Relative Margin Support Tensor Machines for gait and action recognition

Irene Kotsia and Ioannis Patras
School of Electronic Engineering and Computer Science
Queen Mary University of London, UK, Mile End Road, London E1 4NS
{irene.kotsia,i.patras}@elec.qmul.ac.uk

ABSTRACT

In this paper, we formulate the Relative Margin Support Tensor Machines (RMSTMs) problem, as an extension of the Relative Margin Machines (RMMs). While the typical Support Tensor Machines (STMs) aim at finding a solution that is greatly influenced by the data spread, the proposed RMSTMs maximize the margin in a way relative to the spread of the data. The efficiency of the proposed method is illustrated on the problems of gait and action recognition, where the results acquired verify the superiority of the method in terms of classification performance.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Delphi theory

Keywords

Σ-Support Vector Machines, Relative Margin Support Vector Machines, Multilinear Support Tensor Machines, **Σ**-Support Tensor Machines, Relative Margin Support Tensor Machines,

1. INTRODUCTION

The constantly increasing need for data storage, a result of the continuously growing size of high dimensional data, has made the need for proper information representations a vital one, constituting tensors an ideal choice. Indeed, during the past few years, tensors have greatly attracted the interest of scientific community, causing research to make a turn towards their direction.

Tensors are efficient representations of multidimensional objects, whose elements can be accessed with two or more indices. The number of indices required to access a tensor defines its order, while each index specifies a mode. Thus, tensors are ideal when it comes to representing images (2nd order tensors), gait sequences (3rd order tensors), color videos (4th order tensors) etc. They can be widely used in many areas, such as 3D object recognition, 3D face reconstruction, medical image analysis, activity recognition, gait recognition etc.

Thus, during the past few years, various fundamental methods have been extended to handle tensors. Among them the Multilinear Principal Component Analysis (MPCA) [1] (extension of PCA), Support Tensor Machines (STMs) [2] (extension of Support Vector Machines (SVMs)) and Canonical Analysis Correlation of tensors [3].

More specifically, most of the methods developed for gait recognition during the past few years have used gait silhouette sequences [4], [5], [6], [7] due to their easy extraction. Several studies have shown that gait silhouettes are ideal for object or person recognition when they are detailed enough, as humans can recognize readily the identity of the object or person under investigation [8]. Lately however, gait silhouette sequences have been regarded as 3 order tensors, thus enabling their manipulation using multilinear techniques [1].

Most common techniques regarding gait recognition include Hidden Marcov Models (HMMs) [7], Linear Time Normalization [4], Gait Energy Images [6], Multilinear PCA [1] and temporal correlation of extracted gait silhouette sequences [5] (the above mentioned methods constitute the ones we will compare our results with, in order to prove the superiority of the proposed method in terms of classification accuracy).

Regarding action recognition, the research conducted in the past years can be distinguished in three categories: articulated model-based approaches [9], feature-based approaches [10], and template-based approaches [11], [12]. The first category includes methods that use articulated models and perform recognition based on those acquired models. The second category is based on the tracking of landmark points, thus performing recognition based on their acquired trajectories. The third category contains silhouettes, hand shapes etc. The temporal transitions of the above mentioned templates is studied to achieve recognition. Recently, the advantages of tensors have been studied in [3] in order to achieve action recognition using video tensors.

In this paper we propose the extension of Relative Margin Support Vector Machines (RMVMs) to Relative Margin Support Tensor Machines (RMSTMs). The typical STMs provide a solution that is based on finding the maximum margin that separates the data, ignoring their spread. The way to find the maximum magrin solution while taking under consideration the spread of the data instead of regarding the absolute large margin solution, is presented in this paper. Thus, novel formulations that deal with the above mentioned case are introduced in the RMSTMs problem.

The rest of this paper is organized as follows. Some useful notations that will be used throughout the paper are presented in Section 2. In Section 3, the Relative Margin Machines are presented. More specifically, a short overview of the maximum magrin SVMs, the $\mathbf{\Sigma}$-SVMs and the reasoning behind RMMs is provided (Subsections 3.1, 3.2 and 3.3, respectively). In Section 4 the extensions of SVMs, $\mathbf{\Sigma}$-SVMs and RMMs, to the novel STMs, $\mathbf{\Sigma}$-STMs and RMSTMs are described in detail (Subsections 4.1, 4.2 and 4.3, respectively). The power of the proposed classifiers is demonstrated in the gait and action recognition problems in Section 5. Finally, conclusions are drawn in Section 6.

2. USEFUL NOTATIONS IN MULTILINEAR ALGEBRA

An n-th order tensor is a collection of measurements indexed by n indices, each index corresponding to a mode. Thus, vectors are first-order and matrices are second-order tensors [13]. We will use lower case letters (e.g. x) to denote scalars, boldface lowercase letters (e.g. $\mathbf{x}$) and boldface capital letters (e.g. $\mathbf{X}$) to denote vector and matrices, respectively. Tensors of order 3 or higher will be denoted by boldface Euler script calligraphic letters (e.g. $\mathcal{X}$).

The i-th element of a vector $\mathbf{x} \in \mathbb{R}_+^I$ is denoted by $x_i, i = 1, 2, \ldots, I$. In a similar way, the elements of an n-th order tensor $\mathcal{X}$ will be denoted by $x_{i_1 i_2 \ldots i_n}$, $i_\ell = 1, 2, \ldots, I_\ell$, $\ell = 1, 2, \ldots, n$. To indicate the objects resulting by fixing one of the indices to a specific value, we introduce the generic subscript :. That is, the i-th row of a matrix $\mathbf{X}$ is denoted as $\mathbf{x}_{i:}$. Unless otherwise stated, the j-th column of a matrix $\mathbf{X}$ will be denoted compactly as $\mathbf{x}_j = \mathbf{x}_{:j}$.

The matricization (also unfolding or flattening of a tensor) is the reordering of the tensor elements into a matrix. The n-mode matricization of a tensor $\mathcal{A} \in \Re^{I_1 \times I_2 \times \cdots \times I_N}$, denoted as $\mathbf{A}_{(n)}$, arranges the n-mode fibers to become the columns of the final matrix. Each tensor element $(i_1, i_2, \ldots, i_N)$ maps to the matrix element $(i_n, j))$ as

$$j = 1 + \sum_{k=1, k\neq n}^{N} (i_k - 1) J_k, \text{with} \;\; J_k = \prod_{k=1, k\neq n}^{k-1} I_m. \quad (1)$$

In this paper, we will focus on n-th order tensors, i.e. $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_n}$. $\mathcal{X}$ can represent a database consisting of L samples. Every database sample is a tensor of order $(n-1)$ denoted as $\mathcal{X}_{:i_n} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_{n-1}}$, $i_n = 1, 2, \ldots, L$, that is indexed by $(n-1)$ indices $(i_1, i_2, \ldots, i_{n-1})$. For example, a database consisting of gait samples is a 4-th order tensor $\mathcal{X} \in \mathbb{R}_+^{I_1 \times I_2 \times I_3 \times I_4}$, where I_1 and I_2 refer to the image dimensions (height and width), respectively, I_3 corresponds to the number of images in every tensor sample and I_4 is the number of gait samples in the database.

