# On the Evaluation of Tensor-Based Representations for Optimum-Path Forest Classification

3 authors:

Ricardo Ricci Lopes
Amsterdam University Medical Center
**3** PUBLICATIONS   **1** CITATION

SEE PROFILE

Kelton A. P. da Costa
University of São Paulo
**39** PUBLICATIONS   **258** CITATIONS

SEE PROFILE

João Paulo Papa
São Paulo State University
**306** PUBLICATIONS   **3,130** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

OPF - Others View project

Drilling Analysis View project

# On the Evaluation of Tensor-based Representations for Optimum-Path Forest Classification

Ricardo Lopes[1], Kelton Costa[2], and João Papa[2]

[1] Instituto de Pesquisas Eldorado, Campinas
ricardoriccilopes@gmail.com
[2] Department of Computing, São Paulo State University
papa@fc.unesp.br,kelton.costa@gmail.com

**Abstract.** Tensor-based representations have been widely pursued in the last years due to the increasing number of high-dimensional datasets, which might be better described by the multilinear algebra. In this paper, we introduced a recent pattern recognition technique called Optimum-Path Forest (OPF) in the context of tensor-oriented applications, as well as we evaluated its robustness to space transformations using Multilinear Principal Component Analysis in both face and human action recognition tasks considering image and video datasets. We have shown OPF can obtain more accurate recognition rates in some situations when working on tensor-oriented feature spaces.

**Keywords:** Optimum-Path Forest; Tensors; Gait and Face Recognition

## 1 Introduction

Methodologies for data representation have been widely pursued in the last decades, being most part of them based on *Vector Space Models* (VSM). Roughly speaking, given a dataset $\mathcal{X}$ e a label set $\mathcal{Y}$, each sample $\mathbf{x}_i \in \mathcal{X}$ is represented as an $n$-dimensional point on that space with a label associated to it, i.e., each sample can be modelled as a pair $(\mathbf{x}_i, y_i)$, $y_i \in \mathbb{N}$ and $i = 1, 2, \ldots, |\mathcal{X}|$. Therefore, a machine learning algorithm aims at finding a decision function $f : \mathcal{X} \rightarrow \mathcal{Y}$ that leads to the best feature space partition [1].

Advances in storage technologies have fostered an increasing number of large data repositories, composed mainly of high-resolution images and videos. Such new environments now require more efficient and effective data representation and classification approaches, which shall take into account the high-dimensionality nature of the data [5]. Images acquired through cell phones, for instance, may contain thousands of hundreds of pixels, being 2-dimensional data by nature. As such, a specific segment of researchers have devoted a considerable effort to study more natural data representation systems. One of the most actively data description approaches is related to the well-known *Tensor Space Model* (TSM), in which a dataset sample is represented as a tensor instead of a regular point,

being the properties of such space ruled by the multilinear algebra. Roughly speaking, we can consider an image as a 2-order tensor (matrix), a video as a 3-order tensor (cube), and a scalar number is considered an 1-order tensor. Therefore, tensor-based representations can be understood as a generalization of vector-space models.

Although one can find a number of tensor-based machine learning works out there, they are responsible for only a few percentage of the published literature. Vasilescu and Terzopoulos [13], for instance, used tensorial models to dimensionality reduction in the context of face-oriented person identification. Other works focused on the extension of some well-known techniques in computer vision, i.e., Principal Component Analysis and Linear Discriminant Analysis, to tensor models [4]. In addition, Sutskever et al. [12] employed tensorial factorization together with Bayesian clustering to learn relational structures in text recognition, and Ranzato et al. [10] used a deep learning-based approach parameterized by tensors in the context of image processing.

Later on, Cai et al. [2] introduced the Support Tensor Machines (STMs) for text categorization, which is a tensor-based variant of the so-called Support Vector Machines (SVMs) classifier. The original samples were mapped as 2-order tensors, and the problem of identifying the hyperplane with maximum margin in the vector space was then changed to a tensor space. A tensor-oriented neural network was also considered for text recognition by Socher et al. [11]. As aforementioned, one can notice a lack of research regarding tensor-based machine learning, since only a few techniques have been considered in such context.

