たとえば (a) 行 (5) 列を見ると，垂直方向のエッジ全体に反応する傾向があり，カーテンや窓枠，棚等にも 'boxer' 内のエッジと同じように反応していることが分かる．一方，(b)，(d) 行に示した提案手法 CAV による可視化結果は②から⑤にかけて識別根拠の局在性を維持しており，どの段階においても対象物内部または周辺に識別根拠が分布しており，かつ，①から⑤へと進むに連れて識別根拠が高解像度化していることが分かる．これによりたとえば (b) 行 (4) 列を見ると，犬の顔の中でも特に目元の領域を識別根拠としていることが分かり，従来よりも詳細に識別根拠を確認可能である．また，図 8 の (d) 行 (4) 列を見ると 'tiger cat' に対する識別根拠を可視化しているにもかかわらず，犬の眉間部分に強い反応が出ていると分かる．この点に関しては次章で考察する．

また，識別根拠を高解像度で可視化できる一方で，(b)，(d) 行 (5) 列のように入力層に近付くにつれて，大局的な傾向を判断しにくくなる傾向がある．これは，提案手法が入力層に近付くにつれて大局的な正負の寄与率を相殺することで，局所的な寄与率を表現しているためと考える．また，低次特徴量の段階では相殺により寄与率の絶対値が小さくなる場合でも，より高次の特徴量においては寄与率の絶対値が大きくなる場合があり，提案手法は識別層から入力層までの各層における寄与率の可視化により，大局から局所にかけた識別根拠を段階的に表現可能な手法であると考える．高次特徴量による寄与率は大局的な傾向を確認しやすいが解像度が低く，低次特徴量による寄与率は解像度が高いが局所的な傾向を表している．そこで高解像度かつ大局的，局所的な傾向を維持した識別根拠を可視化するために，図 8 (6) 列に示すように全層の識別根拠の平均値を可視化した．たとえば図 8 の (b) 行 (6) 列を見ると犬の顔全体が識別根拠であり，特に目元が重要な識別根拠となっていることが分かる．図 8 の (a)，(c) 行 (6) 列に，Grad-CAM による全層の識別根拠の平均値を示しているが，背景部分等の対象物以外の箇所にも強い識別根拠を示す傾向がある．これは低層の特徴量を用いた識別根拠において局在性が低下しているためと考える．

また，図 8 以外の画像を用いた識別根拠可視化実験の結果を報告する．使用した画像を**図 9**，可視化結果を**図 10** に示す．図 8 と同様に，図 10 の (1) 列の結果は Grad-CAM も CAV も同様の傾向であるが，Grad-CAM は入力層に近い層の特徴量を用いた場合（図 10 の (a)，(c)，(e)，(g) 行 (4)，(5) 列），識別根拠の局在性が低下する傾向がある．

たとえば図 10 の (a) 行 (5) 列は 'zebra' に対する⑤の特徴量を用いた Grad-CAM による識別根拠可視化結果であるが，'zebra' の領域だけでなく，背景の木や 'elephant' の領域にも識別根拠が示されている．

一方，CAV は識別根拠の局在性が維持されており，図 10 の (b) 行 (1) 列から (5) 列を見ると，①から⑤に進むにつれて 'zebra' の縞模様に識別根拠が集中していく傾向があると分かる．その他のクラスに対する可視化結果でも同様であり，Grad-CAM と比較して，CAV は局在性を維持したまま識別根拠を高解像度化していることが分かる．また，図 10 の (6) 列に示す全層の識別根拠可視化結果の平均値は大局的および局所的な識別根拠の傾向が同時に表れており，全体的な識別根拠可視化の傾向を確認するために有効であると考える．

