Computing and Informatics, Vol. 35, 2016, 1338–1355

# A NEW SPARSE REPRESENTATION ALGORITHM FOR 3D HUMAN POSE ESTIMATION

Azam Andalib, Seyed Morteza Babamir, Alireza Faraji

*Department of Computer Engineering*
*University of Kashan, Kashan, Iran*
*e-mail:* azam.andalib@grad.kashanu.ac.ir
{babamir, arfaraji}@kashanu.ac.ir

**Abstract.** This paper addresses the problem of recovering 3D human pose from single 2D images using Sparse Representation. While recent Sparse Representation (SR) based 3D human pose estimation methods have attained promising results estimating human poses from single images, their performance depends on the availability of large labeled datasets. However, in many real world applications, accessing to sufficient labeled data may be expensive and/or time consuming, but it is relatively easy to acquire a large amount of unlabeled data. Moreover, all SR based 3D pose estimation methods only consider the information of the input feature space and they cannot utilize the information of the pose space. In this paper, we propose a new framework based on sparse representation for 3D human pose estimation which uses both the labeled and unlabeled data. Furthermore, the proposed method can exploit the information of the pose space to improve the pose estimation accuracy. Experimental results show that the performance of the proposed method is significantly better than the state of the art 3D human pose estimation methods.

**Keywords:** Sparse representation, dictionary learning, 3D human pose estimation, local linear embedding

## 1 INTRODUCTION

Estimation of 3D human poses from single images has greatly affected many computer vision applications, including visual surveillance, activity recognition, gesture recognition, motion capture, video indexing and retrieval, human-computer interaction, to name a few [10, 5, 7, 31].

3D human pose estimation aims to infer a human pose, represented by joint positions or angles, from input images or videos [12, 9, 14, 23]. Previous methods commonly tackled this problem as a regression or a manifold learning task in which features are embedded to a parametrised 3D pose space.

Despite the merits of the existing pose estimation methods, this problem has remained as a very dificult and still unsolved problem for various reasons.

First, recovering 3D human poses directly from 2D images is inherently ambiguous due to loss of depth information.

Second, the visual appearance and the shape of the humans change greatly in images because of factors such as viewpoints, lighting conditions, clothing, and poses.

Because of these challenges, there was a considerable previous work done on this problem [5] that we will discuss in the next section.

## 2 RELATED WORK

In general, the approaches in this area can be divided into two classes: *generative approaches* (model based) and *discriminative approaches* (learning based).

Generative approaches employ a known parametric body model based on prior knowledge and estimate the human pose by inverting the kinematics or by solving an optimization problem [10, 7, 15]. These approaches estimate body configuration by searching high dimensional spaces (body configuration and geometric transformation) which is typically formulated deterministically as a nonlinear optimization problem, e.g. [10], or probabilistically as a maximum likelihood problem, e.g. [7].

Although the generative methods are flexible representing large classes of poses and useful for training and hypothesis verification, these computationally expensive methods need good initialization and proper models. Furthermore, the methods of this category can only find sub-optimal solutions due to the fact that their objective functions are not usually convex.

Discriminative approaches [8, 12, 14, 19, 21, 16, 24] have attempted to directly learn a mapping from 2D images to 3D poses which is in contrast with generative methods that search the pose space for configurations with good image alignment.

The mapping is often approximated using regression models [12]. Such approaches have great potentials in solving the fundamental initialization problem for generative approaches.

Although such methods are faster and more flexible than the generative methods, one major weakness of these methods is that the capability of inferring poses with good precision depends on the amount of training data. Unfortunately, the acquisition of a sufficiently representative training set may not be possible. More precisely, the construction of labeled human pose datasets (images of humans and their 3D poses) is inherently difficult due to the fact that no existing system can provide accurate 3D ground truth for humans in real-world, non-instrumented scenes.

Recently, some researchers have utilized the Sparse Representation (SR) framework for estimating the human poses from monocular images [22, 20, 25, 30, 26].

Huang et al. [22] proposed a SR based method which is capable of dealing with occlusion in which each test data point is expressed as a compact linear combination of the training visual inputs, and the pose of the test sample can be recovered by the same linear combination of the training poses.

Ji et al. [25] introduced a robust dual dictionaries learning (DDL) approach which can handle corrupted input images. An efficient algorithm is also provided to solve DDL optimization model.

Zolfaghari et al. [30] proposed a coupled sparse dictionary learning method which learns the sparse representation of a new input using both the feature and pose space information and then estimates the corresponding 3D pose by a linear combination of the bases of a learned dictionary using the input data.

Despite the merits of SR based algorithms, these methods have two big disadvantages:

1. None of these methods can use the information of the pose training data. Precisely speaking, all of the SR based methods learn the dictionary and the sparse codes without considering the fact that the dictionary should be learned so that the samples which have similar poses, should have similar sparse codes.
2. The performance of these methods is highly dependent on the number of the labeled training data points. Unfortunately, in many pose estimation problems, accessibility to a large set of labeled data may not be possible due to the fact that labeling data is expensive and time consuming. On the other hand, unlabeled data points are easily available in abundance what motivated us to develop a semi-supervised learning method which utilizes a large amount of unlabeled data, along with a limited number of labeled data, to build better models for pose estimation tasks.

To address the aforementioned problems, this paper presents a semi-supervised sparse representation based 3D human pose estimation which takes into account the local structure of the pose data points. Precisely speaking, using Local Fisher Discriminant Analysis (LFDA) algorithm [18], we utilize the locality structure of the samples in the output (pose) space, by which the dictionary is learned such that the sparse codes of the input data points which their poses are near each other in the output space, have similar sparse codes. Moreover, we exploit the Local Linear Embedding (LLE) algorithm [1] to preserve the global structure of both the labeled and unlabeled data.