Some years ago, Papa et al. [7, 6] introduced a new pattern recognition technique called Optimum-Path Forest, which models the problem of pattern classification as a graph partition task, in which each dataset sample is encoded as a graph node and connected to others through an adjacency relation. The main idea is to rule a competition process among some key samples (*prototypes*) that try to conquer the remaining nodes in order to partition the graph into optimum-path trees, each one rooted at one prototype. In this paper, we introduce OPF in the context of tensor-space learning, since it has never been evaluated in such representation model so far. We present some results in the context of face recognition using an image-oriented dataset, as well as human action recognition in video data. The remainder of this paper is organized as follows. Sections 2 and 3 present a theoretical background regarding OPF and the methodology and experiments, respectively. Finally, Section 4 states conclusions and future works.

## 2 Optimum-Path Forest Classification

Let $\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2$ be a labeled dataset, such that $\mathcal{D}_1$ and $\mathcal{D}_2$ stands for the training and test sets, respectively. Let $\mathcal{S} \subset \mathcal{D}_1$ be a set of prototypes of all classes (i.e., key samples that best represent the classes). Let $(\mathcal{D}_1, A)$ be a complete graph whose nodes are the samples in $\mathcal{D}_1$, and any pair of samples defines an arc in

$A = \mathcal{D}_1 \times \mathcal{D}_1$. Additionally, let $\pi_s$ be a path in $(\mathcal{D}_1, A)$ with terminus at sample $s \in D_1$.

The OPF algorithm proposed by Papa et al. [7, 6] employs the path-cost function $f_{max}$ due to its theoretical properties for estimating prototypes (Section 2.1 gives further details about this procedure):

$$f_{max}(\langle s \rangle) = \begin{cases} 0 & \text{if } s \in S \\ +\infty & \text{otherwise,} \end{cases}$$
$$f_{max}(\pi_s \cdot \langle s, t \rangle) = \max\{f_{max}(\pi_s), d(s, t)\}, \tag{1}$$

where $d(s, t)$ stands for a distance between nodes $s$ and $t$, such that $s, t \in \mathcal{D}_1$. Therefore, $f_{max}(\pi_s)$ computes the maximum distance between adjacent samples in $\pi_s$, when $\pi_s$ is not a trivial path. In short, the OPF algorithm tries to minimize $f_{max}(\pi_t)$, $\forall t \in \mathcal{D}_1$.

### 2.1 Training

We say that $\mathcal{S}^*$ is an optimum set of prototypes when the OPF algorithm minimizes the classification errors for every $s \in \mathcal{D}_1$. We have that $S^*$ can be found by exploiting the theoretical relation between the minimum-spanning tree and the optimum-path tree for $f_{max}$. The training essentially consists of finding $S^*$ and an OPF classifier rooted at $S^*$. By computing a Minimum Spanning Tree (MST) in the complete graph $(\mathcal{D}_1, A)$, one obtain a connected acyclic graph whose nodes are all samples of $\mathcal{D}_1$ and the arcs are undirected and weighted by the distances $d$ between adjacent samples. In the MST, every pair of samples is connected by a single path, which is optimum according to $f_{max}$. Hence, the minimum-spanning tree contains one optimum-path tree for any selected root node.

The optimum prototypes are the closest elements of the MST with different labels in $\mathcal{D}_1$ (i.e., elements that fall in the frontier of the classes). By removing the arcs between different classes, their adjacent samples become prototypes in $S^*$, and the OPF algorithm can define an optimum-path forest with minimum classification errors in $\mathcal{D}_1$.

