

## 非構造化データのテンソルデータ処理教育への導入事例

山本 直樹<sup>1,\*</sup> 石田 明男<sup>2</sup> 村上 純<sup>1</sup>

## Introduction Example of Unstructured Data to Tensor Data Processing Education

Naoki Yamamoto<sup>1,\*</sup>, Akio Ishida<sup>2</sup>, Jun Murakami<sup>1</sup>

Big data includes data such as PDFs, audios, images, and videos, which are called unstructured data, and data scientists need to have the skills to process those data. This research aims at education on unstructured data processing of images and videos. Regarding tensor data processing, we have been developing tools and learning materials that are useful for learning it. Furthermore, we have been conducting education using them. This time, as an applied content of the education, we introduce the education of image / video processing using HOSVD, which is one of the tensor decomposition, into subjects of our advanced course and department. Therefore, this paper describes that case. As a result, it was found that students could generally perform the processing by applying HOSVD.

キーワード：テンソルデータ処理、画像／動画データ、テンソル分解、高次特異値分解、プログラミング教育

**Keywords** : Tensor data processing, Image and video data, Tensor decomposition, HOSVD, Programming education

## 1. はじめに

PDF、音声、画像、動画などのデータは非構造化データと呼ばれ、一般にこれらのデータの分析を行う前にはデータの整理や変換が必要とされる<sup>(1)</sup>。このような非構造化データの処理には、言語処理、音声／音楽処理、画像／動画処理があるが、これらはデータサイエンティストに必要なスキルとして分類されている<sup>(2)</sup>。本研究では、これらの処理のうち画像／動画処理の教育を対象とする。

これまで我々は、立体パズルやマクマホン立方体を題材として、高階テンソル（多次元配列）データの操作や演算を学習するためのツールや教材の開発をしている<sup>(3),(4),(5)</sup>。さらに、これらの研究成果を活用して、本校専攻科の科目においてテンソルデータ処理に関する教育を実践している<sup>(6),(7)</sup>。

これまでの取組みでは、テンソルデータ処理の比較的基礎的な内容を扱っているが、本論文では、より応用的な内容である画像／動画処理の教育を専攻科および本科の科目に導入したので報告する。

## 2. 非構造化データのテンソルデータ処理の理論とプログラミング環境

## 2.1 高階テンソルとテンソル分解

今回、非構造化データとして、カラー画像およびカラー動画を扱うが、これらはそれぞれ、3次元および4次元配列のデータとして扱われる。本研究では、この多次元配列のことを高階テンソルと呼び、例えば、 $n$ 階テンソルは $n$ 次元配列を表す。

高階テンソルデータを低次元化する処理手法にテンソル分解があるが、本教育ではその重要な手法の1つである高次特異値分解（HOSVD; Higher-Order Singular Value Decomposition）<sup>(8)</sup>を取り扱う。HOSVDは3階以上のテンソルに対する特異値分解（SVD）のことで、行列のSVDを拡張したものである。ここで、 $K$ 階テンソルのHOSVDの定義を次に示す<sup>(8)</sup>。

## 〔定義1〕 HOSVD

$K$ 階テンソル $\mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_K}$ は、大きさが $I_1 \times I_2 \times \dots \times I_K$ で、その各要素 $t_{i_1 i_2 \dots i_K}$  ( $i_k = 1, 2, \dots, I_k; k = 1, 2, \dots, K$ )に実数値をもつ。このとき、 $\mathcal{T}$ はSVDの特異値に相当するコアテンソル $\mathcal{C} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_K}$ と正規直交行列 $\mathbf{U}^{(k)} \in \mathbb{R}^{I_k \times I_k}$  ( $k = 1, 2, \dots, K$ )の $n$ -モード積<sup>(8)</sup>と呼ばれるテンソルと行列の積の形で、次のように分解される。

$$\mathcal{T} = \mathcal{C} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \dots \times_K \mathbf{U}^{(K)} \quad (1)$$

この分解をHOSVDと呼ぶ。ただし、 $\times_k$ は $k$ -モード積を

<sup>1</sup> 電子情報システム工学系  
〒861-1102 熊本県合志市須屋 2659-2  
Faculty of Electronics and Information Systems Engineering,  
2659-2 Suya, Koshi-shi, Kumamoto, Japan 861-1102

<sup>2</sup> リベラルアーツ系  
〒861-1102 熊本県合志市須屋 2659-2  
Faculty of Liberal Arts,  
2659-2 Suya, Koshi-shi, Kumamoto, Japan 861-1102

\* Corresponding author:  
E-mail address: naoki@kumamoto-nct.ac.jp (N. Yamamoto).

