Compositional Hierarchical Tensor Factorization: Representing Hierarchical Intrinsic and Extrinsic Causal Factors

M. Alex O. Vasilescu
maov@cs.ucla.edu
Department of Computer Science
University of California, Los Angeles

Eric Kim
ekim@cs.ucla.edu
Tensor Vision Technologies
Los Angeles, CA, USA

1 INTRODUCTION

Statistical data analysis that disentangles the causal factors of data formation and computes a representation that facilitates the analysis, visualization, compression, approximation, and/or interpretation of the data is challenging and of paramount importance.

“Natural images are the composite consequence of multiple constituent factors related to scene structure, illumination conditions, and imaging conditions. Multilinear algebra, the algebra of higher-order tensors, offers a potent mathematical framework for analyzing the multifactol structure of image ensembles and for addressing the difficult problem of disentangling the constituent factors or modes.” (Vasilescu and Terzopoulos, 2002 [59])

Scene structure is composed from a set of objects that appear to be formed from a recursive hierarchy of perceptual wholes and parts whose properties, such as shape, reflectance, and color, constitute a hierarchy of intrinsic causal factors of object appearance. Object appearance is the compositional consequence of both an object’s intrinsic and extrinsic causal factors, where the extrinsic causal factors are related to illumination, and imaging conditions. Therefore, this paper proposes a unified tensor model of wholes and parts, and introduces a compositional hierarchical tensor factorization that disentangles the hierarchical causal structure of object image formation, and subsumes multilinear block tensor decomposition as a special case. The resulting object representation is an interpretable combinatorial choice of wholes’ and parts’ representations that renders object recognition robust to occlusion and reduces training data requirements. We demonstrate our approach in the context of face recognition by training on an extremely reduced dataset of synthetic images, and report encouraging face verification results on two datasets – the Freiburg dataset, and the Labeled Face in the Wild (LFW) dataset consisting of real-world images, thus, substantiating the suitability of our approach for data-starved domains.

ACM Reference Format:
M.A.O. Vasilescu and E. Kim. 2019. Compositional Hierarchical Tensor Factorization: Representing Hierarchical Intrinsic and Extrinsic Causal Factors, In The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD ’19): Tensor Methods for Emerging Data Science Challenges, August 04-08, 2019, Anchorage, AK. ACM, New York, NY, USA, 9 pages.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
KDD ’19: TMEDSC, August 4-8, 2019. Anchorage, AK, USA © 2019 Copyright held by the owner/author(s). ACM ISBN 978-1-4503-6201-6/19/08.
global object features, and local features mitigates the shortcoming of both approaches [40, 43].

Deep learning methods, which have become a highly successful approach for object recognition, compute feature hierarchies composed of low-level and mid-level features either in a supervised [37], unsupervised [23, 34, 47] or semi-supervised manner [44]. This has been achieved by composing modules of the same architectures, such as Restricted Boltzmann Machines [23], autoencoders [34], or various forms of encoder-decoder networks [7, 28, 47].

Cohen et al.’s [11, 12] theoretical results have show that convolutional neural networks (CNNs) are equivalent to hierarchical Tucker factorization (hierarchical multilinear tensor SVD) [19, 20], and shallow networks are equivalent to linear tensor factorizations, aka CANDECOMP/Parafac (CP) tensor factorization [8, 21]. Vasilescu and Terzopoulos [54, 56, 61–63] demonstrated that tensor algebra is a suitable, interpretable framework for mathematically representing and disentangling the causal structure of data formation in computer vision, computer graphics and machine learning [57, 63], Figure 1. In addition to the lack of interpretability, implementation differences between CNNs and a tensor algebraic approach impacts data needs, memory/storage and computational complexity, often rendering CNN models difficult to deploy on mobile devices, or any devices with limited computational resources.

Inspired and inspired by Cohen et al.’s [11, 12] theoretical results, and by the TensorFaces and Human Motion Signatures approach [54, 60, 62], we propose a unified tensor model of wholes and parts based on a reconceptualization of the data tensor as a hierarchical data tensor. Defining a hierarchical data tensor enables a single elegant mathematical model that can be optimized in a principled manner instead of employing a myriad of individual part-based engineering solutions that independently represent each part and attempt to compute all possible dimensionality reduction permutations. Our factorization optimizes simultaneously across all the wholes and parts of the hierarchy, learns a convolitional feature hierarchy of low-level, mid-level and high-level features, and computes an interpretable compositional object representation of parts and wholes. Our resulting object representation is a combinatorial choice of part representations, that renders object recognition robust to occlusion while bypassing large training data requirements.