Let $\mathbf{a} \in \mathbb{R}_+^I$ and $\mathbf{b} \in \mathbb{R}_+^J$ be two non-negative real valued vectors. Their outer product yields a matrix $\mathbf{C} \in \mathbb{R}_+^{I \times J}$

$$\mathbf{C} = \mathbf{a} \otimes \mathbf{b} \quad \text{with elements } c_{ij} = a_i\, b_j. \quad (2)$$

Consequently, the outer product of n vectors $\mathbf{a}_\ell \in \mathbb{R}_+^{I_\ell}$, $\ell = 1, 2, \ldots, n$, $\mathbf{a}_1 \otimes \mathbf{a}_2 \otimes \cdots \otimes \mathbf{a}_n = \bigotimes_{\ell=1}^n \mathbf{a}_\ell$ yields a tensor $\mathcal{A} \in \mathbb{R}_+^{I_1 \times I_2 \times \cdots \times I_n}$.

An important operation between a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_n}$ and a matrix $\mathbf{U} \in \mathbb{R}^{J \times I_\ell}$ is the ℓ-mode product denoted as $\mathcal{X} \times_\ell \mathbf{U}$ which yields a tensor $\mathcal{Y}$ of size $I_1 \times \cdots \times I_{\ell-1} \times J \times I_{\ell+1} \times \cdots I_n$ having elements [13]

$$y_{i_1 \ldots, i_{\ell-1} j i_{\ell+1} i_n} = \sum_{i_\ell=1}^{J} x_{i_1 i_2 \ldots i_n}\, u_{j\, i_\ell}, \;\; j = 1, 2, \ldots, J \quad (3)$$

with $i_m = 1, 2, \ldots, I_m$ and $m = 1, 2, \ldots, n$. The product $\mathcal{X} \times_1 \mathbf{U}_1 \times_2 \cdots \times_n \mathbf{U}_n$ will be denoted in compact notation as $\mathcal{X} \times_{k=1}^n \mathbf{U}_k$.

Let us also introduce the compact notation

$$\mathcal{X} \overline{\times}_j \mathbf{U}_l \triangleq \mathcal{X} \times_1 \mathbf{U}_1 \cdots \times_{j+1} \mathbf{U}_{j+1} \times_{j-1} \mathbf{U}_{j-1} \cdots \times_n \mathbf{U}^n. \quad (4)$$

In the remaining of the paper, $\odot$ and $/$ tensor, vector and matrix operators will denote the elementwise multiplication and division between tensors, vectors and matrices.

3. RELATIVE MARGIN SUPPORT VECTOR MACHINES

3.1 Support Vector Machines (SVMs)

Let a dataset $(\mathbf{x}_i, y_i)_{i=1}^n$, where $\mathbf{x}_i \in \Re^m$ with $y_i \in \{\pm 1\}$. The maximum margin Support Vector Machines (SVMs) formulation is given as:

$$\begin{array}{l} \min_{\mathbf{w}, b, \boldsymbol{\xi} \geq 0} \frac{1}{2} ||\mathbf{w}||^2 + C \boldsymbol{\xi}^T \mathbf{1} \\ \text{s.t. } y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i, \forall 1 \leq i \leq n. \end{array} \quad (5)$$

The above formulation aims at maximizing the margin of the Support Vectors while minimizing the upper bound on the misclassification errors.

3.2 Σ-Support Vector Machines (Σ-SVMs)

Below we will briefly present the $\mathbf{\Sigma}-SVMs$, as proposed in [14]. Lets consider the whitening of the data with the covariance (total scatter) matrix:

$$\mathbf{\Sigma} = \sum_{i=1}^n \mathbf{x}_i \mathbf{x}_i^T - \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n \mathbf{x}_i \mathbf{x}_j^T. \quad (6)$$

Let us also consider $\boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i$, the mean value of the data.

The formulation of $\mathbf{\Sigma}$-SVMs is given by:

$$\min_{\mathbf{w}, b, \boldsymbol{\xi} \geq 0} \quad \tfrac{1-D}{2} ||\mathbf{w}||^2 + \tfrac{D}{2} \left|\left| \mathbf{\Sigma}^{\frac{1}{2}} \mathbf{w} \right|\right|^2 + C \boldsymbol{\xi}^T \mathbf{1} \quad (7)$$

$$\text{s.t. } y_i \left(\mathbf{w}^T (\mathbf{x}_i - \boldsymbol{\mu}) + b \right) \geq 1 - \xi_i$$

where $0 \leq D \leq 1$ is the parameter that handles the two regularization terms, $||\mathbf{w}||^2$ and $||\mathbf{\Sigma}^{\frac{1}{2}} \mathbf{w}||^2$.

The dual problem of (7) is then given by:

$$\begin{array}{l} \max_{0 \leq \boldsymbol{\alpha} \leq C\mathbf{1}} \quad \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \alpha_i y_i (\mathbf{x}_i - \boldsymbol{\mu})^T ((1-D)\mathbf{I} + D\mathbf{\Sigma})^{-1} \\ \qquad \sum_{j=1}^n \alpha_j y_j (\mathbf{x}_j - \boldsymbol{\mu}) \\ \qquad \text{s.t. } y^T \alpha = 1. \end{array} \quad (8)$$

3.3 Relative Margin Machines (RMMs)

Relative Margin Machines (RMMs) were introduced to deal with a possible bad scaling of the data [14]. In order to achieve that, the bounding of the projections of the training data was used. The trade off is now between the projections and the margin, resulting in finding a large *relative* margin.

The Relative Margin Machines (RMMs) are given by the following formulation:

$$\begin{aligned}&\min_{\mathbf{w},b,\boldsymbol{\xi}\geq 0} \tfrac{1}{2}\,||\mathbf{w}||^2 + C\boldsymbol{\xi}^T\mathbf{1}\\ &\text{s.t. } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1-\xi_i,\\ &\tfrac{1}{2}(\mathbf{w}^T\mathbf{x}_i+b)^2 \leq \tfrac{B^2}{2}.\end{aligned} \tag{9}$$

As can be seen, the above formulation has one extra parameter in addition to the SVMs parameters, B (where $B \geq 1$). Let us denote as $\mathbf{w}_C$ and b_C the solutions obtained by solving the SVM (5) for a particular value of C. If $B > \max_i|\mathbf{w}_C^T\mathbf{x}_i + b_C|$, then the solution obtained is the same with the SVM estimate. If B is of a smaller value, the solutions obtained are different than that of the SVM estimate.