表し、 $n$ -モード積の演算子である。

(定義終わり)

ところで、式(1)の分解では右辺の演算から元のテンソルを完全に再構成できる。元のテンソルの低次元化は、式(1)における $\mathbf{C}$ と $\mathbf{U}^{(k)}$ の大きさを減らすことにより行え、これを Truncated HOSVD と呼び、この定義を次に示す<sup>(9)</sup>。

[(定義2) Truncated HOSVD ]

定義 1 の $\mathcal{T}$ の近似として、テンソルのランク  $\text{rank} - (r_1, r_2, \dots, r_K)$  をもつ  $K$  階テンソル $\hat{\mathcal{T}}$ を求めるものとする。なお、 $r_k (r_k \leq I_k; k = 1, 2, \dots, K)$ は  $k$ -ランクを表し、文献(8)の  $n$ -ランクのことである。いま、式(1)の $\mathbf{U}^{(k)}$ の $r_k + 1$ 列以降を打ち切った大きさ  $I_k \times r_k$  の行列を $\hat{\mathbf{U}}^{(k)} \in \mathbb{R}^{I_k \times r_k}, (k = 1, 2, \dots, K)$ とすると、 $\mathcal{T}$ は次式で近似される。

$$\mathcal{T} \approx \hat{\mathcal{T}} = \hat{\mathbf{C}} \times_1 \hat{\mathbf{U}}^{(1)} \times_2 \hat{\mathbf{U}}^{(2)} \dots \times_K \hat{\mathbf{U}}^{(K)} \quad (2)$$

ここで、 $\hat{\mathbf{C}}$ は打ち切られたコアテンソルであり、次式として定義される。

$$\hat{\mathbf{C}} = \mathcal{T} \times_1 \hat{\mathbf{U}}^{(1)\top} \times_2 \hat{\mathbf{U}}^{(2)\top} \dots \times_K \hat{\mathbf{U}}^{(K)\top} \in \mathbb{R}^{r_1 \times r_2 \times \dots \times r_K} \quad (3)$$

ただし、 $\top$ は行列の転置を表す。

(定義終わり)

ここで、式(2)のように近似された場合のデータ圧縮率を以下に定義する<sup>(10),(11)</sup>。

[(定義3) HOSVD の圧縮率 ]

式(2)で近似されたとき、HOSVD の圧縮率 $c_{\text{hosvd}}$ は、近似前後のデータ量をそれぞれ $\alpha, \beta$ として、次式により定義する。

$$\begin{aligned} c_{\text{hosvd}} &= 1 - \beta/\alpha, \\ \alpha &= I_1 I_2 \dots I_K, \\ \beta &= I_1 r_1 + I_2 r_2 + \dots + I_K r_K + r_1 r_2 \dots r_K \end{aligned} \quad (4)$$

(定義終わり)

また、ここでは画像の画質評価にピーク信号対雑音比 (PSNR) を利用する。この定義は次の通りである<sup>(12)</sup>。

[(定義4) PSNR ]

大きさ $H \times W$ の濃淡画像 $\mathbf{f}, \mathbf{g}$ があり、参照画像を $\mathbf{f}$ 、テスト画像を $\mathbf{g}$ とすると、PSNR は次式で定義される。

$$\text{PSNR}(\mathbf{f}, \mathbf{g}) = 10 \cdot \log_{10} \frac{\text{MAX}_I^2}{\text{MSE}(\mathbf{f}, \mathbf{g})} \quad (5)$$

ここで、 $\text{MAX}_I$ は画像が取りうる最大ピクセル値、 $\text{MSE}(\mathbf{f}, \mathbf{g})$ は次式で与えられる平均2乗誤差である。

$$\text{MSE}(\mathbf{f}, \mathbf{g}) = \frac{1}{HW} \sum_{i=1}^H \sum_{j=1}^W (f_{ij} - g_{ij})^2 \quad (6)$$