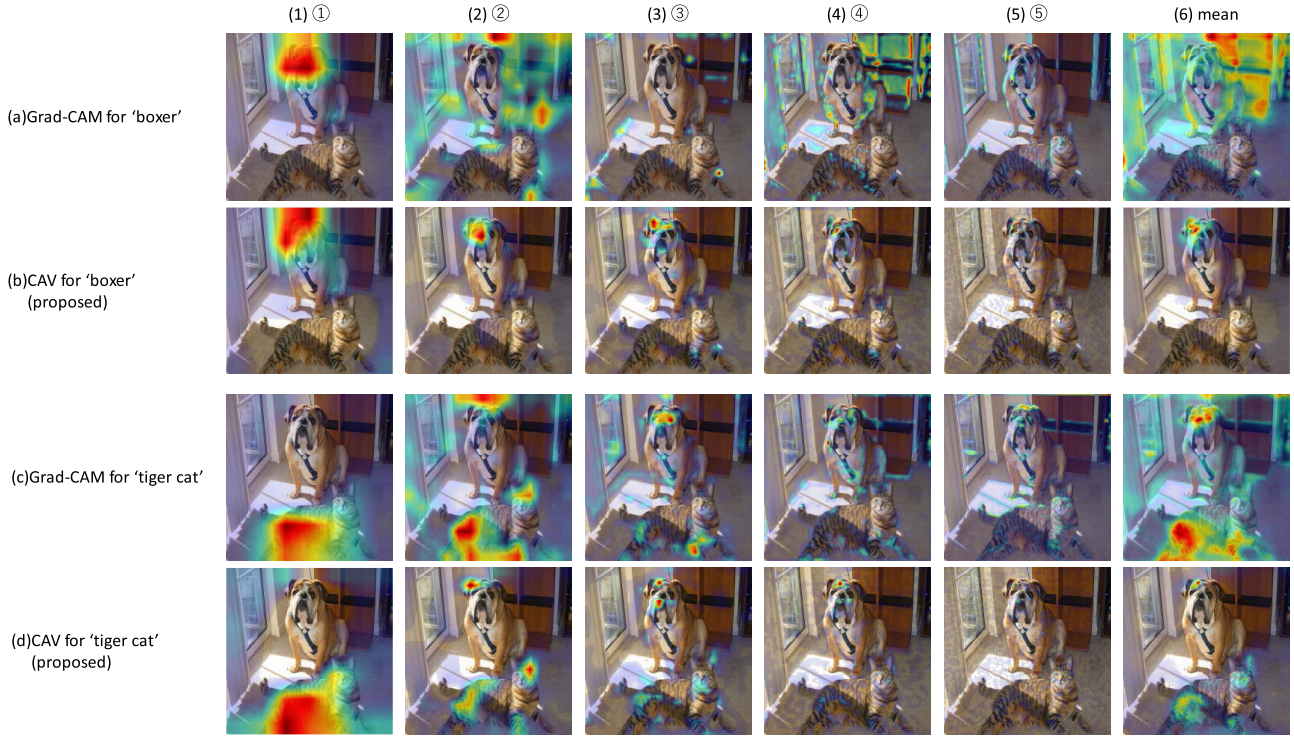

図 8　Grad-CAM と CAV による可視化結果．(a) 行は Grad-CAM による 'tiger cat' に対する可視化結果，(b) 行は CAV による 'tiger cat' に対する可視化結果，(c) 行は Grad-CAM による 'boxer' に対する可視化結果，(d) 行は CAV による 'boxer' に対する可視化結果を示している．(1) 列から (5) 列は可視化を行う層を表しており，(6) 列は全層における識別根拠の平均値を可視化した結果である

Fig. 8　Visualization results using Grad-CAM and CAV. Row (a) and (b) shows the visualization result of 'tiger cat' using Grad-CAM and CAV, row (c) and (d) shows the visualization result of 'boxer' using Grad-CAM and CAV respectively. Column (1) to (5) indicate layers that are used for classification reasons visualization, and column (6) is visualization of mean of classification reasons visualization in all layers.

(a)'Shetland sheepdog' and 'tennis ball'　(b)'zebra' and 'elephant'

図 9　識別根拠可視化実験に用いた画像．(a) は Shetland sheepdog という種類の犬とテニスボールを写した画像 [12]，(b) はシマウマと象を写した画像 [13] である．VGG モデルに入力するために，トリミングおよびリサイズ処理を適用した

Fig. 9　Images that were used for classification reasons visualization experiment. (a) is a picture with Shetland sheepdog and tennis ball [12], (b) is a picture with zebra and elephant [13]. Due to input to VGG model, we applied trimming and resizing for each image.

4.2　誤識別サンプルの解析

CAV を用いて，VGG モデルによる誤識別サンプルの解析を行った．ImageNet でも使用されているサンプルを用いて VGG モデルによる識別を行い，誤識別したサンプルの一部について，正解クラスと識別結果クラスの識別根拠を可視化し，誤識別原因を検証した．比較のために従来手法である Grad-CAM でも同様の識別根拠可視化を行った．実験に使用した誤識別画像を**図 11** に示す．

図 11 (a)，(b) はどちらも正解が 'pencil sharpener'（鉛筆削り）であるが，識別結果は 'padlock'（南京錠）であった．(c) は正解が 'goblet'（台と脚があり取っ手のないグラス）であるが，識別結果は 'perfume'（香水）であった．

図 11 の各画像について，正解クラスと識別結果のクラス両方の識別根拠を Grad-CAM および CAV で解析した結果を**図 12** に示す．4.1 節の実験と同様に複数の層における特徴量を用いた可視化を行ったが，代表として (1)，(4) 列に①，(2)，(5) 列に③の特徴量を用いた識別根拠の可視化結果，(3)，(6) 列に全層における識別根拠の平均値を示している．これは①における可視化結果は大局的な傾向，③における可視化結果は局所的な傾向，全層の平均は全体的な傾向をよく表していると考えたためである．

図 12 の (a1)，(a2) 行に図 11 (a) のサンプルの解析結果を示す．まず図 12 の (a1) 行 (1) 列と (a2) 行 (1) 列に，正解クラスと識別結果クラスの Grad-CAM による①の特徴量を用いた識別根拠可視化結果を示す．(a2) 行 (1) 列の方