Let us assume that the value of B is smaller than the threshold. Then, the Lagrangian of (9) is the following:

$$\begin{aligned}&L_{RMMs}(\mathbf{w},\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\lambda},B) = \tfrac{1}{2}||\mathbf{w}||^2 + C\boldsymbol{\xi}\mathbf{1} -\\ &-\textstyle\sum_{i=1}^n \alpha_i\left(y_i(\mathbf{w}^T\mathbf{x}_i + b - 1 + \xi_i\right) - \boldsymbol{\beta}^T\boldsymbol{\xi} +\\ &+\textstyle\sum_{i=1}^n \lambda_i\left(\tfrac{1}{2}(\mathbf{w}^T\mathbf{x}_i+b)^2 - \tfrac{1}{2}B^2\right)\end{aligned} \tag{10}$$

where $\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\lambda} \geq 0$ are the Lagrangian multipliers that correspond to the constraints.

Differentiating with respect to the primal variables and equating them to zero, it can be shown that:

$$\begin{aligned}&(\mathbf{I}+\textstyle\sum_{i=1}^n \lambda_i\mathbf{x}_i\mathbf{x}_i^T)\mathbf{w} - b\sum_{i=1}^n\lambda_i\mathbf{x}_i = \sum_{i=1}^n\alpha_i y_i\mathbf{x}_i\\ &b = \tfrac{1}{\boldsymbol{\lambda}^T\mathbf{1}}(\textstyle\sum_{i=1}^n\alpha_i y_i - \sum_{i=1}^n\lambda_i\mathbf{w}^T\mathbf{x}_i)\\ &C\mathbf{1} = \boldsymbol{\alpha}+\boldsymbol{\beta}.\end{aligned} \tag{11}$$

Denoting by:

$$\boldsymbol{\Sigma}_{\boldsymbol{\lambda}} = \sum_{i=1}^n \lambda_i\mathbf{x}_i\mathbf{x}_i^T - \frac{1}{\boldsymbol{\lambda}^T\mathbf{1}}\sum_{i=1}^n\sum_{j=1}^n \lambda_i\mathbf{x}_i\lambda_j\mathbf{x}_j^T$$

and by:

$$\mu_{\boldsymbol{\lambda}} = \frac{1}{\boldsymbol{\lambda}^T\mathbf{1}}\sum_{j=1}^n\lambda_j\mathbf{x}_j,$$

the dual of (9) can be shown to be:

$$\begin{aligned}\max_{0\leq\boldsymbol{\alpha}\leq C\mathbf{1},\boldsymbol{\lambda}\geq 0}\quad &\textstyle\sum_{i=1}^n\alpha_i - \tfrac{1}{2}\sum_{i=1}^n\alpha_i y_i(\mathbf{x}_i-\boldsymbol{\mu}_{\boldsymbol{\lambda}})^T(\mathbf{I}+\boldsymbol{\Sigma}_{\boldsymbol{\lambda}})^{-1}\\ &\textstyle\sum_{j=1}^n\alpha_j y_j(\mathbf{x}_i-\boldsymbol{\mu}_{\boldsymbol{\lambda}}) - \tfrac{1}{2}B^2\boldsymbol{\lambda}^T\mathbf{1}\\ &\text{s.t. } y^T\alpha = 1.\end{aligned} \tag{12}$$

$\boldsymbol{\Sigma}_{\boldsymbol{\lambda}}$ corresponds to a "shape matrix" (potentially row rank) determined by the $\mathbf{x}_i$s that have nonzero λ_i. From the KKT conditions of (9) we have:

$$\lambda_i(\frac{1}{2}(\mathbf{x}_i\mathbf{w}^T+b)^2 - \frac{B^2}{2}) = 0. \tag{13}$$

Consequently $\lambda_i > 0$ implies that:

$$(\frac{1}{2}(\mathbf{x}_i\mathbf{w}_k^T+b)^2 - \frac{B^2}{2}) = 0. \tag{14}$$

Note that the constraint $\frac{1}{2}(\mathbf{x}_i\mathbf{w}_k^T+b)^2 \leq \frac{B^2}{2}$ can be equivalently posed as two linear constraints: $(\mathbf{x}_i\mathbf{w}_k^T+b)^2 \leq B$ and $-(\mathbf{x}_i\mathbf{w}_k^T+b)^2 \leq B$. Thus the problem to solve is a quadratic one.

4. RELATIVE MARGIN SUPPORT TENSOR MACHINES

The proposed method involves the extension of the RMMs proposed in [14] in order to deal with tensors. We will make a brief description of the typical Support Tensor Machines (STMs) [2] and continue with a detailed description of the novel $\boldsymbol{\Sigma}$-STMs and Relative Margin Support Tensor Machines (RMSTMs).

4.1 Support Tensor Machines (STMs)

Let a dataset be represented by the tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_n}$ where I_n is the number of samples in the dataset. The dataset is separated into two classes with I_n^A and I_n^B denoting the number of samples of each class. The label $y_i = 1$ is assigned to the samples belonging to the first class , while the label $y_i = -1$ is assigned to the samples belonging to the second class.

For the maximum margin STMs we aim at finding a multilinear decision function :

$$g(\mathcal{X}) = \text{sign}\left[\mathcal{X}\prod_{i=1}^{n-1}\times_i\mathbf{w}_i + b\right]. \tag{15}$$

The projection vectors $\mathbf{w}_j \in \mathbb{R}^{I_j}$ for every dimension $j = 1,\ldots,n$ and the bias term b are derived from solving the following soft STM problem:

$$\begin{aligned}&\min_{\mathbf{w}_j|_{j=1}^M,b,\boldsymbol{\xi}} \tfrac{1}{2}\left|\left|\textstyle\bigotimes_{k=1}^{n-1}\mathbf{w}_k\right|\right|^2 + C\textstyle\sum_{i=1}^{I_n}\xi_i\\ &\text{s.t. } y_i\left[\mathcal{X}_{:i}\textstyle\prod_{i=1}^{n-1}\times_k\mathbf{w}_k + b\right] \geq 1-\xi_i,\quad 1\leq i\leq I_n,\ \xi_i\geq 0.\end{aligned} \tag{16}$$

If we keep every term but $\mathbf{w}_j$ fixed, then the problem becomes convex and quadratic. Otherwise, the above form of the optimization problem is not convex with respect to all projection vectors $\mathbf{w}_k$ with $k = 1,\ldots,I_n$.