(定義終わり)

なお、PSNR が高いほど画像の画質が良いことを表す。

## 2.2 利用するプログラミング環境

表1 プログラミングに用いる関数

rTensor パッケージ	
as.tensor	テンソルオブジェクトの生成
hosvd	(Truncated) HOSVD の計算
fnorm	フロベニウスノルムの計算
imager パッケージ	
load.image	画像データの読み込み
width	データサイズの取得
height	
depth	
spectrum	画像、ビデオなどの種別の取得
as.cimg	cimg クラスへの変換
plot	画像の表示
load.video	ビデオデータの読み込み
play	ビデオの再生
save.video	ビデオの保存

本教育では、プログラミング言語として R<sup>(13)</sup>を利用する。テンソル処理用の R パッケージとして rTensor<sup>(14)</sup>、画像/動画処理用パッケージとして imager<sup>(15)</sup>をインストールして利用する。なお、R のパッケージとは別に、imager の関数からビデオを制御するために FFmpeg<sup>(16)</sup>と呼ばれるビデオ処理用のアプリケーションをインストールしておく必要がある。表1にプログラミングで利用する主な関数を示す。

## 3. テンソルデータ処理教育への導入事例

### 3.1 専攻科目での実施事例

この事例は、令和元年度および2年度の本校専攻科1年次後期選択開講科目「データサイエンス」で実施したものである。授業のテーマは、「テンソル分解・非構造化データ処理」である。

テンソル分解の内容としては、テンソル分解の概要、HOSVD の定義とアルゴリズム、hosvd 関数の仕様、フロベニウスノルムの定義、HOSVD の R での実装例について説明した。非構造化データ処理の内容としては、imager パッケージの概要と提供される関数、カラー画像の HOSVD の概要と R での実装例、元画像と再構成画像の比較、画質の評価尺度の1つである PSNR について説明した。

次に、動画データの前処理、FFmpeg のインストール、カラー動画の HOSVD の概要と R での実装例、元動画と再構成動画の比較、HOSVD の圧縮率について説明した後に、演習を行った。なお、動画データ処理については、令和2年度から新たに追加した内容である。

ここでは、8つ準備した実装例のうち2つを紹介する。

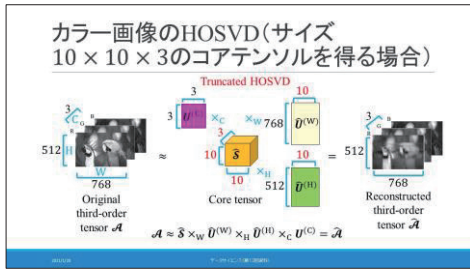


図1 カラー画像の Truncated HOSVD の説明

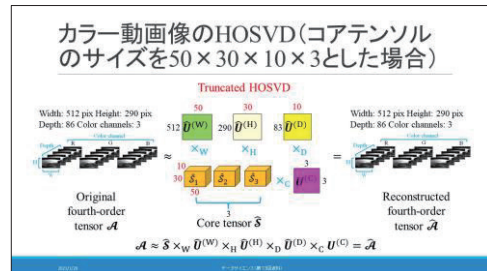


図3 カラー動画の Truncated HOSVD の説明



図2 元画像と再構成画像の比較例



図4 元動画と再構成動画の比較例

表2 図2の結果を確認するRスクリプト

```
library(rTensor) # rTensor パッケージ利用
library(imager) # imager パッケージ利用
# 画像データの読み込みと表示
img <- load.example("parrots"); plot(img)
# 画像サイズの取得
w <- width(img); h <- height(img)
s <- spectrum(img) # データ種別の取得
# テンソルオブジェクトの生成
A <- as.tensor(array(img[, , 1, ], dim=c(w,h,s)))
# Truncated HOSVD の計算
hosvdA <- hosvd(A, ranks=c(10,10,3))
recA <- hosvdA$est@data # 再構成画像の取得
# 再構成画像の cimg クラスへの変換と表示
recimg <- as.cimg(recA); plot(recimg)
# R,G,B 成分の PSNR の計算
MAX <- 1.0
MSE_R <- fnorm(A[, , 1]-recA[, , 1])^2/(h*w)
PSNR_R <- 10*log10(MAX^2/MSE_R)
MSE_G <- fnorm(A[, , 2]-recA[, , 2])^2/(h*w)
PSNR_G <- 10*log10(MAX^2/MSE_G)
MSE_B <- fnorm(A[, , 3]-recA[, , 3])^2/(h*w)
PSNR_B <- 10*log10(MAX^2/MSE_B)
```