Our compositional tensor factorization and learnt feature hierarchy is also applicable to Convolutional Neural Networks and their variants. Since CNNs learn millions of parameters that may lead to redundancy and poor generalization, our factorization and dimensionality reduction approach may be employed to reparameterize and reduce a tensor of CNN parameters, potentially resulting in better generalization [30, 32, 36, 45].

DeepFace, a convolutional neural network approach for face verification, [10, 24, 51, 52, 68] celebrated closing the gap on human-level performance for face verification on the LFW database of 13,233 images of 5,749 people in the news, by training on a large dataset of 4,400,000 facial images from 4,030 people, the same order of magnitude as the number of people in the test data set [46].

---

1While the Tucker factorization performs one matrix SVD to represent a causal factor subspace (one orthonormal mode matrix), a Hierarchical Tucker is a hierarchical computation of the Tucker factorization that employs a stack of SVDs to represent a causal factor subspace (one orthonormal mode matrix). For computational efficiency, the authors [19, 20] prescribe a stack of QR decompositions instead of a stack of SVD.

2The suitability of the tensor framework was also demonstrated in the context of computer graphics by synthesizing new textures [62], performing expression retargeting/animation [65] and human motion synthesis [55, 56].
A generalization of the product of two matrices is the product of a tensor. The mode-
matrixizing of tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M}$ is defined as the matrix $A_{(m)} \in \mathbb{R}^{I_m \times (I_1 \cdot I_2 \cdot \cdots \cdot I_{m-1} \cdot I_{m+1} \cdot \cdots \cdot I_M)}$. As the parenthetical ordering indicates, the mode-
matrixizing of tensor $\mathcal{A}$ is the interaction between different causal factors of data formation. (Data
should be centered, but for this display it was added back.) (b) These set of images demonstrate the models ability to disentangle
the causal factors. Illumination effects, such as highlights, cast shadows and shading [61] are progressively reduced. (c) Vlasic et al [65]
generate new images by performing multilinear expression re-targeting.

**Definition 1 (Mode-$m$ Matrizing).** The mode-$m$ matrixing of tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M}$ is defined as the matrix $A_{(m)} \in \mathbb{R}^{I_m \times (I_1 \cdot I_2 \cdot \cdots \cdot I_{m-1} \cdot I_{m+1} \cdot \cdots \cdot I_M)}$. As the parenthetical ordering indicates, the mode-$m$ column vectors are arranged by sweeping all the other mode indices through their ranges, with smaller mode indexes varying more rapidly than larger ones; thus,

$$A_{(m)} = a_{i_1 \ldots i_m \ldots i_M}, \text{ where } j = i_m \text{ and } k = 1 + \sum_{n=m}^{M} (i_n - 1) \prod_{l=1}^{n-1} I_l.$$  

A generalization of the product of two matrices is the product of a tensor and a matrix [9, 15].

**Definition 2 (Mode-$m$ Product, $\times_m$).** The mode-$m$ product of a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M}$ and a matrix $B \in \mathbb{R}^{I_m \times I_M}$, denoted by $\mathcal{A} \times_m B$, is a tensor of dimensionality $\mathbb{R}^{I_1 \times \cdots \times I_{m-1} \times I_m \times I_{m+1} \times \cdots \times I_M}$ whose entries are

$$[A \times_m B]_{i_1, \ldots, i_{m-1}, j_I, i_{m+1}, \ldots, I_M} = \sum_{i_m} a_{i_1, \ldots, i_{m-1}, i_m, i_{m+1}, \ldots, I_M} b_{j_I, i_m}.$$  

The mode-$m$ product can be expressed in tensor notation, as $C = \mathcal{A} \times_m B$, or in terms of matrixized tensors, as $C_{[m]} = BA_{[m]}$.  

The M-mode SVD (aka. the Tucker decomposition) is a “generalization” of the conventional matrix (i.e., 2-mode) SVD which may be written in tensor notation as

$$D = U_1 \times_1 U_2 \times_2 \cdots \times_M U_M.$$  

**Algorithm 1** M-mode SVD algorithm.

**Input** the data tensor $D \in \mathbb{R}^{I_1 \times \cdots \times I_M}$.

1. For $m := 1, \ldots, M$.
   Let $U_m$ be the left orthonormal matrix of the SVD of $D_{[m]}$, the mode-$m$ matrixized $D$.