The j-th problem for solving with respect to $\mathbf{w}_j$ is given by:

$$\begin{aligned}&\min_{\mathbf{w}_j|_{j=1}^M,b,\boldsymbol{\xi}^j} \tfrac{\eta_j}{2}||\mathbf{w}_j||^2 + C\textstyle\sum_{i=1}^{I_n}\xi_i^j\\ &\text{s.t. } y_i\left[\mathbf{w}_j^T(\mathcal{X}_{:i}\times_j\mathbf{w}_k) + b\right] \geq 1-\xi_i^j,\quad 1\leq i\leq I_n\ \xi_i\geq 0\end{aligned} \tag{17}$$

where $\eta_j = \prod_{k=1,k\neq j}^n ||\mathbf{w}_k||^2$. The optimal vector $\mathbf{w}_j$ can be found by the saddle point of the Lagrangian:

$$\begin{aligned}&L_{STMs}^{(j)}(\mathbf{w}_j,b,\boldsymbol{\xi}^j) = \tfrac{\eta_j}{2}||\mathbf{w}_j||^2 + C\textstyle\sum_{i=1}^{I_n}\xi_i^j -\\ &-\textstyle\sum_{i=1}^{I_n}a_i^j\left(y_i\left[\mathbf{w}_j^T(\mathcal{X}_{:i}\overline{\times_j}\mathbf{w}_k) + b\right] - 1 + \xi_i^j\right) -\\ &-\textstyle\sum_{i=1}^{I_n}\kappa_i\xi_i.\end{aligned} \tag{18}$$

as

$$\begin{aligned}&\nabla_{\mathbf{w}_j}L_{STMs}^{(j)} = 0 \Rightarrow\\ \mathbf{w}_j\ &= \tfrac{1}{\eta_j}\textstyle\sum_{i=1}^{I_n}a_i^j y_i\mathcal{X}_{:i}\overline{\times_j}\mathbf{w}_k.\end{aligned} \tag{19}$$

The whole procedure is repeated iteratively for every mode, so as to find $\mathbf{w}_k,\ \ k = 1\ldots M$ [2].

4.2 Σ-STMs

In order to define the $\mathbf{\Sigma}$-STMs formulation, we follow the rationale behind $\mathbf{\Sigma}$-SVMs. More specifically, let us define the n-mode scatter matrices as:

$$\mathbf{\Sigma}_{(n)} = \sum_{k=1}^{m} (\mathbf{X}_{k(n)} - \overline{\mathbf{X}}_{(n)})(\mathbf{X}_{k(n)} - \overline{\mathbf{X}}_{(n)})^T \tag{20}$$

where $\mathbf{X}_{k(n)}$ is the n-mode unfolding of the tensor sample $\mathcal{X}_k$ and $\overline{\mathbf{X}}_{(n)}$ is the mean value of $\mathbf{X}_{k(n)}$.

Thus, the formulation of the $\mathbf{\Sigma}$-STMs is the following:

$$\min_{\mathbf{w}_j|_{j=1}^{M}, b, \boldsymbol{\xi} \geq 0} \quad \tfrac{1-D}{2} \left|\left|\bigotimes_{k=1}^{n-1} \mathbf{w}_k\right|\right|^2 + \tfrac{D}{2} \left|\left|\bigotimes_{k=1}^{n-1} \mathbf{\Sigma}_{(n)}{}^{\frac{1}{2}} \mathbf{w}_k\right|\right|^2 + $$
$$+ C \textstyle\sum_{i=1}^{I_n} \xi_i \tag{21}$$

$$\text{s.t. } y_i \left[(\mathcal{X}_{:i} - \overline{\mathcal{X}}) \prod_{i=1}^{n-1} \times_k \mathbf{w}_k + b \right] \geq 1 - \xi_i, \quad 1 \leq i \leq m \quad \xi_i \geq 0.$$

where $\overline{\mathcal{X}}_{:} = \frac{1}{I_n} \sum_{i=1}^{I_n} \mathcal{X}_{:i}$ is the mean tensor for the mode (n) and $0 \leq D \leq 1$ is the parameter that handles the two regularization terms $\left|\left|\bigotimes_{k=1}^{n-1} \mathbf{w}_k\right|\right|^2$ and $\left|\left|\bigotimes_{k=1}^{n-1} \mathbf{\Sigma}_{(n)}{}^{\frac{1}{2}} \mathbf{w}_k\right|\right|^2$.

For the j-th vector $\mathbf{w}_j$, the above optimization problem is reformulated as:

$$\min_{\mathbf{w}_j|_{j=1}^{M}, b, \boldsymbol{\xi} \geq 0} \quad \tfrac{1-D}{2} \eta_j \mathbf{w}_j^T \mathbf{w}_j + \tfrac{D}{2} \eta_j^{\Sigma} \mathbf{w}_j^T \mathbf{\Sigma}_{(j)} \mathbf{w}_j + $$
$$+ C \textstyle\sum_{i=1}^{I_n} \xi_i \tag{22}$$

$$\text{s.t. } y_i \left[\mathbf{w}_j^T ((\mathcal{X}_{:i} - \overline{\mathcal{X}}) \overline{\times}_j \mathbf{w}_r) + b \right] \geq 1 - \xi_i, \quad 1 \leq i \leq m \quad \xi_i \geq 0$$

where

$$\eta_j^{\Sigma} = \prod_{i=1, i \neq j}^{n-1} \mathbf{w}_i^T \mathbf{\Sigma}_{(i)} \mathbf{w}_i.$$

The Lagrangian is given by:

$$\begin{aligned} & L^{(j)}_{\mathbf{\Sigma}-STMs}(\mathbf{w}_j, b, \boldsymbol{\xi}^j) = \tfrac{1-D}{2} \eta_j \mathbf{w}_j^T \mathbf{w}_j + \\ & + \tfrac{D}{2} \eta_j^{\Sigma} \mathbf{w}_j^T \mathbf{\Sigma}_{(j)} \mathbf{w}_j + C \textstyle\sum_{i=1}^{I_n} \xi_i^j - \\ & - \textstyle\sum_{i=1}^{I_n} a_i^j \left(y_i \left[\mathbf{w}_j^T (((\mathcal{X}_{:i} - \overline{\mathcal{X}}) \overline{\times}_j \mathbf{w}_k) + b \right] - 1 + \xi_i^j \right) - \\ & - \textstyle\sum_{i=1}^{I_n} \kappa_i \xi_i. \end{aligned} \tag{23}$$

By letting $\tilde{\mathbf{\Sigma}}_{(j)} = ((1-D)\eta_j \mathbf{I}_j + D \eta_j^{\Sigma} \mathbf{\Sigma}_{(j)})$, we have:

$$\begin{aligned} \nabla_{\mathbf{w}_j} L_j &= 0 \Rightarrow \\ \tilde{\mathbf{\Sigma}}_{(j)} \mathbf{w}_j &= \textstyle\sum_{i=1}^{I_n} a_i^j y_i (\mathcal{X}_{:i} - \overline{\mathcal{X}}) \overline{\times_j} \mathbf{w}_k \Rightarrow \\ \mathbf{w}_j &= \tilde{\mathbf{\Sigma}}_{(j)}^{-1} \textstyle\sum_{i=1}^{I_n} a_i^j y_i (\mathcal{X}_{:i} - \overline{\mathcal{X}}) \overline{\times_j} \mathbf{w}_k. \end{aligned} \tag{24}$$

The whole procedure is repeated iteratively for every mode, so as to find $\mathbf{w}_k, \quad k = 1 \ldots M$.