最初の例は、imager パッケージに予め用意された RGB 画像である”parrots”を用いたものである。これを、大きさ  $512 \times 768 \times 3$  の 3 階テンソルとして取り扱った。図1は、このデータに Truncated HOSVD を適用して、データの次元削減を行う原理の説明に用いたものである。図2は、表2に記述されたスクリプトの実行結果を示し、元画像と再構成画像の比較の説明に用いた。学生は、演習時にこのスクリプトにおける hosvd 関数のオプションのパラメータを適宜変更して、図2に示されるような実行結果

表3 図4の結果を確認するRスクリプト

```
library(rTensor)
library(imager)
# 動画データを読み込み
X <- load.video("car_data.mp4")
play(X, loop=TRUE) # 動画の再生
# 動画サイズの取得
w <- width(X); h <- height(X); d <- depth(X)
s <- spectrum(X) # データ種別の取得
# テンソルオブジェクトの生成
B <- as.tensor(array(X, dim=c(w,h,d,s)))
# Truncated HOSVD の計算
hosvdB <- hosvd(B, ranks=c(50,30,10,3))
recB <- hosvdB$est@data # 再構成動画の取得
recvideo2 <- as.cimg(recB) # cimg クラスに変換
# 再構成動画の再生
play(recvideo2, loop=TRUE)
# 動画の保存
save.video(recvideo2, "rec_car_data2.mp4", fps=30)
# 圧縮率の計算
alpha <- w*h*d*s
beta <- w*50+h*30+d*10+s*3+50*30*10*3
c_hosvd <- 1-beta/alpha
```

を比較した。この実施について、課題レポートの取組み状況から、令和元年度は、約 81% の学生が HOSVD を実行して得られた結果の比較ができ、そのうち約 55% の学生が PSNR の結果の確認までできていた。令和2年度では、約 87% の学生が HOSVD の結果の比較ができ、そのうち約 63% が PSNR の確認までできていた。

2 つ目の例を図3および図4に示す。ここでは、GoPro HERO8 Black により撮影された車載動画を利用した。これを、FFmpeg を用いて前処理し、大きさ  $290 \times 512 \times 86 \times$

3の4階テンソルとして取り扱った。図3は、動画像データの次元削減の原理を説明するために利用し、図4は、元データと再構成データの比較を行う表3のスキプートの実行結果の解説に利用した。学生は、演習時に、表3のスキプートの `hosvd` 関数のパラメータを変更して、再構成された動画像がどのように変化するかと比較や圧縮率の計算を行った。この実施状況として、令和2年度は、約77%の学生がHOSVDを適用した結果の比較ができ、そのうち約54%の学生が圧縮率の結果確認までできていた。

### 3.2 本科実験科目での実施事例

もう1つは、令和2年度の間情報システム工学科4年次情報工学実験IIにおける創造実験で実施した事例である。テーマは、「テンソル分解による気象データの分析に関する研究」である。学生には、前節の事例で述べた科目で利用した説明資料を配布し、それを参考にして実験を行わせた。

学生は、情報通信研究機構(NICT)から提供されているひまわり8号の衛星画像を収集して実験に利用した。その画像にTruncated HOSVDを適用して、いくつかの再構成画像の取得とPSNRの計算を行い、再構成画像の比較、コアテンソルの大きさとPSNRの関係について考察させた。また、晴れと曇りの時の衛星画像から、それぞれ得られたコアテンソル値を比較して、両者間でその値の分布の仕方に違いがあることについて考察を行わせた。以上が実施できたことから、学生は自ら収集した画像データにHOSVDを適用し、データ分析に活用できていることが分かった。