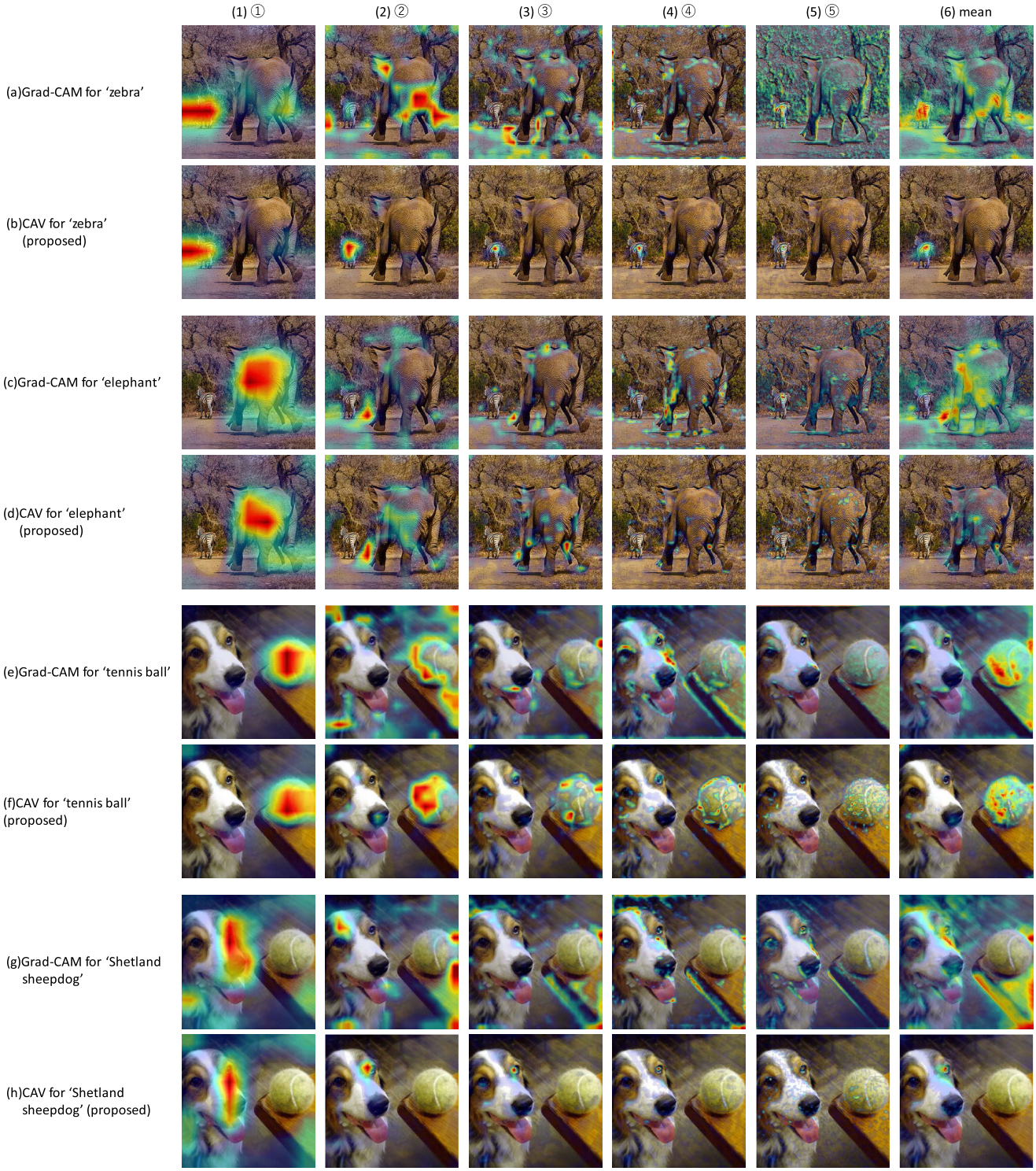

図 10　Grad-CAM と CAV による識別根拠可視化結果．(a) 行は Grad-CAM による ‘zebra’ の識別根拠可視化結果，(b) 行は CAV による ‘zebra’ の識別根拠可視化結果，(c) 行は Grad-CAM による ‘elephant’ の識別根拠可視化結果，(d) 行は CAV による ‘elephant’ の識別根拠可視化結果，(e) 行は Grad-CAM による ‘tennis ball’ の識別根拠可視化結果，(f) 行は CAV による ‘tennis ball’ の識別根拠可視化結果，(g) 行は Grad-CAM による ‘Shetland sheepdog’ の識別根拠可視化結果，(h) 行は CAV による ‘Shetland sheepdog’ の識別根拠可視化結果である．(1) 列から (5) 列は可視化を行う層を表しており，(6) 列は全層における識別根拠の平均値を可視化した結果である

Fig. 10　Visualization results using Grad-CAM and CAV. Row (a) and (b) shows the visualization result of ‘zebra’ using Grad-CAM and CAV, row (c) and (d) shows the visualization result of ‘elephant’ using Grad-CAM and CAV, Row (e) and (f) shows the visualization result of ‘tennis ball’ using Grad-CAM and CAV, row (g) and (h) shows the visualization result of ‘Shetland sheepdog’ using Grad-CAM and CAV respectively. Column (1) to (5) indicate layers that are used for classification reasons visualization, and column (6) is visualization of mean of classification reasons visualization in all layers.