The dual problem of (22) is thus defined as:

$$\max_{0 \leq \alpha_i^j \leq C} \quad -\tfrac{1}{2} \textstyle\sum_{i=1}^{m} \sum_{k=1}^{m} a_i^j a_k^j y_i y_k ((\mathcal{X}_{:i} - \overline{\mathcal{X}}) \overline{\times}_j \mathbf{w}_r)^T$$
$$\tilde{\mathbf{\Sigma}}_{(j)}^{-1} ((\mathcal{X}_{:k} - \overline{\mathcal{X}}) \overline{\times}_j \mathbf{w}_r) + \textstyle\sum_{i=1}^{m} \alpha_i^j \tag{25}$$
$$\text{s.t. } \ {\boldsymbol{\alpha}^j}^T \mathbf{y} = 0. \tag{26}$$

4.3 Relative Margin Support Tensor Machines

Following the same reasoning with the one used to define RMMs, we also use the projections of the training data to define Relative Margin Support Tensor Machines (RMSTMs). Their formulation is as follows:

$$\begin{aligned} & \min_{\mathbf{w}_j|_{j=1}^{M}, b, \boldsymbol{\xi} \geq 0} \tfrac{1}{2} \left|\left|\bigotimes_{k=1}^{n-1} \mathbf{w}_k\right|\right|^2 + C \textstyle\sum_{i=1}^{I_n} \xi_i \\ & \text{s.t. } y_i \left[\mathcal{X}_{:i} \prod_{i=1}^{n-1} \times_k \mathbf{w}_k + b \right] \geq 1 - \xi_i, \\ & \tfrac{1}{2} (\mathcal{X}_i \prod_{i=1}^{n-1} \times_k \mathbf{w}_k + b)^2 \leq \tfrac{B^2}{2}. \end{aligned} \tag{27}$$

The partial optimization problem in terms of $\mathbf{w}_j$ is given by:

$$\min_{\mathbf{w}_j, b, \boldsymbol{\xi}^j \geq 0} \quad \tfrac{1}{2} \eta_j \mathbf{w}_j^T \mathbf{w}_j + C \textstyle\sum_{i=1}^{I_n} \xi_i^j \tag{28}$$
$$\text{s.t. } y_i \left[\mathbf{w}_j^T (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r) + b \right] \geq 1 - \xi_i,$$
$$\tfrac{1}{2} (\mathbf{w}_j^T (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r) + b)^2 \leq \tfrac{B^2}{2}.$$

As can be seen, an extra parameter B (where $B \geq 1$) is introduced. If $\mathbf{w}_C$ and b_C the solutions acquired for the maximum margin STMs problem for a specific value C and
$B > \max_i |(\mathbf{w}_{j,C}^T \mathcal{X}_{:i} \overline{\times}_j \mathbf{w}_r) + b_C|$, then the solution is the same for both STMs and RMSTMs. If however B is of a smaller value, then the solution is different.

Let us assume that the value of B is smaller than the threshold. Then, the Lagrangian of (28) is given by:

$$\begin{aligned} & L_{RMSTMs}(\mathbf{w}_j, \mathbf{a}^j, \boldsymbol{\xi}^j, \boldsymbol{\beta}^j, \boldsymbol{\lambda}^j) = \tfrac{1}{2} \eta_j \mathbf{w}_j^T \mathbf{w}_j + \\ & + C \textstyle\sum_{i=1}^{I_n} \xi_i^j - \sum_{i=1}^{n} \alpha_i^j \left(y_i \mathbf{w}_j^T (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r) + b \right) - \\ & - 1 + \xi_i - {\boldsymbol{\beta}^j}^T \boldsymbol{\xi}^j + \\ & + \textstyle\sum_{i=1}^{n} \lambda_i \left(\tfrac{1}{2} (\mathbf{w}_j^T (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r) + b)^2 - \tfrac{1}{2} B^2 \right) \end{aligned} \tag{29}$$

where $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\lambda} \geq 0$ are the Lagrangian multipliers corresponding to the constraints. Differentiating with respect to the primal variables and equating them to zero, it can be shown that:

$$\begin{aligned} & (\eta_j \mathbf{I} + \textstyle\sum_{i=1}^{n} \lambda_i (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r)(\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r)^T) - b \sum_{i=1}^{n} \lambda_i \mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r = \\ & = \textstyle\sum_{i=1}^{n} \alpha_i y_i \mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r) \\ & b = \tfrac{1}{\boldsymbol{\lambda}^T \mathbf{1}} (\textstyle\sum_{i=1}^{n} \alpha_i y_i - \sum_{i=1}^{n} \lambda_i \mathbf{w}_j^T (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r)) \\ & C\mathbf{1} = \boldsymbol{\alpha} + \boldsymbol{\beta} \end{aligned} \tag{30}$$

Denoting by:

$$\begin{aligned} \mathbf{\Sigma}_{\boldsymbol{\lambda}} \quad & = \textstyle\sum_{i=1}^{n} \lambda_i \mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r^T) \\ & - \tfrac{1}{\boldsymbol{\lambda}^T \mathbf{1}} \textstyle\sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j (\mathcal{X}_{:i} \times_j \mathbf{w}_r)(\mathcal{X}_{:i} \times_j \mathbf{w}_r)^T \end{aligned}$$

and by $\boldsymbol{\mu}_{\boldsymbol{\lambda}} = \frac{1}{\boldsymbol{\lambda}^T \mathbf{1}} \sum_{j=1}^{n} \lambda_j \mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r$, the dual of (29) can be shown to be:

$$\max_{0 \leq \boldsymbol{\alpha} \leq C, \boldsymbol{\lambda} \geq 0} \quad \textstyle\sum_{i=1}^{n} \alpha_i - \tfrac{1}{2} \sum_{i=1}^{n} \alpha_i y_i (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r - \boldsymbol{\mu}_{\boldsymbol{\lambda}})^T (\mathbf{I} + \mathbf{\Sigma}_{\boldsymbol{\lambda}})^{-1}$$
$$\textstyle\sum_{j=1}^{n} \alpha_j y_j (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r - \boldsymbol{\mu}_{\boldsymbol{\lambda}}) - \tfrac{1}{2} B^2 \boldsymbol{\lambda}^T \mathbf{1}. \tag{31}$$
$$\text{s.t. } y_i \left[\mathbf{w}_j^T (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r) + b \right] \geq 1 - \xi_i,$$
$$\tfrac{1}{2} (\mathbf{w}_j^T (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r) + b)^2 \leq \tfrac{B^2}{2}.$$