## 4. まとめ

本論文では、テンソルデータ処理の応用として、HOSVDによる画像および動画像の非構造化データの処理について教育した事例について報告した。専攻科の科目では、Rを用いて画像/動画処理を行った。学生の課題の取組み状況から、画像処理は8割強の学生が、動画処理は8割弱の学生が実施できていることが分かった。また、本科の創造実験では、学生は専攻科の科目で利用した教材を参考にしながら実験を進め、HOSVDを理解して画像データの分析に適用できていることが分かった。

今後の課題としては、専攻科の科目で、学生はHOSVDによる画像/動画処理は概ね実施できているものと考えられるが、PSNRや圧縮率の確認までできている学生の割合が低い傾向が見られたため、課題の内容をさらに工夫することが挙げられる。また、講義の冒頭でテンソル

分解の概要説明をする際に、具体的な応用例を紹介して学生に興味を持たせるなど、学習への動機付けを工夫していくことも課題の1つである。

(令和3年10月11日受付)

(令和3年12月24日受理)

### 参考文献

- (1) 総務省:「総務省 ICT スキル総合習得プログラム講座 3-1」, [https://www.soumu.go.jp/ict\\_skill/pdf/ict\\_ev\\_el\\_3\\_1.pdf](https://www.soumu.go.jp/ict_skill/pdf/ict_ev_el_3_1.pdf), (2021.8.11 閲覧).
- (2) データサイエンティスト協会, 情報処理推進機構:「データサイエンティストのためのスキルチェックリスト/タスクリスト概説」, <https://www.ipa.go.jp/files/000083733.pdf>, (2021.8.16 閲覧).
- (3) A. Ishida, et al.: “Solving 3-D Puzzles Using Tensor Decomposition and Application to Education of Multidimensional Data Analysis”, *International Journal of Machine Learning and Computing*, Vol.8, No.5, pp.447-453 (2018).
- (4) N. Yamamoto, et al.: “Development of Teaching Tool for Supporting Understanding of Tensor Decomposition Using MacMahon’s Coloured Cubes”, *International Journal of Information and Education Technology*, Vol.10, No.1, pp.14-19 (2020).
- (5) N. Yamamoto, et al.: “Development of Online Learning Material for Data Science Programming Using 3D Puzzle”, *International Journal of Information and Education Technology*, Vol.11, No.4, pp.154-163 (2021).
- (6) 山本直樹 他:「マクマホン立方体を用いたテンソルデータ処理に関するデータサイエンス教育の試み」, 熊本高専研究紀要, Vol.12, pp.45-48 (2021).
- (7) 山本直樹 他:「高専専攻科におけるテンソルデータ処理プログラミング教育の実践」, ソフトウェア・シンポジウム 2021 論文集, ソフトウェア技術者協会, p.124 (2021).
- (8) L. De Lathauwer, et al.: “A Multilinear Singular Value Decomposition”, *SIAM Journal on Matrix Analysis and Applications*, Vol.21, No.4, pp.1253-1278 (2000).
- (9) N. Vannieuwenhoven, et al.: “A New Truncation Strategy for the Higher-Order Singular Value Decomposition”, *SIAM Journal on Scientific Computing*, Vol.34, No.2, pp.1027-1052 (2012).
- (10) 井上光平 他:「同時低ランク近似による行列データの次元削減」, 電子情報通信学会論文誌 D-II, Vol.J88-D-II, No.7, pp.1166-1172 (2005).
- (11) 井上光平 他:「テンソルの同時低ランク近似と主成分分析の関係」, 情報処理学会論文誌 コンピュータビジョンとイメージメディア, Vol.1, No.2, pp.1-9 (2008).
- (12) A. Horé, et al.: “Image Quality Metrics: PSNR vs. SSIM”, *Proceeding of 20th International Conference on Pattern Recognition*, pp.2366-2369 (2010).
- (13) R. C. Team: “R: A Language and Environment for Statistical Computing”, R Foundation for Statistical Computing, Vienna, Austria.
- (14) J. Li, et al.: “Package ‘rTensor’”, <https://cran.r-project.org/web/packages/rTensor/rTensor.pdf>, (Retrieved Aug. 17, 2021).
- (15) S. Barthelme, et al.: “Package ‘imager’”, <https://cran.r-project.org/web/packages/imager/imager.pdf>, (Retrieved Aug. 17, 2021).
- (16) FFmpeg [Online]. Available: <https://www.ffmpeg.org/>, (Retrieved Aug. 17, 2021).