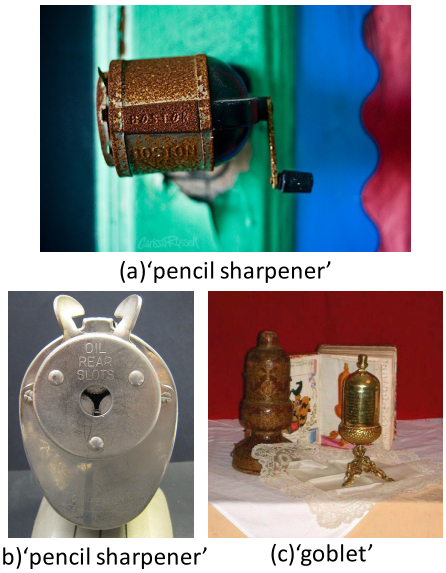

図 11　誤識別サンプルの解析実験に用いた画像．(a), (b) は ‘pencil sharpener’，(c) は ‘goblet’ とタグ付けされている

Fig. 11　Images that were used for analysis experiment of error sample. (a) and (b) are labeled as ‘pencil sharpener’, (c) is labeled as ‘goblet’.

がやや広く分布している傾向があるが，どちらのクラスに対する可視化結果も，対象物を中心に反応があり，具体的な特徴差については考察が難しい．また，図 12 の (a1) 行 (2) 列と (a2) 行 (2) 列に示す，Grad-CAM による③の特徴量を用いた識別根拠可視化結果を比較した場合も，識別根拠が対象物以外の部分に散在しており，やはり明確な差は述べにくい．図 12 の (a1) 行 (3) 列と (a2) 行 (3) 列に示す識別根拠の平均値による結果においても，対象物上の識別根拠の分布に明確な差は見つからない．

CAV による可視化結果を比較する．図 12 の (a1) 行 (4) 列と (a2) 行 (4) 列から，CAV による①の特徴量を用いた識別根拠可視化結果は Grad-CAM と同様の傾向と分かる．しかし，図 12 の (a1) 行 (5) 列と (a2) 行 (5) 列から分かるように CAV による③の特徴量を用いた識別根拠可視化結果が Grad-CAM とは大きく異なる．正解クラスである ‘pencil sharpener’ では鉛筆を差し込む面，識別結果クラスである ‘padlock’ では金属上の印字部分に強く識別根拠が表れており，これらが両クラス間を区別する視覚的特徴であると予想する．図 12 の (a1) 行 (6) 列と (a2) 行 (6) 列に示す識別根拠の平均においても，同様に ‘padlock’ に対して金属上の印字部分に強い識別根拠を示している．

次に図 12 の (b1), (b2) 行に示す，図 11 (b) の解析結果を説明する．図 11 (b) も図 11 (a) と同様に，正解は ‘pencil sharpener’，識別結果は ‘padlock’ である．図 12 の (b1), (b2) 行 (1), (4) 列を比較すると両手法の①の特徴量を用いた識別根拠可視化結果には大きな差が見当たらない．また，図 12 の (b1), (b2) 行 (2) 列に示す，Grad-CAM による③の特徴量を用いた識別根拠可視化結果においても，両クラス間の差は判別しにくい．一方，図 12 (b1), (b2) 行 (5) 列より，CAV による③の特徴量を用いた識別根拠可視化結果では，正解クラスである ‘pencil sharpener’ では鉛筆差込口，識別結果クラスである ‘padlock’ では印字部分に強い識別根拠が表れていることが分かり，これは図 11 (a) の検証結果と一致する．このことから ‘pencil sharpener’ を ‘padlock’ に誤識別する原因の 1 つとして，金属上の印字という視覚的特徴が大きく影響していると予想する．

図 12 (b1), (b2) 行 (3), (6) 列に示す，全層の識別根拠の平均値を比較した場合も，CAV の場合は ‘pencil sharpener’ では鉛筆差込口，‘padlock’ では印字部分に強い識別根拠が集中していることが観察できるが，Grad-CAM の場合は差が不明瞭な傾向がある．

次に図 12 (c1), (c2) 行に示す，図 11 (c) の解析結果を説明する．図 11 (c) は ‘goblet’ を ‘perfume’ に誤識別した例である．図 12 の (c1), (c2) 行 (1), (4) 列に示すとおり，大局的な傾向は両手法とも同様であり，‘goblet’ については対象物の台座から脚の部分，‘perfume’ については台座から杯の領域にかけて識別根拠が分布している．図 12 の (c1), (c2) 行 (2) 列に示す Grad-CAM の③の特徴量による可視化結果を見ると，識別根拠の局在性が低下して，対象物以外の領域にも識別根拠が散在していることが分かる．図 12 の (c1), (c2) 行 (3) 列より，平均値による全体的な傾向を見ても明確な差については判別が難しい．一方，CAV では図 12 の (c1) 行 (5) 列に示すように，③の特徴量を用いた ‘goblet’ に対する識別根拠は脚の付け根部分に集中していることが分かる．また，図 12 の (c2) 行 (5) 列から ‘perfume’ に対しては台座と杯の接続部分に識別根拠が集中していることが分かる．図 12 の (c1), (c2) 行 (6) 列より，全層の識別根拠の平均値からも ‘goblet’ に対しては脚の付け根部分，‘perfume’ に対しては台座と杯の接続部分が各クラスの重要な特徴量であることが分かる．