2. Set $Z := D \times_1 U_1^T \times_2 U_2^T \cdots \times_M U_M^T$.

**Output** mode matrices $U_1, \ldots, U_M$ and the core tensor $Z$.  

3 GLOBAL TENSOR FACTORIZATION

There are two classes of data tensor modeling techniques that stem from: the rank-$K$ decomposition (CANDECOMP/Parafac decomposition) and the multilinear rank-(R, R, \ldots, R), such as Tucker decomposition Multilinear-PCA, multilinear (tensor) ICA, plus various kernel variations that are doubly nonlinear [57].

![Figure 1: (a) TensorFaces representation: This image illustrates a multilinear tensor factorization of a 4th-order training data tensor, $D \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$, where $I_1, I_2, I_3, I_4$ are the number of pixels, illuminations, views and people, respectively. A vectorized image, $d$, is represented by a set of coefficient vectors, one for the illumination, viewpoint, and person, and expressed mathematically as $d = T \times_1 I^T \times_2 V^T \times_3 P^T$, where the TensorFaces basis, $T$, governs the interaction between different causal factors of data formation. (Data should be centered, but for this display it was added back.) (b) These set of images demonstrate the models ability to disentangle the causal factors. Illumination effects, such as highlights, cast shadows and shading [61] are progressively reduced. (c) Vlasic et al [65] generate new images by performing multilinear expression re-targeting.](image1)

![Figure 2: Matrixizing a (3rd-order) tensor. The tensor can be matrixized in 3 ways to obtain matrices comprising its 1-mode, 2-mode, and 3-mode vectors. Note that this matrixizing is not cyclical, unlike the one defined in [15].](image2)
3.1 Representation: Multilinear Tensor Factorization

Within the tensor mathematical framework, an ensemble of observations is organized in a higher order data tensor, $\mathcal{D}$. Given a data tensor $\mathcal{D}$ of labeled, vectorized training facial images $\mathbf{d}_{pce}$, where the subscripts denote the causal factors of facial image formation, the person $p$, view $v$, illumination $l$, and expression $e$ labels, the M-mode SVD [15, 16, 33, 56, 61] or its kernel variant [57] may be employed to multilinearly decompose the data tensor,

$$\mathcal{D} = \mathcal{T} \times_p \mathbf{U}_p \times_v \mathbf{U}_v \times_l \mathbf{U}_l \times_e \mathbf{U}_e,$$

and compute the mode matrices $\mathbf{U}_p$, $\mathbf{U}_v$, $\mathbf{U}_l$, and $\mathbf{U}_e$ that span the causal factor representation. The extended core computed by, $\mathcal{T} = \mathcal{D} \times_p \mathbf{U}_p^T \times_v \mathbf{U}_v^T \times_l \mathbf{U}_l^T \times_e \mathbf{U}_e^T$, governs the interaction between the causal factors (Figure 1). This approach makes the assumption that the representations for a causal factor are well modeled by a Gaussian distribution. An image $\mathbf{d}_{pce}$ is represented by a person, view, illumination and expression coefficient vectors as,

$$\mathbf{d}_{pce} = \mathcal{T} \times_p \mathbf{p}_p \times_v \mathbf{v}_v \times_l \mathbf{l}_l \times_e \mathbf{e}_e. \quad (5)$$

An important advantage of employing vectorized observations in a multilinear tensor framework is that all images of a person, regardless of viewpoint, illumination and expression are mapped to the same person coefficient vector, thereby achieving zero intra-class scatter. Thus, multilinear analysis creates well separated people classes by maximizing the ratio of inter-class scatter to intra-class scatter. Alternatively, one can employ the Multilinear (Tensor) Independent Component Analysis (MICA) algorithm [63]. M-mode ICA, which takes advantage of higher-order statistics to compute the mode matrices that span the causal factor representation, and the MICA basis tensor that governs their interaction.

3.2 Recognition: Multilinear Projection

While TensorFaces (MPCA) [59, 61] is a handy moniker for an approach that learns from an image ensemble the interaction and representation of various causal factors that determine observed data, with Multilinear (Tensor) ICA [63] as more sophisticated approach, none of the interaction models prescribe a solution for how one might determine the causal factors of a single unlabeled test image that is not part of the training set. Multilinear projection [58, 64] simultaneously projects one or more unlabeled test images that are not part of the training data set into multiple constituent causal factor spaces associated with data formation, in order to infer the mode labels:

$$\mathcal{R} = \mathcal{T}^{+} \mathbf{d}_{\text{test}} = \mathbf{r}_p \circ \mathbf{r}_v \circ \mathbf{r}_l \circ \mathbf{r}_e. \quad (6)$$

The multilinear projection of a facial image computes the illumination, view and person representation by taking advantage of the expected rank-1 structure of the response tensor $\mathcal{R}$, the unit vector constraint associated with $\mathbf{r}_p$, $\mathbf{r}_v$, $\mathbf{r}_l$, and $\mathbf{r}_e$ via a tensor decomposition using the CP-decomposition, i.e., a rank-1 tensor decomposition.