From the KKT conditions of (29) we have:

$$\lambda_i \left(\frac{1}{2} (\mathcal{X}_i \times_k \mathbf{w}^T + b)^2 - \frac{B^2}{2} \right) = 0. \tag{32}$$

Consequently $\lambda_i > 0$ implies that

$$(\frac{1}{2}(\mathcal{X}_i \times_k \mathbf{w}_k^T + b)^2 - \frac{B^2}{2}) = 0. \quad (33)$$

It can be seen that the constraint $\frac{1}{2}(\mathcal{X}_i \times_k \mathbf{w}_k^T + b)^2 \leq \frac{B^2}{2}$ can be reformulated as two linear constraints: $(\mathcal{X}_i \times_k \mathbf{w}_k^T + b)^2 \leq B$ and $-(\mathcal{X}_i \times_k \mathbf{w}_k^T + b)^2 \leq B$, thus making the problem to solve, a quadratic one.

4.4 RMSTMs with Linear Constraints

In order to implement a fast algorithm, the quadratic constraints bounding the projections should be replaced with linear ones. The equivalent of (27) with linear constraints is given by:

$$\min_{\mathbf{w}_j|_{j=1}^M, b, \boldsymbol{\xi} \geq 0} \quad \frac{1}{2} \left|\left|\bigotimes_{k=1}^{n-1} \mathbf{w}_k\right|\right|^2 + C \sum_{i=1}^{I_n} \xi_i \quad (34)$$

$$\text{s.t. } y_i \left[\mathcal{X}_{:i} \prod_{i=1}^{n-1} \times_k \mathbf{w}_k + b\right] \geq 1 - \xi_i,$$
$$(\mathcal{X}_i \prod_{i=1}^{n-1} \times_k \mathbf{w}_k + b) \leq B$$
$$-(\mathcal{X}_i \prod_{i=1}^{n-1} \times_k \mathbf{w}_k + b) \leq B$$

Then, the Lagrangian of (34) is given by:

$$L(\mathbf{w}_j, \mathbf{a}^j, \boldsymbol{\xi}^j, \boldsymbol{\beta}^j, \boldsymbol{\lambda}^j, \boldsymbol{\lambda}^{*j}) = \frac{1}{2}\eta_j \mathbf{w}_j^T \mathbf{w}_j + C \sum_{i=1}^{I_n} \xi_i^j$$
$$- \sum_{i=1}^n \alpha_i^j \left(y_i \mathbf{w}_j^T (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r) + b - 1 + \xi_i\right) -$$
$$-{\boldsymbol{\beta}^j}^T \boldsymbol{\xi}^j + \sum_{i=1}^n {\lambda_i}^j \left((\mathbf{w}_j^T (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r) + b) - B\right) +$$
$$+ \sum_{i=1}^n \lambda_i^{j*} \left(-(\mathbf{w}_j^T (\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r) + b) - B\right) \quad (35)$$

as:

$$\nabla_{\mathbf{w}_j} L = 0 \Rightarrow$$
$$\eta_j \mathbf{w}_j = \sum_{i=1}^n (a_i^j y_i - {\lambda_i}^j + {\lambda_i^j}^*)(\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_k). \quad (36)$$

The Wold dual problem of (35) is the following:

$$\max_{\mathbf{a}^j, \boldsymbol{\lambda}^j, \boldsymbol{\lambda}^{*j}} \quad (\mathbf{a}^j \odot \mathbf{y} - \boldsymbol{\lambda}^j + \boldsymbol{\lambda}^{*j})^T \mathbf{G}^j (\mathbf{a}^j - \boldsymbol{\lambda}^j + \boldsymbol{\lambda}^{*j}) +$$
$$+ {\mathbf{a}^j}^T \mathbf{1} - B {\boldsymbol{\lambda}^j}^T \mathbf{1} - B {\boldsymbol{\lambda}^{*j}}^T \mathbf{1} \quad (37)$$
$$\text{s.t. } \quad {\mathbf{a}^j}^T \mathbf{1} - {\boldsymbol{\lambda}^j}^T \mathbf{1} + {\boldsymbol{\lambda}^{*j}}^T \mathbf{1} = 0$$

where $\mathbf{G}^j = [\frac{1}{\eta_j}(\mathcal{X}_{:i} \overline{\times_j} \mathbf{w}_r)^T (\mathcal{X}_{:k} \overline{\times_j} \mathbf{w}_r)]$.

The above problem can be solved in an iterative way, optimizing in each step a subset of the dual variables.

5. EXPERIMENTAL RESULTS

In this Section, we will present the acquired experimental results in order to justify the superiority of RMSTMs over STMs. The gait and actions recognition problems will be studied. In our experiments, the leave-one-out cross-validation approach was used to test the generalization performance of the classifiers. The experiments were performed on an Intel Core 2 Quad PC (2,66 GHz) processor with 4GB RAM memory.

5.1 Gait recognition experiments

The database used for the gait recognition experiments was the USF HumanID Gait Challenge data sets version 1.7, as used in [1], for comparison reasons.

The database includes 452 sequences from 74 subjects (persons) walking in elliptical paths in front of the camera. Three variations are provided for each subject: viewpoint (left/right), shoe type (two different types) and surface type (grass/concrete). Seven experiments (referred to as probe sets) are available, containing each one 71 sequences from each subject. The probe sets are of increasing complexity/ difficulty with probe set A being the easiest one and probe set G being the most difficult. There are no common sequences between the gallery sets and any of the probe sets and each probe set is unique. The largest dimension of the gait samples contained in the database, i.e tensors of dimension $128 \times 88 \times 40$ were used for the experiments. An example of a gait sequence can be seen in Figure 1.

The acquired results for all probe sets when STMs and RMSTMs are shown in the last two columns of Tables 1 and 2 (rank 1 and 5 identification rates, respectively). The highest accuracy rate for every probe set is emphasized in bold. In the same Tables, a comparison of the recognition rates achieved for each probe set with the state of the art [1], [4], [5], [6], [7], is also provided.

As can be seen, the recognition rates of the proposed method are the highest for probe sets C to G, for rank 1 results. The acquired mean accuracy rate of RMSTMs is also higher than that of all other mentioned methods. For rank 5 results, the recognition rates (and the mean accuracy rates) of STMs and the proposed RMSTMs are the highest for all probe sets. It is also worth mentioning here that in state of the art methods, the recognition accuracy drops drastically as the difficulty of the probe set increases, while in STMs and the proposed RMSTMs the accuracy rates remain in satisfactory levels. This is due to the use of tensors for the description of the features used for classification.