### 2.2 Classification

For any sample $t \in \mathcal{D}_2$, we consider all arcs connecting $t$ with samples $s \in \mathcal{D}_1$, as though $t$ were part of the training graph. Considering all possible paths from $S^*$ to $t$, we find the optimum path $P^*(t)$ from $S^*$ and label $t$ with the class $\lambda(R(t))$ of its most strongly connected prototype $R(t) \in S^*$. This path can be identified incrementally, by evaluating the optimum cost $C(t)$ as follows:

$$C(t) = \min\{\max\{C(s), d(s, t)\}\}, \ \forall s \in D_1. \tag{2}$$

Let the node $s^* \in \mathcal{D}_1$ be the one that satisfies Equation 2 (i.e., the predecessor $P(t)$ in the optimum path $P^*(t)$). Given that $L(s^*) = \lambda(R(t))$, the classification simply assigns $L(s^*)$ as the class of $t$. An error occurs when $L(s^*) \neq \lambda(t)$.

## 3 Experimental Evaluation

In this section, we present the methodology and experiments used to validate OPF in the context of tensor-based feature representation.

### 3.1 Datasets

We considered two public datasets, as follows:

- Gait-based Human ID[3]: this dataset comprises 1,870 sequences from 122 individuals aiming at the automatic identification of humans from gait. Since this dataset is composed of videos, it has an interesting scenario for the application of tensor representations; and
- AT&T Face Dataset[4]: formerly "ORL Dataset", it comprises $92 \times 112$ images of human faces from 40 subjects.

The "Gait-based Human ID" dataset has seven fixed scenarios, called PrbA, PrbB, PrbC, PrbD, PrbE, PrbF and PrbG. In such case, the algorithms are trained in a "Gallery" set, and then tested on each scenario. In addition to that, we employed a cross-validation procedure over the entire dataset (i.e., PrbA $\cup$ PrbB $\cup \ldots \cup$ PrbG) for comparison results. Notice the cross-validation has been applied to "AT&T Face dataset" as well. Figure 1 displays some examples of dataset samples.
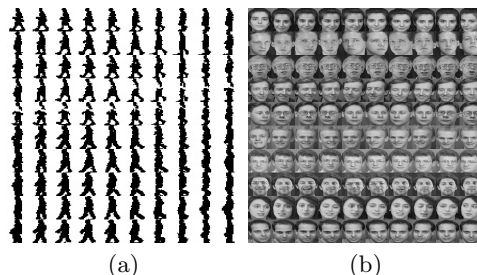


(a)          (b)

**Fig. 1.** Some dataset samples from (a) Gait-based Human ID and (b) AT&T Face datasets.

### 3.2 Experiments

In this work, we compared OPF [8] in two distinct scenarios, VSM and TSM, i.e., vector- and tensor-space models, respectively. Additionally, we evaluated SVM to the same context using RBF (Radial Basis Function), Linear, Polynomial

---

[3] http://figment.csee.usf.edu/GaitBaseline/

[4] http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

(Poly) and sigmoid (Sig) kernels, as well as SVM without kernel mapping (SVM-noKernel). In regard to SVM with kernel functions, we employed the well-known LIBSVM [3], and with respect to SVM without kernel mapping, we employed LIBLINEAR library [9]. Finally, we used an accuracy rate proposed by Papa et al. [7] that considers unbalanced datasets.

**Gait-based Human ID** In this section, we present the results considering the Gait-based Human ID dataset in two distinct rounds of experiments: (i) in the first one, called "fixed data", we used the very same aforementioned configuration composed of seven fixed scenarios; and (ii) in the latter experiment, we performed a cross-validation with random generated folds (we called here "random data"). Table 1 presents the results considering the "fixed data" experiment. The techniques labeled with "PCA" stand for vector-space models, and the ones labeled with "MPCA" and "MPCA-LDA" denote the tensor-space modeling. In the latter approach, a Linear Discriminant Analysis was performed after MPCA. The most accurate techniques are highlighted in bold.