4 COMPOSITIONAL HIERARCHICAL TENSOR FACTORIZATION

In prior tensor based research, an imaged object was represented in terms of global causal factor representations that are not robust to occlusions. This section introduces a compositional hierarchical tensor factorization that derives its name from its ability to represent a hierarchy of intrinsic and extrinsic causal factors of data formation.

A hierarchical representation may be effectively employed to recognize occluded objects, including self-occlusion that occurs during out-of-plane rotation relative to the camera viewpoint. The efficacy of our approach is demonstrated by our LFW and Freiburg verification experiments that compare global and hierarchical representations in Section 5.

Within the tensor mathematical framework, an ensemble of training data is organized in a higher order data tensor, $\mathcal{D}$. A data tensor $\mathcal{D} \in \mathbb{R}^{I_0 \times I_t \times \cdots \times I_c}$ contains a collection of vectorized observation, $\mathbf{d}_{i_0}, \ldots, \mathbf{d}_{i_t} \in \mathbb{R}^{I_0}$ where each subscript, $i_c$, denotes one of the $C$ causal factors that have created the observation and have resulted in $I_0$ measurements, i.e., a total of $I_0$ pixels per image. In this paper, we will report results based on a data tensor $\mathcal{D}$ of labeled, vectorized training facial images $\mathbf{d}_{pce}$, where the subscripts denote the causal factors of facial image formation, the person $p$, view $v$, illumination $l$, and expression $e$ labels.

4.1 Hierarchical Data Tensor

We identify a general base case object and two special cases whose intrinsic and extrinsic causal factors we would like to represent. An object may be composed of (i) two partially overlapping children-parts and parent-whole that has data not contained in any of the children-parts, (ii) a set of non-overlapping parts, or (iii) a set of fully overlapping parts, which resembles to the rank-$(L,M,N)$ or a rank-$(L,M,.)$ block tensor decomposition [14], but which is too restrictive for our purpose. Figure 3 depicts the general base case and the two special base cases. In real scenarios, parent-wholes have children-parts that are recursively composed of children themselves, Fig. 4.

The data wholes, and parts are extracted by employing a filter bank, $\{\mathbf{H}_s | \mathbf{H}_s \in \mathbb{C}^{I_0 \times I_t \times \cdots \times I_c}, 1 \leq s \leq S\}$ where each 2D convolutional filter is implemented as a doubly (triply) block circulant matrix $\mathbf{H}_s$, where $\sum_{s=1}^{S} \mathbf{H}_s = \mathbf{I}$, and $s$ refers to the data segment. Convolution is a matrix-vector multiplication, between a circulant matrix and a vectorized observation, $\mathbf{d} = \mathbf{H}_s \mathbf{d}$ which in tensor notation is written as $\mathbf{d} = \mathbf{d} \times \mathbf{H}_s$, where mode 0 is the measurement mode. The segment data tensor, $\mathcal{D}_s = \mathcal{D} \times \mathbf{H}_s$, is the result of multiplying (convolving) every observation, $\mathbf{d}$, with the block circulant matrix (filter), $\mathbf{H}_s$. A filter $\mathbf{H}_s$ may be any of any type, or have any spatial scope. When a filter matrix is a block identity matrix, $\mathbf{H}_s = \mathbf{I}$, the filter matrix multiplication with a vectorized observation has the effect of segmenting a portion of the data without any blurring, sub-sampling or upsampling. Measurements associated with perceptual parts may may not be tightly packed into a block a priori, as in the case of vectorized images, but chunking may be achieved by a trivial permutation.