More specifically, one can see that when all the frames are considered as a tensor from which features are to be extracted, instead of extracting features seperately in a vector format and combining them afterwards, the spatial and temporal correlation of the initial information is preserved. For example, the geometry of spatial points is taken under consideration, something that does not happen when vectors are used. Even in the simplest case when an image is considered as a 2 mode tensor instead of a vector (as we will show below in the action recognition experiments 5.2), the results can be very promising.

5.2 Action recognition experiments

For the action recognition experiments, the Weizmann database was used [12]. It contains nine activities (bend, jack, jump, pjump, run, side, skip, walk, wave1 and wave2) performed by nine subjects. An example of 5 frames per action period, for each action to be recognized, is shown in the first column of Figure 2.

In order to perform action recognition, spatio-temporal salient points were extracted using the method presented in [15]. More specifically, the points extracted are salient both in space and time and are detected by measuring the variations in the information context of pixel neighborhoods. They correspond to activity-variation peaks and are used to achieve invariance against the translation of the subjects performing the actions. From the set of points extracted for every sequence, we create a single image by projecting them on a single frame (i.e. by ignoring their temporal location). That single image contains all the spatiotemporal points detected in the sequence, that appear in grayscale format. An example of the image we create for every action can be seen in the middle column of Figure 2. For visualization purposes, the grayscale points were all painted white so as to be visible to the reader. A set of dilations and

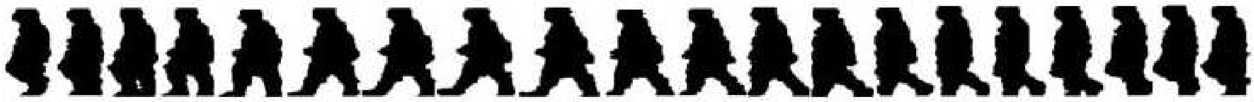

Figure 1: An example of a gait sequence.

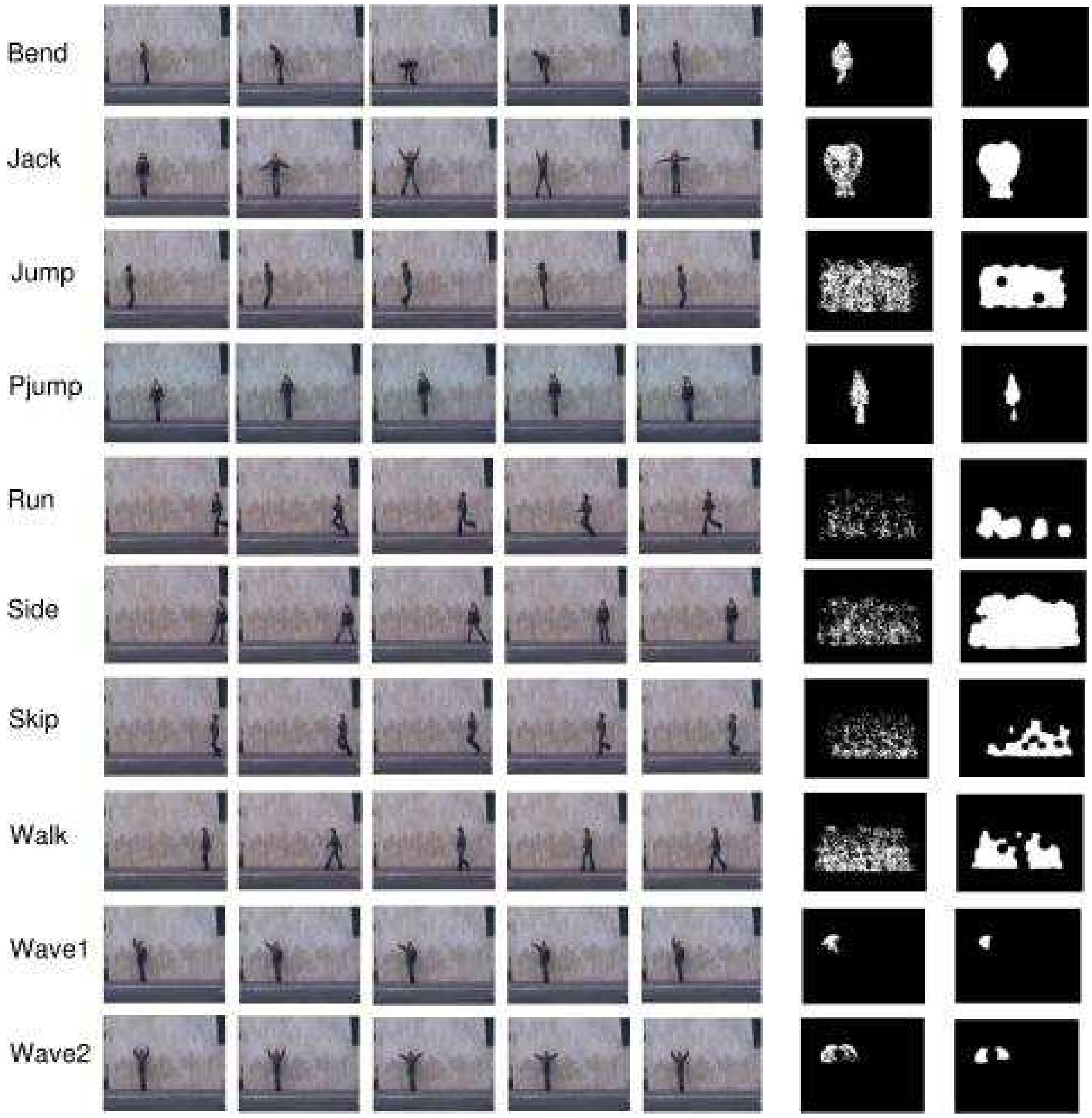

Figure 2: An example of the action sequences (first column), spatiotemporal points detected (second column) and binary masks extracted for action classification (third column).

Table 1: Comparison of proposed method with state of the art (rank 1).

Probe Set	Baseline[5]	HMM[7]	LTN[4]	GEI[6]	ETG[1]	ETGLDA[1]	STMs	MVSTMs
A	79	99	94	**100**	92	99	78	81
B	66	**89**	83	85	85	88	75	77
C	56	78	78	80	76	83	83	**86**
D	29	35	33	30	39	36	78	**80**
E	24	29	24	33	29	29	81	**83**
F	30	18	17	21	21	21	78	**80**
G	10	24	21	29	21	21	82	**85**
Average	42	53	50	54	52	54	80	**82**

Table 2: Comparison of proposed method with state of the art (rank 5).