以上により，提案手法 CAV を用いることで局在性を維持したままで識別根拠を高解像度化することが可能となり，従来手法よりもより詳細に，誤識別原因や各クラスの重要な特徴量を可視化できるといえる．

4.3　処理速度

表 1 に示す実験環境において，Grad-CAM および CAV を実装し，処理速度を比較した．なお，モデルは上述の VGG モデルであり，GPU は使用せず CPU 処理のみで実行している．40 種類の画像を用いて各サンプル 5 回ずつ処理速度を計測し，その平均値を求めた．なお，処理時間には順方向の演算を含んでいる．結果を**表 2** に示す．

CAV は Grad-CAM よりも約 1.4 倍程度処理時間が長い結果を得た．理由は Full connection 層および Convolution 層において，各要素について入力と重みを乗算し，出力で正規化するという処理を行う点が最も大きく影響している

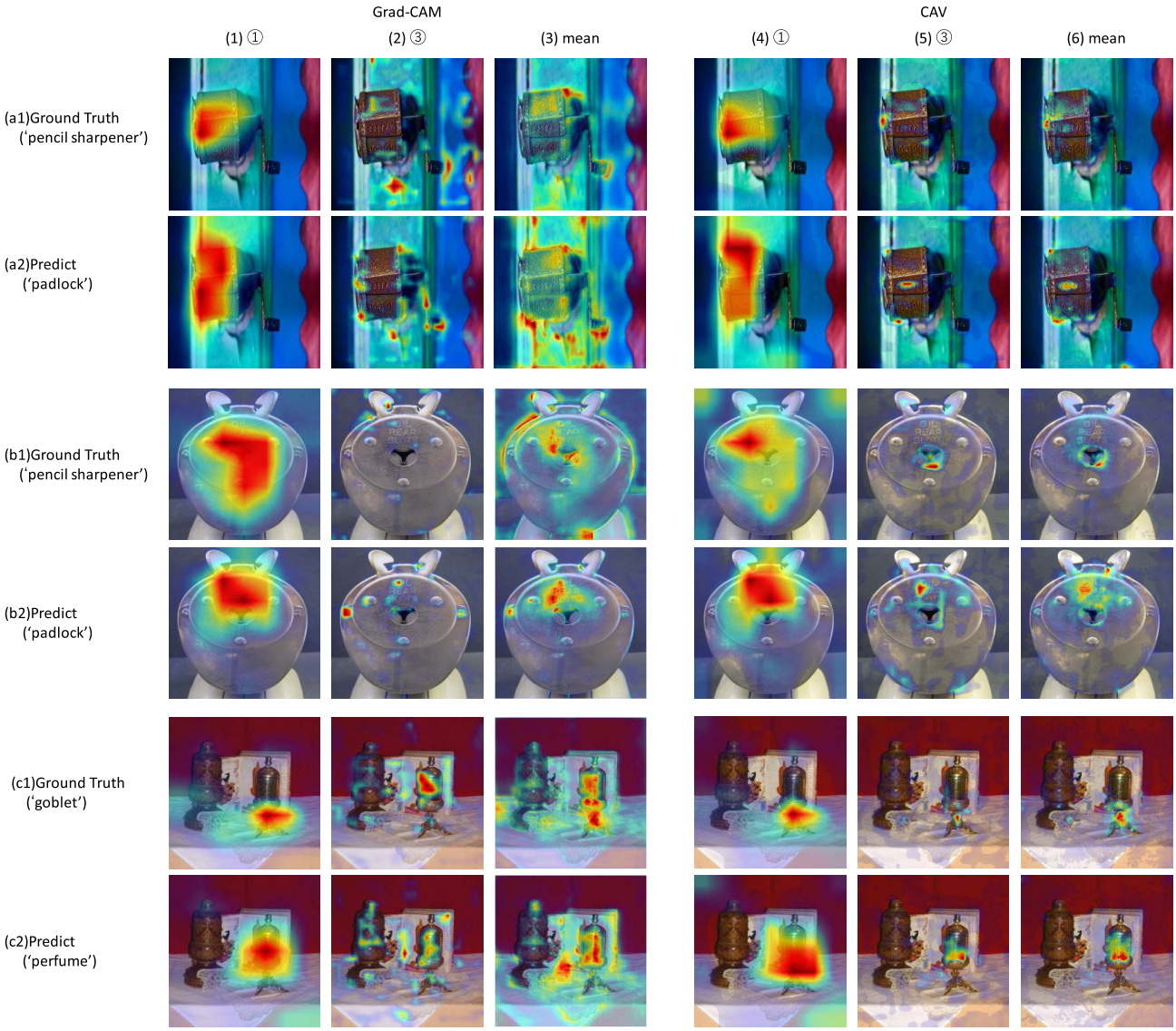

図 12　誤識別サンプルの解析結果．(a1), (a2) 行は図 11 (a), (b1), (b2) 行は図 11 (b), (c1), (c2) 行は図 11 (c) の識別根拠可視化結果を示す．また，(a1), (b1), (c1) 行は各サンプルの正解クラス，(a2), (b2), (c2) 行は識別結果のクラスの識別根拠可視化結果である．(1), (2), (3) 列は Grad-CAM による可視化結果であり，(1) は①，(2) は③の特徴量を用いて可視化した結果，(3) は全層の識別根拠の平均である．(4), (5), (6) 列は CAV による可視化結果であり，(4) は①，(5) は③の特徴量を用いて可視化した結果，(3) は全層の識別根拠の平均を示す