The data tensor, $\mathcal{D}$, is expressed in terms of its recursive hierarchy of wholes and parts by defining and employing a hierarchical data tensor, $\mathcal{D}_h$, that contains along its super-diagonal the collection of wholes and parts, $\mathcal{D}_h$, Fig. 3(a)

$$\mathcal{D} = \sum_{s=1}^{S} \mathcal{D} \times \mathbf{H}_s,$$}
where $I_x = [I_1, I_2, \ldots, I_5] \in \mathbb{R}^{l_x \times S_l}$ is a concatenation of 5 identity matrices, one for each data segment. The three different ways of rewriting $D$ in terms of a hierarchy of wholes and parts, eq. 7-9, results in three equivalent compositional hierarchical tensor factorizations:

$$D = \sum_{s=1}^{5} (Z \times_{0} U_{0s} \times_{1} U_{1s} \times_{2} U_{2s} \times_{3} U_{3s}) \times_{4} H_s$$

$$D_H = \sum_{s=1}^{5} (Z_H \times_{0} U_{0H} \times_{1} U_{1H} \times_{2} U_{2H} \times_{3} U_{3H})$$

The expression of $D$ in terms of a hierarchical data tensor is a mathematical conceptual device that enables a unified mathematical model of wholes and parts that can be expressed completely as a mode-m product (matrix-vector multiplication) and whose factorization can be optimized in a principled manner. Dimensionality reduction of the compositional representation is performed by optimizing

$$\epsilon = \frac{1}{2} \left\| D - (\tilde{Z}_H \times_{0} \tilde{U}_{0H} \times_{1} \tilde{U}_{1H} \times_{2} \tilde{U}_{2H} \times_{3} \tilde{U}_{3H}) \right\|$$

$$\times_{0} I_{l_0} \times_{1} I_{l_1} \times_{2} I_{l_2} \times_{3} \tilde{U}_{l_3}$$

$$\sum_{c=0}^{C} \lambda_c \left\| \tilde{U}_{cH} \tilde{U}_{cH}^T - I \right\|^2$$

where $\tilde{U}_{cH}$ is the composite representation of the $c$th mode, and $Z_H$ governs the interaction between causal factors. Our optimization may be initialized, only, by setting $Z_H$ and $U_{cH}$ to the M-mode SVD of $D_H$, and performing dimensionality reduction through truncation, where $U_{cH} \in \mathbb{R}^{S_{l_c} \times J_c}$ and $Z_H \in \mathbb{R}^{l_H \times \times J_c \times \times J_c}$ and $J_c \leq S_{l_c}$.

In the face recognition application we discuss later, $c = 0$ refers to the measurement mode, i.e., the pixel values in an image, and the range of values $0 \leq c \leq C$ refers to the $C$ causal factors.

For computational efficiency, we may perform M-mode SVD on each data tensor segment $D$, and concatenate terms along the diagonal of $Z_H$ and $U_{cH}$ for $c \leq C$. However, the most computationally efficient initialization, first, computes the M-mode SVD of $D$, multiplies (convolves) the core tensor, $Z$ with $H_s$ followed by a concatenation of terms along the diagonal of $Z_H$ and duplication of $U_{cH}$ along the diagonal of $U_{cH}$. The last initialization approach makes segment specific dimensionality reduction problematic, since part-based standard deviation, $\sigma_{l_0 \times l_1 \times \ldots}$, is not computed.
4.2 Compositional Hierarchical Factorization Derivation

For notational simplicity, we re-write the loss function as,
\[ e := \frac{1}{2} \| D - Z_{\mathcal{H}} \times_0 U_{\mathcal{H}} - \cdots \times_s U_{\mathcal{S}} \times_t U_{\mathcal{C}} \|^2 \\
+ \sum_{c=0}^{C} \sum_{s=1}^{S} \lambda_c \| U_{c,s}^T U_{c,s} - I \| \]  
(14)

where \( \hat{U}_{c,s} = I_{c,s} \odot \hat{G}_c \), \( \hat{G}_c \in \mathbb{R}^{I_{c,s} \times S_L} \) is permutation matrix that groups the columns of \( U_{c,s} \) based on the segment, \( s \), to which they belong, and the inverse permutation matrices have been multiplied into \( Z_{\mathcal{H}} \) resulting into a core that has also been grouped based on segments. The data tensor, \( D \), may be expressed in matrix form as in eq. 16 and reduces to the more efficiently block structure as in eq. 17
\[ D = Z_{\mathcal{H}} \times_0 U_{\mathcal{H}} \times_1 U_{\mathcal{S}} \times_2 U_{\mathcal{C}} \]  
(15)

\[ D_{[c]} = U_{c,0} Z_{\mathcal{H}} \odot U_{C,c} \odot U_{S,c} \odot U_{C,S,c} \odot U_{C,c,S} \odot U_{C,S,c} \odot U_{C,S,c} \]  
(16)

\[ U_{c,0} = U_{c,0} W_c^{T} = U_{c,0} W_c^{T} \]  
(17)

\[ \left[ \begin{array}{cccc}
Z_{[c]} & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 \\
0 & Z_{[c]} & 0 & \cdots \\
0 & 0 & \cdots & 0 \\
\end{array} \right] \left[ \begin{array}{c}
U_{C,c} \odot U_{S,c} \odot U_{C,S,c} \odot U_{C,S,c} \odot U_{C,S,c} \odot U_{C,S,c} \odot U_{C,S,c} \\
\end{array} \right] \]  
(18)

where \( \odot \) is the Kronecker product, and \( \times \) is the block-matrix Khatri-Rao product.8 The matricized block diagonal form of \( Z_{\mathcal{H}} \) in eq. 17 becomes evident when employing our modified data centric matrixizing operator based on the definition 1, where the initial mode is the measurement mode.