Probe Set	Baseline[5]	HMM[7]	LTN[4]	GEI[6]	ETG[1]	ETGLDA[1]	STMs	MVSTMs
A	96	**100**	99	**100**	96	**100**	**100**	**100**
B	81	90	85	85	90	93	**100**	**100**
C	76	90	83	88	81	88	**100**	**100**
D	61	65	65	55	55	71	**100**	**100**
E	55	65	67	55	52	60	**100**	**100**
F	46	60	58	41	58	59	**100**	**100**
G	33	50	48	48	50	60	**100**	**100**
Average	64	74	72	67	69	76	**100**	**100**

erosions is then applied in order to create the binary mask used for classification. An example of the binary mask for each one of the actions is provided in the last column of Figure 2.

The recognition accuracies achieved when STMs and RMSTMs were used, were equal to 84.46% and 87.76%, respectively. Thus, the use of RMSTMs introduces an increase of 3.3%. In order to better study the problem, the confusion matrices have been computed. The confusion matrix is a $n \times n$ matrix containing information about the actual class label Action$_{ac}$ (in its columns) and the label obtained through classification Action$_{cl}$ (in its rows). The diagonal entries of the confusion matrix are the percentages that correspond to the cases when actions are correctly classified, while the off-diagonal entries correspond to misclassifications. The confusion matrix when STMs were used is given in Table 3, while the equivalent results when RMSTMs were used are provided in Table 4.

As can be seen from the confusion matrices, the use of RMSTMs improves the recognition accuracy results in the cases when most misclassifications were observed. More specifically, the misclassification of jump as bend and that one of pjump as wave2 are now omitted. Thus, even the use of even an image as a 2-mode can provide satisfactory results when tensors are used.

6. CONCLUSIONS

In this paper the novel Relative Margin Support Tensor Machines are proposed as an extension of the Relative Margin Support Vector Machines. The proposed RMSTMs exploit the spread of the data in order to find the final solution. The efficiency of the proposed method has been studied for the problems of gait and action recognition, where the results achieved illustrated the superiority of the method in terms of classification performance.

7. ACKNOWLEDGEMENT

This work was supported by the EPSRC grant 'Recognition and Localization of Human Actions in Image Sequences' (EP/G033935/1).

8. REFERENCES

[1] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos. Mpca: Multilinear principal component analysis of tensor objects. *IEEE Transactions on Neural Networks*, 19(1):18–39, 2008.

[2] D. Tao, X. Li, X. Wu, W. Hu, and S. J. Maybank. Supervised tensor learning. *Knowledge and Information Systems*, 13(1):1–42, 2007.

[3] T.-K. Kim and R. Cipolla. Canonical Correlation Analysis of Video Volume Tensors for Action Categorization and Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(8):1415–1428, August 2009.

[4] N. V. Boulgouris, K. N. Plataniotis, and D. Hatzinakos. Gait recognition using linear time normalization. *Pattern Recognition*, 39(5):969–979, February 2006.

[5] S. Sarkar, P. J. Phillips, Z. Liu, I. Robledo, P. Grother, and K. W. Bowyer. The human ID gait challenge problem: Data sets, performance, and analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(2):162–177, September 2005.

[6] J. Han and B. Bhanu. Individual recognition using gait energy image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(2):316–322, February 2006.

[7] A. Kale, A. N. Rajagopalan, A. Sunderesan, N. Cuntoor, A. Roy-Chowdhury, V. Krueger, and R. Chellappa. Identification of humans using gait. *IEEE Transactions on Image Processing*, 1163–1173, September 2004.

[8] S. Belongie, J. Malik, and J. Puzicha. Shape Matching and Object Recognition Using Shape Contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, April 2002.

[9] D. Ramanan, D. A. Forsyth,A. Zisserman. Tracking People by Learning Their Appearance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1):65–81, January 2007.

[10] A. Oikonomopoulos, M. Pantic. Human body gesture recognition using adapted auxiliary particle filtering. *Advanced Signal and Video Based Surveillance.*, 441–446,

Table 3: Confusion matrix of STMs.

$\text{Action}_{ac}\%\backslash\text{Action}_{cl}\%$	Bend	Jack	Jump	Pjump	Run	Side	Skip	Walk	Wave1	Wave2
Bend	100.0	0	11.1	0	0	0	0	0	0	0
Jack	0	100.0	0	0	0	0	11.1	0	0	0
Jump	0	0	33.4	11.1	0	0	11.1	11.1	0	0
Pjump	0	0	0	66.7	0	0	0	0	0	0
Run	0	0	22.2	0	100.0	0	11.1	0	0	0
Side	0	0	0	0	0	100.0	0	0	0	0
Skip	0	0	0	11.1	0	0	66.7	0	0	0
Walk	0	0	33.3	0	0	0	0	88.9	0	0
Wave1	0	0	0	0	0	0	0	0	88.9	0
Wave2	0	0	0	11.1	0	0	0	0	11.1	100.0

Table 4: Confusion matrix of RMSTMs.

$\text{Action}_{ac}\%\backslash\text{Action}_{cl}\%$	Bend	Jack	Jump	Pjump	Run	Side	Skip	Walk	Wave1	Wave2
Bend	100.0	0	0	0	0	0	0	0	0	0
Jack	0	100.0	0	0	0	0	11.1	0	0	0
Jump	0	0	44.5	11.1	0	0	11.1	11.1	0	0
Pjump	0	0	0	77.8	0	0	0	0	0	0
Run	0	0	22.2	0	100.0	0	11.1	0	0	0
Side	0	0	0	0	0	100.0	0	0	0	0
Skip	0	0	0	11.1	0	0	66.7	0	0	0
Walk	0	0	33.3	0	0	0	0	88.9	0	0
Wave1	0	0	0	0	0	0	0	0	88.9	0
Wave2	0	0	0	0	0	0	0	0	11.1	100.0

2007.

[11] B. Stenger, A. Thayananthan, P. H.S. Torr, and R. Cipolla. Model-Based Hand Tracking Using a Hierarchical Bayesian Filter. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9), September 2006.

[12] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *Transactions on Pattern Analysis and Machine Intelligence*, 29(12):2247–2253, December 2007.

[13] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[14] P. Shivaswamy and T. Jebara. Relative Margin Machines. *Neural Information Processing Systems*, 2008.

[15] A. Oikonomopoulos, I. Patras, and M. Pantic. Spatiotemporal salient points for visual recognition of human actions. *IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics*, 36(3):710–719, June 2006.

[16] D. Cunado, M. Nixon, and J. Carter. Automatic extraction and descrip- tion of human gait models for recognition purposes. *Computer Vision and Image Understanding*, 90:1–41, April 2003.

[17] L. Wang, T. Tan, W. Hu, and H. Ning. Silhouette analysis-based gait recognition for human identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1505–1518, December 2003.