Fig. 12　Result of analysis experiment of error sample. Each row shows the result of classification reasons visualizations. (a1) and (a2) is visualization result for Fig.11(a), (b1) and (b2) is visualization result for Fig.11(b) and (c1) and (c2) is visualization result for Fig.11(c). Row (a1), (b1) and (c1) are visualization results on correct class, (a2), (b2) and (c2) are visualization results on inferred class. Column (1), (2) and (3) are visualization result by Grad-CAM using features in ①, ③ and mean of all layers respectively. Column (4), (5) and (6) are visualization result by CAV using features in ①, ③ and mean of all layers respectively.

と考える．今回は手法の効果検証を目的としたため，処理量の削減について十分な検討ができていないが，たとえば入力と重みの各要素の乗算等，順方向の演算と重複している計算があるため，順方向の一時演算結果の再利用等により，より高速な処理が可能になると考える．

5. 考察

提案手法で可視化した識別根拠について考察する．図 6 で示したサンプルについて図 8 では正の寄与率のみをヒートマップで示したが，提案手法は負の寄与率も算出する．CAV で算出した正の寄与率と負の寄与率を**図 13** に示す．

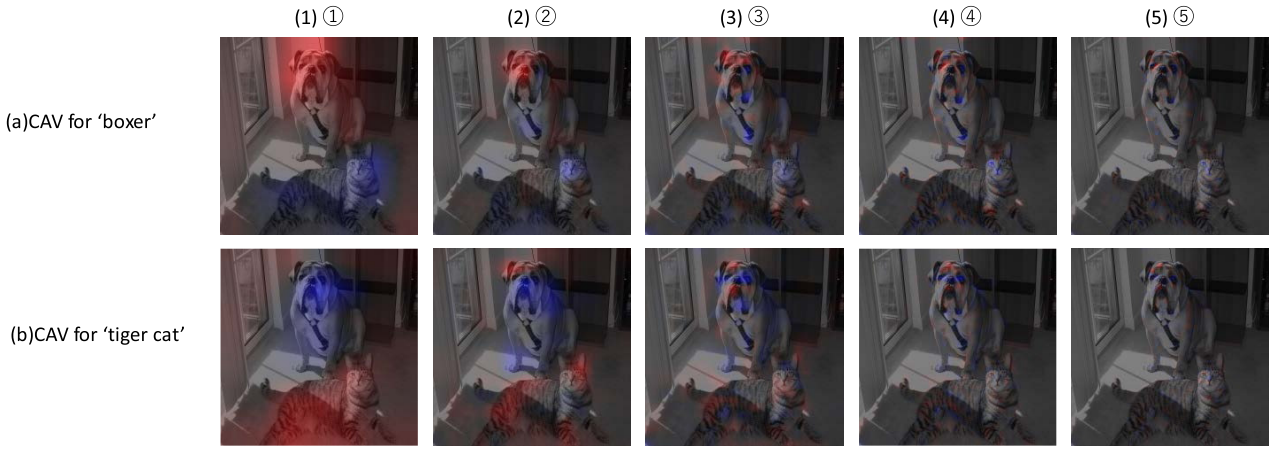

図 13　CAV による正負の寄与率マップ．赤成分は正の寄与率，青成分は負の寄与率を示す．(a) 行は 'tiger cat'，(b) 行は 'boxer' に対する可視化結果であり，グレイ画像化した入力画像と重畳している．(1) 列から (5) 列は可視化を行う層を表している

Fig. 13　Positive and negative contribution rate map by CAV. Red color indicates positive contribution rate and blue color indicates negative contribution rate. In rows (a) and (b) the visualization results of 'tiger cat' and 'boxer' are shown, these have merged with grayed input image. Column (1) to (5) indicate layers that are used for classification reasons visualization.

表 1　実験環境

Table 1　Experiment environment.

OS	Windows 10 64 bit
CPU	3.5GHz 6Cores
Memory	16GB
Programming Language	C++
Development Tool	Visual Studio 2017

表 2　処理速度

Table 2　Processing speed.

Method	Processing time
Grad-CAM	44.3 sec/sample
CAV	61.5 sec/sample

図 13 (a) 行は 'boxer'，図 13 (b) 行は 'tiger cat' に対する可視化結果であり，図 13 の (1) 列から (5) 列は図 5 の①から⑤の特徴量を用いた識別根拠可視化結果を表している．図 13 の赤成分は正の寄与率，青成分は負の寄与率を示している．識別根拠の妥当性を検証するために，識別根拠可視化結果に基づいて，特徴量を選択的に無効化し，識別スコアへの影響を観察する．識別スコアは VGG モデルの最終層の出力に対して Softmax を適用する前の値を使用した．理由は Softmax を適用後の相対的なスコアではなく，各クラスに対する絶対的なスコアの変化を観察するためである．