The compositional hierarchical tensor factorization algorithm computes the mode matrix, \( U_{c,s} \), by computing the minimum of \( e = \| D - Z_{\mathcal{H}} \times_0 U_{\mathcal{H}} \times_1 U_{\mathcal{S}} \times_2 U_{\mathcal{C}} \|^2 \) by cycling through the modes, solving for \( U_{c,s} \) in the equation \( \partial e / \partial U_{c,s} = 0 \) while holding the core tensor \( Z_{\mathcal{H}} \) and all the other mode matrices constant, and repeating until convergence. Note that
\[ \frac{\partial e}{\partial U_{c,s}} = \frac{\partial}{\partial U_{c,s}} \| D_{[c]} - U_{c,0} W_c^{T} \|^2 = -D_{[c]} W_c + U_{c,0} W_c^{T} W_c. \]  
(19)

Thus, \( \partial e / \partial U_{c,s} = 0 \) implies that
\[ U_{c,s} = D_{[c]} W_c^{-1} \left( W_c^{T} W_c \right)^{-1} = D_{[c]} W_c^{T}. \]  
(20)

\[ = D_{[c]} \left( Z_{\mathcal{H}} \odot (U_{c,0} \odot U_{c,1} \cdots \odot U_{c,l} \cdots \odot U_{c,L})^{T} \right)^{+} \]  
(21)

\[ \begin{bmatrix}
Z_{[c]} & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 \\
0 & Z_{[c]} & 0 & \cdots \\
0 & 0 & \cdots & 0 \\
\end{bmatrix} = D_{[c]} \left( U_{c,0} \odot U_{c,1} \cdots \odot U_{c,l} \cdots \odot U_{c,L} \right)^{T} \]  
(22)

whose \( U_{c,s} \) sub-matrices are then subject to orthonormality constraints.

Solving for the optimal core tensor, \( Z_{\mathcal{H}} \), the data tensor, \( D \), approximation is expressed in vector form as,
\[ e = \| \text{vec}(D) - \left( \hat{U}_{0} \odot \hat{U}_{1} \odot \cdots \odot \hat{U}_{S} \odot \hat{U}_{C} \right) \text{vec}(Z_{\mathcal{H}}) \|. \]  
(23)

Solve for the non-zero(nz) terms of \( Z_{\mathcal{H}} \) in the equation \( \partial e / \partial Z_{\mathcal{H}} = 0 \), by removing the corresponding zero columns of the first matrix on right side of the equation below, performing the pseudo-inverse, and setting
\[ \text{vec}(Z_{\mathcal{H}})_{nz} = (U_{c,0} \odot U_{c,1} \cdots \odot U_{c,l} \cdots \odot U_{c,L}^{T}) \text{vec}(D). \]  
(24)

Repeat all steps until convergence. This optimization is the basis of the Compositional Hierarchical Tensor Factorization, Algorithm 2.

Completely Overlapping Parts: When the data tensor is a collection of overlapping parts that have the same multilinear-rank reduction, Fig. 3e, the extended-core data tensor, \( \bar{D} \), computation in matrix form reduces to eq. 27
\[ \bar{D} = T_{\mathcal{H}} \times_0 U_{\mathcal{H}} \times_1 U_{\mathcal{S}} \times_2 U_{\mathcal{C}} \]  
(25)

\[ D_{[c]} = T_{\mathcal{L}_{[l]}} \left( (U_{c,0} \odot U_{c,1} \cdots \odot U_{c,l} \cdots \odot U_{c,L})^{T} \right)^{+} \]  
(26)

\[ \begin{bmatrix}
U_{c,0} \odot U_{c,1} \cdots \odot U_{c,L} \cdots \odot U_{c,L}^{T} \\
\end{bmatrix} = D_{[0]} \left( U_{c,0} \odot U_{c,1} \cdots \odot U_{c,l} \cdots \odot U_{c,L}^{T} \right) \]  
(27)