**Table 1.** Mean recognition rates considering the "fixed data" experiment for the Gait-based Human ID dataset.

| | PrbA | PrbB | PrbC | PrbD | PrbE | PrbF | PrbG |
|---|---|---|---|---|---|---|---|
| OPF-PCA | 50.59% | 49.59% | 49.91% | 50.59% | 49.65% | 50.36% | 50.56% |
| OPF-MPCA | 74.55% | 73.73% | 63.70% | 59.50% | 50.00% | 54.66% | 54.62% |
| OPF-MPCA-LDA | 86.39% | 50.84% | 70.74% | 60.66% | 49.82% | **57.18%** | **56.67%** |
| SVM-noKernel-PCA | 50,64% | 49,66% | 49,86% | 50,55% | 50,03% | 50,38% | 50,09% |
| SVM-Linear-PCA | 51,32% | 49,37% | 49,93% | 50,68% | 49,66% | 50,67% | 51,26% |
| SVM-Poly-PCA | 50,63% | 49,52% | 49,65% | 50,16% | 49,79% | 50,85% | 50,30% |
| SVM-RBF-PCA | 51,32% | 49,37% | 49,93% | 50,68% | 49,66% | 50,67% | 51,27% |
| SVM-Sig-PCA | 51,32% | 49,37% | 49,93% | 50,68% | 49,66% | 50,67% | 51,26% |
| SVM-noKernel-MPCA | 86.22% | **79.52%** | 70.81% | 58.81% | 50.27% | 56.82% | 54.51% |
| SVM-Linear-MPCA | 87.20% | 78.93% | 70.81% | 60.68% | 50.10% | 56.69% | 55.86% |
| SVM-Poly-MPCA | 70.28% | 67.08% | 59.92% | 56.31% | 49.61% | 52.86% | 52.61% |
| SVM-RBF-MPCA | 87.06% | 79.06% | 70.81% | 60.68% | 50.10% | 56.63% | 55.87% |
| SVM-Sig-MPCA | 87.20% | 78.93% | 70.81% | 60.68% | 50.10% | 56.69% | 55.86% |
| SVM-noKernel-MPCA-LDA | 84.86% | 51.29% | 69.34% | 59.63% | 50.10% | 56.17% | 54.82% |
| SVM-Linear-MPCA-LDA | 88.52% | 50.92% | 71.09% | 61.58% | 50.30% | 56.64% | 55.26% |
| SVM-Poly-MPCA-LDA | 68.78% | 49.91% | 55.07% | 55.32% | 50.16% | 53.30% | 52.00% |
| SVM-RBF-MPCA-LDA | **88.68%** | 51.03% | 70.99% | **61.73%** | 50.30% | 56.69% | 55.26% |
| SVM-Sig-MPCA-LDA | 88.20% | 50.90% | **71.25%** | 61.51% | **50.60%** | 56.64% | 55.73% |

From those results, some interesting conclusions can be drawn: (i) tensor-space models have obtained the best results for both classifiers, i.e., OPF and SVM, (ii) SVM obtained the best results in 5 out 7 folds, and (iii) OPF can benefit from tensor-space models, which is the main contribution of this paper. In addition, OPF results were very close to SVM ones, but being much faster for training, since it is parameterless and thus not require an optimization procedure.

In the second round of experiments, we evaluated OPF for tensor-space models using randomly generated folds in two distinct configurations: the first one employs 10% of the whole dataset for training, and the another one that uses 50% of the samples to train the classifiers. Table 2 presents the results considering the aforementioned configurations. Since we have considered a cross-validation procedure over 10 runnings, we performed a statistical validation through Wilcoxon signed-rank test [14].