無効化の手法として 2 種類の手法を試した．手法 1 は寄与率が正（図 13 の赤で示した領域）かつ上位数%の領域に含まれる特徴量を無効化し，手法 2 は寄与率が負（図 13 の青で示した領域）かつ下位数%の領域に含まれる特徴量を無効化する手法である．もしも，識別根拠を正しく算出できていれば，前者の場合は特徴量を無効化しない場合よりも識別スコアが減少し，後者の場合は増大するはずである．2 種類の手法を，'tiger cat' と 'boxer' の両方について，図 5 の①から⑤の層に適用した．無効化する領域の割合は 1%，3%，5%の 3 種類を試した．無効化する具体的な方法は，無効化するピクセルを 0，それ以外の箇所を 1 とするマスク画像を作成し，3×3 の移動平均フィルタを適用後に特徴量マップに乗算する方法を用いた．移動平均フィルタを適用する理由は，特徴量を強制的に 0 にすることでエッジが発生し，本来は画像に含まれない特徴量が発現することを抑制するためである．**表 3** に適用結果を示す．

無効化前の識別スコアは 'boxer' クラスが 5.32，'tiger cat' クラスが 4.30 であった．表 3 (a) は 'boxer' クラスに手法 1 を適用した結果であり，想定どおりすべて無効化前のスコアよりもスコアが減少している．表 3 (b) は 'boxer' クラスに手法 2 を適用した結果であり，こちらも想定どおりすべて無効化前のスコアよりもスコアが増大している．

表 3 (c)，(d) は 'tiger cat' クラスに手法 1，2 を適用した結果であり，こちらも想定通りの結果となっている．また，無効化する面積が増えるほど，手法 1 ではスコアが減少，手法 2 では増大しており，識別スコアを増大する特徴量と抑制する特徴量を正確に示すことができていると考える．

次に，図 8 の (d) 行 (4) 列にて，'tiger cat' に対する識別根拠として犬の眉間部分に反応が表れている点を考察する．図 13 の (b) 行 (4) 列が，図 8 の (d) 行 (4) 列の正負の寄与率を示した結果である．確かに犬の眉間部分には正の寄与率（赤）が強く表れているが，目元やあごの部分等，負の寄与率（青）も多く表れていることが分かる．そこで

表 3　特徴量無効化による識別スコアの変化．(a)，(b) は ‘boxer’ に対する手法 1，2，(c)，(d) は ‘tiger cat’ に対する手法 1，2 の結果

Table 3　Changes of classification score by invalidation of features. (a) and (b) are results for ‘boxer’ using method 1 and 2, (c) and (d) are results for ‘tiger cat’ using method 1 and 2 respectively.

(a)Method 1 for ’boxer’

layer＼rate	5%	3%	1%
①	2.23	2.86	3.85
②	2.63	3.27	4.16
③	2.76	3.38	4.24
④	3.15	3.62	4.65
⑤	4.19	4.51	4.92

(b)Method 2 for ’boxer’

layer＼rate	5%	3%	1%
①	8.29	8.01	6.99
②	8.12	7.47	6.47
③	6.77	6.53	5.99
④	6.46	6.18	5.77
⑤	5.97	5.80	5.59

(c)Method 1 for ’tiger cat’

layer＼rate	5%	3%	1%
①	2.05	2.46	3.21
②	2.48	2.76	3.48
③	3.00	3.43	3.84
④	3.22	3.51	4.01
⑤	3.64	3.85	4.07

(d)Method 2 for ’tiger cat’

layer＼rate	5%	3%	1%
①	6.84	6.36	5.29
②	5.72	5.50	4.95
③	5.76	5.34	4.79
④	5.49	4.98	4.56
⑤	5.01	4.82	4.63

図 14　特徴量無効化箇所．(a) は犬が写った領域内の ‘tiger cat’ に対する正の寄与率が高い特徴量のみを有効化した画像，(b) は犬が写った領域内の ‘tiger cat’ に対する正の寄与率が高い特徴量のみを無効化した画像を表す

Fig. 14　Locations where the features are invalidated. (a) is an image where only high positive contribution rate features for ‘tiger cat’ in region of the dog are validated, (b) is an image where only high positive contribution rate features for ‘tiger cat’ in region of the dog are invalidated.

図 14 に示す特徴量無効化マスクを用いて，さらなる検証を行った．

図 14 の黒塗り部分は特徴量を無効化，すなわち，強制的に 0 にする箇所を示した図であり，図 14 (a) は犬が写った領域内の ‘tiger cat’ に対する正の寄与率が高い特徴量のみを有効化した画像，図 14 (b) は犬が写った領域内の ‘tiger cat’ に対する正の寄与率が高い特徴量のみを無効化した画像を表す．

図 14 (a) に示す箇所を無効化した場合，‘tiger cat’ に対する識別スコアは 4.30 から 5.76 に上昇した．これは，図 14 (a) に示す領域には ‘tiger cat’ の識別スコアを抑制する特徴量が多く含まれていることを示している．一方，

表 4　識別結果の比較．左は特徴量無効化前，右は特徴量無効化後の識別結果の Top10

Table 4　Comparison of classification result. Left table is classification result before feature invalidation, right table is classification result after feature invalidation.