Independent Parts: When the data tensor is a collection of observations made up of non-overlapping parts, Fig. 3d, the data tensor decomposition reduces to the concatenation of a \( M \)-mode SVD of individual parts,
\[ D_{[c]} = \left( U_{c,0} \odot U_{c,1} \cdots \odot U_{c,l} \cdots \odot U_{c,L}^{T} \right)^{T} \]  
(28)

Note, that every \( D_{[c]} \) row does not contain any terms from any other segment-part except segment \( s \). Thus, every \( U_{c,s} \), and \( Z_{\mathcal{H}} \), are computed by performing multilinear subspace learning, Algorithm 1, on the \( D_{[c]} \), and the results are appropriately concatenated in \( U_{c,s} \) and \( Z_{\mathcal{H}} \).
Facial images were warped to an average face

Models were trained on approximately half of one percent (\(0.5\%\)) of Freiburg \[6\]. Each subject was imaged in Maya from 15 different viewpoints (\(\theta = -60^\circ\) to \(+60^\circ\) in \(10^\circ\) steps on the horizontal plane, \(\phi = 0^\circ\)) with 15 different illuminations (\(\theta = -35^\circ\) to \(+35^\circ\) in \(5^\circ\) increments on a plane inclined at \(\phi = 45^\circ\)).

In our experiments, we employed gray-level facial images from the University of Freiburg 3D Morphable Faces dataset [6]. Each subject was imaged from 15 different viewpoints (\(\theta = -60^\circ\) to \(+60^\circ\) in \(10^\circ\) steps on the horizontal plane, \(\phi = 0^\circ\)) with 15 different illuminations (\(\theta = -35^\circ\) to \(+35^\circ\) in \(5^\circ\) increments on a plane inclined at \(\phi = 45^\circ\)).

Training Data: In our experiments, we employed gray-level facial training images rendered from 3D scans of 100 subjects. The scans were recorded using a CyberwareTM 3030PS laser scanner and are part of the 3D morphable faces database created at the University of Freiburg [6]. Each subject was imaged in Maya from 15 different viewpoints (\(\theta = -60^\circ\) to \(+60^\circ\) in \(10^\circ\) steps on the horizontal plane, \(\phi = 0^\circ\)) with 15 different illuminations (\(\theta = -35^\circ\) to \(+35^\circ\) in \(5^\circ\) increments on a plane inclined at \(\phi = 45^\circ\)).

Data Preprocessing: Facial images were warped to an average face template by a piecewise affine transformation given a set of facial landmarks obtained by employing Dlib software [29, 31, 50]. Illumination was normalized with an adaptive contrast histogram equalization algorithm, but rather than performing contrast correction on the entire image, subtiles of the image were contrast normalized, and tiling artifacts were eliminated through interpolation. Histogram clipping was employed to avoid over-saturated regions. Each image, \(d \in \mathbb{R}^{4096}\), was convolved with a set of filters \(\{H_s\}_{s=1, \ldots, 5}\), and the filtered images, \(d \times H_s\), resulted either in a Gaussian or Laplacian data pyramid. Facial parts were extracted from the various layers.

5 COMPOSITIONAL HIERARCHICAL TENSORFACES

Table 1: Empirical results reported for LFW : PCA, TensorFaces and Compositional Hierarchical TensorFaces. Pixels denotes independent facial part analysis. Gaussian/Laplacian refers to using Compositional Hierarchical TensorFaces on a Gaussian/Laplacian data pyramid. Freiburg Experiment:

| Test Dataset | PCA | TensorFaces | Compositional Hierarchical TensorFaces |
|--------------|-----|-------------|---------------------------------------|
|              | Pixels | Gaussian Pyramid | Weighted Gaussian Pyramid | Laplacian | Weighted Laplacian |
| Freiburg     | 65.23% | 90.50% | 88.17% | 77.85% | 94.17% | 90.96% | 77.58% | 93.98% |
| LFW          | 69.23% | 72.72% | 76.72% | 77.85% | 91.16% | 87.58% | 87.38% | 87.38% |

Test on Freiburg: 6 views (+\(60^\circ, +30^\circ, +5^\circ\)); 6 illuminations (+\(60^\circ, +30^\circ, +5^\circ\)), 45 people

Test on Freiburg: 9 views (+\(50^\circ, +40^\circ, +20^\circ, +10^\circ, 0^\circ\)), 9 illum (+\(50^\circ, +40^\circ, +20^\circ, +10^\circ, 0^\circ\)), 45 different people

LFW Experiment: Models were trained on approximately half of one percent (0.5\% \(\pm 1\%\)) of the 4.4M images used to train DeepFace.