**Table 2.** Mean recognition rates considering the "random data" experiment for the Gait-based Human ID dataset. The values in bold stand for the most accurate techniques according to that test.

|  | 10% | 50% |
|---|---|---|
| OPF-PCA | 60.86%±0.41 | 76.92%±0.50 |
| OPF-MPCA | 67.77%±0.47 | 80.55%±0.45 |
| OPF-MPCA-LDA | 63.72%±0.47 | 73.15%±0.42 |
| SVM-noKernel-PCA | 55.68%±0.21 | 56.07%±0.26 |
| SVM-Linear-PCA | 57.79%±0.30 | 65.03%±0.50 |
| SVM-Poly-PCA | 55.27%±0.55 | 70.25%±0.44 |
| SVM-RBF-PCA | 59.06%±0.38 | 73.81%±0.48 |
| SVM-Sig-PCA | 57.62%±0.40 | 65.04%±0.58 |
| SVM-noKernel-MPCA | 72.16%±0.57 | 86.89%±0.42 |
| SVM-Linear-MPCA | **74.06%**±0.66 | **91.42%**±0.48 |
| SVM-Poly-MPCA | 61.07%±1.04 | 83.29%±0.39 |
| SVM-RBF-MPCA | 73.95%±0.61 | **91.41%±0.44** |
| SVM-Sig-MPCA | 74.01%±0.69 | **91.38%±0.43** |
| SVM-noKernel-MPCA-LDA | 66.55%±0.58 | 76.24%±0.41 |
| SVM-Linear-MPCA-LDA | 67.89%±0.54 | 79.98%±0.56 |
| SVM-Poly-MPCA-LDA | 53.77%±1.61 | 69.93%±0.83 |
| SVM-RBF-MPCA-LDA | 68.57%±0.57 | 81.68%±0.52 |
| SVM-Sig-MPCA-LDA | 67.95%±0.60 | 79.59%±0.68 |

Considering this experiment, we can clearly observe SVM-Linear-MPCA with tensor-space modeling has obtained better results using 10% of the entire dataset for training purposes. However, if we take into account 50% of the data for training, only OPF with MPCA outperformed the standard vector-space modeling (i.e., OPF-PCA), since OPF-MPCA-LDA did not achieve better results than OPF-PCA. This is might be due to the poor mapping performed by LDA when considering a bigger training set. Finally, SVM also benefit from tensor-based features, achieving better results than OPF as well.

**AT&T Face Dataset** In this section, we evaluated vector- and tensor-space models considering the task of face recognition. Once again, we employed two distinct configurations, with the first one using 10% of the dataset samples for training, and the another one using 50% to train the classifiers. Table 3 presents

the mean recognition rates through a cross-validation procedure with 10 runnings. Similar techniques according to Wilcoxon statistical test are highlighted in bold.

**Table 3.** Mean recognition rates considering AT&T Face Dataset.

|  | 10% | 50% |
|---|---|---|
| OPF-PCA | **83.97%±1.34** | 96.54%±0.74 |
| OPF-MPCA | **84.52%±0.85** | 96.49%±0.94 |
| OPF-MPCA-LDA | 61.11%±1.52 | 77.54%±1.34 |
| SVM-noKernel-PCA | **83.93%±0.93** | 96.28%±0.75 |
| SVM-Linear-PCA | **83.97%±1.34** | **97.90%±0.55** |
| SVM-Poly-PCA | 59.95%±2.48 | 88.49%±1.27 |
| SVM-RBF-PCA | **83.97%±1.34** | **97.46%±1.16** |
| SVM-Sig-PCA | **83.62%±1.98** | **97.79%±0.76** |
| SVM-noKernel-MPCA | 83.46%±1.23 | 96.10%±0.86 |
| SVM-Linear-MPCA | **84.17%±0.97** | **97.85%±0.63** |
| SVM-Poly-MPCA | 65.61%±1.32 | 92.51%±1.20 |
| SVM-RBF-MPCA | 50.00%±0.00 | 94.49%±2.13 |
| SVM-Sig-MPCA | 51.14%±0.50 | 52.03%±0.92 |
| SVM-noKernel-MPCA-LDA | 57.95%±1.10 | 73.72%±1.26 |
| SVM-Linear-MPCA-LDA | 61.11%±1.52 | 79.00%±1.29 |
| SVM-Poly-MPCA-LDA | 53.02%±0.44 | 68.92%±1.66 |
| SVM-RBF-MPCA-LDA | 54.39%±1.05 | 77.10%±1.76 |
| SVM-Sig-MPCA-LDA | 51.13%±0.55 | 62.85%±0.95 |