before

Rank	Class #	Item
1	242	boxer
2	243	bull mastiff
3	246	Great Dane
4	292	tiger, Panthera tigris
5	282	tiger cat
6	159	Rhodesian ridgeback
7	172	whippet
8	180	American Staffordshire terrier
9	254	pug, pug-dog
10	247	Saint Bernard, St Bernard

after

Rank	Class #	Item
1	242	boxer
2	243	bull mastiff
3	246	Great Dane
4	180	American Staffordshire terrier
5	159	Rhodesian ridgeback
6	254	pug, pug-dog
7	172	whippet
8	225	malinois
9	209	Chesapeake Bay retriever
10	208	Labrador retriever
11	282	tiger cat
12	292	tiger, Panthera tigris

図 14 (b) の場合，‘tiger cat’ に対する識別スコアは 4.30 から 3.62 に低下した．また，**表 4** に図 14 (b) の特徴量を無効化した場合の識別結果の変化を示す．表 4 左は特徴量無効化前，表 4 右は図 14 (b) の特徴量無効化後の識別結果上位 10 クラスであり，青いセルはイヌ科，緑のセルはネコ科のクラスを示している．特徴量無効化後は ‘tiger cat’ のみならず，ネコ科のクラスが順位を下げ，Top 10 のすべてがイヌ科のクラスになっている．このことから VGG モデルは，図 14 (b) の黒塗り部分を，猫に無関係の領域にもかかわらず，ネコ科の動物の視覚的特徴としてとらえていると分かる．

以上から，VGG モデルは犬画像の中に，猫の視覚的特徴を発見しているが，その周辺に猫には存在しない視覚的特徴を多く発見しているため，総合的に猫ではないと判定していることが分かる．このように，提案手法を用いることで識別根拠となる領域や特徴を従来よりも詳細に分析することが可能となる．

6.　結言

本論文では，Neural Network を用いた識別器における識別根拠を，モデル逆解析により可視化する手法 CAV を提案し，画像識別に広く知られているモデルである VGG モデルを用いて，その効果を検証した．その結果，提案手法は従来手法よりも詳細に識別根拠となる領域や特徴を可視化できることを示した．提案手法は，どのような層で構成されたネットワークであっても，寄与率の算出方法を定義することで識別根拠の可視化を可能とする．よって，画像識別だけでなく，イメージキャプショニングや自然言語処理，音声認識等，様々なタスク向けのネットワークにおいても識別や判断の根拠を提示可能と考える．また，提案手法は層の構成や特別な学習も不要なため，学習済みのネットワークに対しても構成変更や再学習を行うことなく，適用可能な汎用性を持つ．本論文では VGG モデルを対象に実験および考察を行ったが，今後より多くのモデルに対する検証を行いつつ，提案技術の展開と応用について検討し

ていく．

参考文献

[1] LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W. and Jackel, L.D.: Backpropagation applied to handwritten zip code recognition, *Neural Computation* (1989).

[2] Springenberg, J.T., Dosovitskiy, A., Brox, T. and Riedmiller, M.A.: Striving for Simplicity: The All Convolutional Net, *CoRR*, abs/1412.6806 (2014).

[3] Zeiler, M.D. and Fergus, R.: Visualizing and understanding convolutional networks, ECCV (2014).

[4] Mahendran, A. and Vedaldi, A.: Salient deconvolutional networks, *European Conference on Computer Vision* (2016).

[5] Selvaraju, R.R., Das, A., Vedantam, R., Cogswell, M., Parikh, D. and Batra, D.: Grad-cam: Why did you say that? Visual explanations from deep networks via gradient-based localization, *CoRR*, abs/1610.02391 (2016).

[6] Simonyan, K., Vedaldi, A. and Zisserman, A.: Deep inside convolutional networks: Visualising image classification models and saliency maps, *CoRR*, abs/1312.6034 (2013).

[7] Erhan, D., Bengio, Y., Courville, A. and Vincent, P.: Visualizing Higherlayer Features of a Deep Network, Technical Report 1341, University of Montreal, (2009).

[8] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A. and Torralba, A.: Learning Deep Features for Discriminative Localization, *CVPR* (2016).

[9] Lin, M., Chen, Q. and Yan, S.: Network in network, *ICLR* (2014).

[10] Simonyan, K. and Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition, *ICLR* (2015).

[11] sabianmaggy, flickr (2005), available from ⟨https://www.flickr.com/photos/40765798@N00/202734059⟩.

[12] carterse, flickr (2008), available from ⟨https://bit.ly/2BEtSMA⟩.

[13] Steve Slater, flickr (2010), available from ⟨https://www.flickr.com/photos/wildlife_encounters/8023932296⟩.

柿下 容弓　（正会員）

2008 年電気通信大学大学院情報通信工学専攻修士課程修了．2008 年株式会社日立製作所に入社．現在，株式会社日立製作所研究開発グループ・研究員．画像認識，機械学習の研究に従事．

服部 英春

1994 年東京電機大学大学院理工学研究科情報科学専攻修士課程修了．1994 年株式会社日立製作所に入社．現在，株式会社日立製作所研究開発グループ・主任研究員．医用画像，車載画像向け画像認識，画像処理，機械学習の研究に従事．映像情報メディア学会，日本医用画像工学会，各会員．