Train on Freiburg:

15 views (+\(60^\circ, +50^\circ, +40^\circ, +30^\circ, +20^\circ, +10^\circ, +5^\circ, 0^\circ\)), 15 illuminations (+\(60^\circ, +50^\circ, +40^\circ, +30^\circ, +20^\circ, +10^\circ, +5^\circ, 0^\circ\)), 100 people

Test on LFW: We report the mean accuracy and standard deviation across standard literature partitions [25], following the Unrestricted, labeled outside data supervised protocol.
Experiments: The composite person signature was computed for every test image by employing the multilinear projection algorithm [58, 64], and signatures were computed with weighted nearest neighbor.

To validate the effectiveness of our system on real-world images, we report results using the “Labeled Faces in the Wild” dataset (LFW) [25]. This dataset contains 13,233 facial images of 5,749 people. The photos are unconstrained (i.e., “in the wild”), and include variation due to pose, illumination, expression, and occlusion. The dataset consists of 10 train/test splits of the data. We report the mean accuracy and standard deviation across all splits (Table 1, Fig. 4(b-c) depicts our ROC curves). We follow the supervised “Unrestricted, labeled outside data” paradigm.

Results: While we cannot celebrate closing the gap on human performance, our results are promising. DeepFace, a CNN model, improved the prior art verification rates on LFW from 70% to 97.35%, by training on 4.4M images of 200 × 200 pixels from 4,030 people, the same order of magnitude as the number of people in the LFW database. We trained on less than one percent (1%) of the 4.4M total images used to train DeepFace. Images were synthetically rendered by Maya from 3D scans of 100 subjects [6], with an the intraocular distance of approximately 20 pixels and with a facial region captured by 10,414 pixels (image size ≈ 100 × 100 pixels). We have currently achieved verification rates just shy of 80% on LFW. When data is limited CNN models do not converge or generalize.

6 CONCLUSION

In analogy to autoencoders which are inefficient neural network implementation of principal component analysis, a pattern analysis method based in linear algebra, CNNs are neural network implementations of tensor factorizations. This paper contributes to the tensor algebraic paradigm and models cause-and-effect as multilinear tensor interaction between intrinsic and extrinsic hierarchical causal factors of data formation. Causal factor representations are interpretable, hierarchical and statistically invariant to all other causal factors. The data tensor is re-conceptualized into a hierarchical data tensor; a unified tensor model of wholes and parts is proposed; and a new compositional hierarchical tensor factorization is derived. Our approach is demonstrated in the context of facial images by training on a very small set of synthetically rendered images. While we have not closed the gap on human performance, we report encouraging face verification results on two test data sets âÅS the Freiburg, and the Labeled Faces in the Wild (LFW) datasets. CNNs verification rates improved the 70% prior art to 97.35% when they employed 4.4M images from 4,030 people, the same order of magnitude as the number of people in the LFW database. We have currently achieved verification rates just shy of 80% on LFW by employing synthetic images from 100 people for a total of less than one percent (1%) of the total images employed by DeepFace. By comparison, when data is limited CNN models do not converge or generalize.

REFERENCES

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, 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. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski. Face recognition by independent component analysis. IEEE Transactions on Neural Networks, 13(6):1450–1464, 2002.

[3] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, and Y. Bengio. Theano: A CPU and GPU math expression compiler. In Proceedings of the Python for Scientific Computing Conference (SciPy), June 2010. Oral Presentation.

[4] V. Blanz and T. A. Vetter. Morphable model for the synthesis of 3D faces. In Proc. ACM SIGGRAPH ’99 Conf., pages 187–194, 1999.

[5] C. Cadieu and B. Olshausen. Learning transformational invariants from natural movies. In Proceedings of the 19th International Conference on Neural Information Processing Systems, NIPS’09, page 209–216, 2009.

[6] J. D. Carroll and J. J. Chang. Analysis of individual differences in multidimensional scaling via an N-way generalization of Eckart-Young decomposition. Psychometrika, 35:283–319, 1970.

[7] J. D. Carroll, S. Pruzansky, and J. B. Kruskal. CANDELMAN: A general approach to multidimensional analysis of many-way arrays with linear constraints on parameters. Psychometrika, 35:283–319, 1970.

[8] J. C. Chen, R. Ranjan, A. Kumar, C. H. Chen, V. M. Patel, and R. Chellappa. An end-to-end system for unconstrained face verification with deep convolutional neural networks. In 2013 IEEE International Conference on Computer Vision Workshop (ICCVW), pages 360–368, Dec 2015.