The results showed OPF-MPCA has obtained the best results using 10% of the dataset for training, but with OPF-PCA e some SVM variations with similar results. Considering 50% of the dataset, SVM achieved the best results, but it seems the tensor-based representation did not play an important role in this experiment, although it has achieved very good results. A possible idea to handle such problem would be to extract features from images, and then map such features to tensor-space models, since in this work we adopted a holistic-based face recognition, i.e., we used the pixels' intensities for pattern classification purposes.

## 4 Conclusions

Tensor-based representations considering machine learning-oriented applications have been pursued in the last years aiming to obtain a more realistic and natural representation of high-dimensional data. In this paper, we evaluated the performance of OPF classifier in the context of tensor-space models. The experiments involved two distinct scenarios: gait classification in video images, and face recognition in gray-scale images. We can conclude OPF classifier can benefit from such tensor-based feature space representations.

## Acknowledgment

## References

1. Cai, D., He, X., Han, J.: Learning with tensor representation. Tech. rep., University of Illinois at Urbana-Champaign, Department of Computer Science (2006)
2. Cai, D., He, X., J.-R., W., Han, J., W.-Y., M.: Suport tensor machines for text categorization. Tech. rep., University of Illinois at Urbana-Champaign, Department of Computer Science (2006)
3. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2, 27:1–27:27 (2011), software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`
4. He, X., Cai, D., Niyogi, P.: Tensor subspace analysis. In: Weiss, Y., Schölkopf, B., Platt, J. (eds.) Advances in Neural Information Processing Systems 18. pp. 499–506. MIT Press (2006)
5. Lu, H., Plataniotis, K.N., Venetsanopoulos, A.N.: A survey of multilinear subspace learning for tensor data. Pattern Recognition 44(7), 1540–1551 (2011)
6. Papa, J.P., Falcão, A.X., Albuquerque, V.H.C., Tavares, J.M.R.S.: Efficient supervised optimum-path forest classification for large datasets. Pattern Recognition 45(1), 512–520 (2012)
7. Papa, J.P., Falcão, A.X., Suzuki, C.T.N.: Supervised pattern classification based on optimum-path forest. International Journal of Imaging Systems and Technology 19, 120–131 (2009)
8. Papa, J.P., Suzuki, C.T.N., Falcão, A.X.: LibOPF: A library for the design of optimum-path forest classifiers (2014), software version 2.1 available at `http://www.ic.unicamp.br/~afalcao/LibOPF`
9. R.-E. Fan, K.-W. Chang, C.J.H.X.R.W., Lin, C.J.: LIBLINEAR: A library for large linear classification journal of machine learning research 9, 1871–1874 (2008)
10. Ranzato, M., Krizhevsky, A., Hinton, G.E.: Factored 3-way restricted boltzmann machines for modeling natural images. In: Thirteenth International Conference on Artificial Intelligence and Statistics. JMLR Proceedings, vol. 9, pp. 621–628. JMLR.org (2010)
11. Socher, R., Perelygin, A., Wu, J.Y., Chuang, J., Manning, C.D., Ng, A.Y., Potts, C.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Conference on Empirical Methods in Natural Language Processing. pp. 1631–1642 (2013)
12. Sutskever, I., Tenenbaum, J.B., Salakhutdinov, R.: Modelling relational data using bayesian clustered tensor factorization. In: Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C., Culotta, A. (eds.) Advances in Neural Information Processing Systems 22, pp. 1821–1828. Curran Associates, Inc. (2009)
13. Vasilescu, M.A.O., Terzopoulos, D.: Multilinear subspace analysis of image ensembles. In: IEEE Conference on Computer Vision and Pattern Recognition. vol. 2, pp. 93–9 (2003)
14. Wilcoxon, F.: Individual comparisons by ranking methods. Biometrics Bulletin 1(6), 80–83 (1945)