The remainder of this paper is organized as follows: In Sections 3, 4, and 5 we briefly describe the sparse representation framework, LFDA and LLE algorithms. In Section 6, we introduce the proposed method in detail. We describe the experimental results in Section 7. Finally, the conclusion and the future work are discussed in Section 8.

## 3 SPARSE REPRESENTATION

In recent years, sparse representation has attracted much attention to itself from the signal processing community [3, 11, 27, 28]. This attention is due to the fact that many natural signals are sparse in their nature and can be fairly approximated by their sparse codes.

This reconstruction is obtained by a linear combination of some atoms (bases). We refer to the collection of these atoms as a **dictionary**. A dictionary is defined as a set of vectors capable of providing a highly succinct representation for a set of representative signal vectors. We denote a dictionary by $D = [d_1, d_2, \dots, d_K] \in \mathbb{R}^{M \times K}$, where $d_i$ denotes the $i^{\text{th}}$ atom with dimension $M$.

Suppose that we want to find some atoms of the dictionary $D$, such that the reconstructed signal $\hat{X}$ is as close as possible to the original signal $X$, and the combination coefficients $\alpha$ are as sparse as possible. The formulation of this problem can be expressed as

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \|\alpha\|_0 \quad \text{s.t. } \|\hat{X} - X\|_2^2 \leq T \tag{1}$$

where $\hat{X} = D\alpha$, $T$ denotes a predefined threshold, and $\|.\|_0$ denotes $l_0$ norm which is defined as

$$\|x\|_0 = \#\{j \quad \text{s.t. } x_j \neq 0\} = \lim_{q \to 0} \left( \sum_{j=1}^{M} |x_j|^q \right). \tag{2}$$

Unfortunately, due to the NP-hard nature of the above problem, $\hat{\alpha}$ cannot be computed efficiently, but it has been shown that $l_1$ norm also creates sparse solutions [2]. Thus, the Equation (1) can be reformulated as

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \|X - D\alpha\|_2^2 + \lambda \|\alpha\|_1. \tag{3}$$

In Equation (3), $\lambda$ is a regularization parameter which controls the tradeoff between sparseness and reconstruction error. Using $l_1$ norm rather than $l_0$ norm has two advantages.

First, if the solution of the problem at hand is sparse enough, $l_1$ norm has the same result as $l_0$ norm. Second, a $l_0$ norm problem has a high time complexity and cannot be solved in a reasonable amount of time. On the other hand, $l_1$ norm of this representation converts the non-convex problem to a convex one. If the dictionary is not a pre-defined one, we have to find both the sparse codes and a proper dictionary, thus the general form of the optimization problem changes to

$$[\hat{\alpha}, \hat{D}] = \underset{\alpha, D}{\operatorname{argmin}} \|X - D\alpha\|_2^2 + \lambda \|\alpha\|_1. \tag{4}$$

Unfortunately, the objective function of Equation (4) is not convex respect to $\alpha$ and $D$ simultaneously. However, by fixing one parameter, $\alpha$ or $D$, and minimizing the other one, the problem can be treated as a convex problem. Methods such as K-means Singular Value Decomposition (K-SVD) [4] and the Method of Optimized Directions (MOD) [11] can be used to solve the above problem efficiently.

In these methods, the sparse codes and the dictionary are updated in two phases. One phase is when the sparse codes are being updated explicitly when the dictionary is fixed. This task can be done by using methods such as greedy orthogonal matching pursuit [3] if we want to use $l_0$ norm as measure of sparseness, or by basis pursuit [2] if we want to use $l_1$ norm.

Another phase is when the atoms of the dictionary are getting updated. In MOD, the sparse codes are fixed and all atoms of dictionary are updated using least square criteria, whereas in K-SVD, not only the atoms of the dictionary are updated, but also the nonzero coefficients of the sparse code are updated in the same time.

## 4 LOCAL FISHER DISCRIMINANT ANALYSIS

Local Fisher Discriminant Analysis (LFDA) algorithm was originally proposed for supervised dimension reduction problem that effectively combines the ideas of Fisher Discriminant Analysis (FDA) and locality preserving projection (LPP) [18]. In LFDA method, the local between-class scatter matrix $S^{lB}$ and the local within-class scatter matrix $S^{lW}$ are defined as: [18]

$$S^{(LB)} = \frac{1}{2}\sum_{i,j=1}^{N} W_{i,j}^{(lb)}(x_i - x_j)(x_i - x_j)^T,$$
$$S^{(LW)} = \frac{1}{2}\sum_{i,j=1}^{N} W_{i,j}^{(lw)}(x_i - x_j)(x_i - x_j)^T$$

where $\{x_i\}_{i=1}^N$ are input samples, and $W_{i,j}^{(lb)}$, $W_{i,j}^{(lw)}$ are the $N \times N$ matrices which are defined based on the class label of the input signals [18]. Then, the LFDA method for supervised dimension reduction can be formulated as

$$\hat{T} = \underset{T}{\operatorname{argmax}}\, tr\left(T^T S^{(LB)} T (T^T S^{(LW)} T)^{-1}\right) \tag{5}$$

where $T$ is the transformation matrix such that nearby data pairs in the same class are made close and the data pairs in different classes are separated from each other.

## 5 LOCAL LINEAR EMBEDDING

Local Linear Embedding [1] is one of the algorithms that tries to preserve the topological structure of the data by retaining locally linear relationships between close

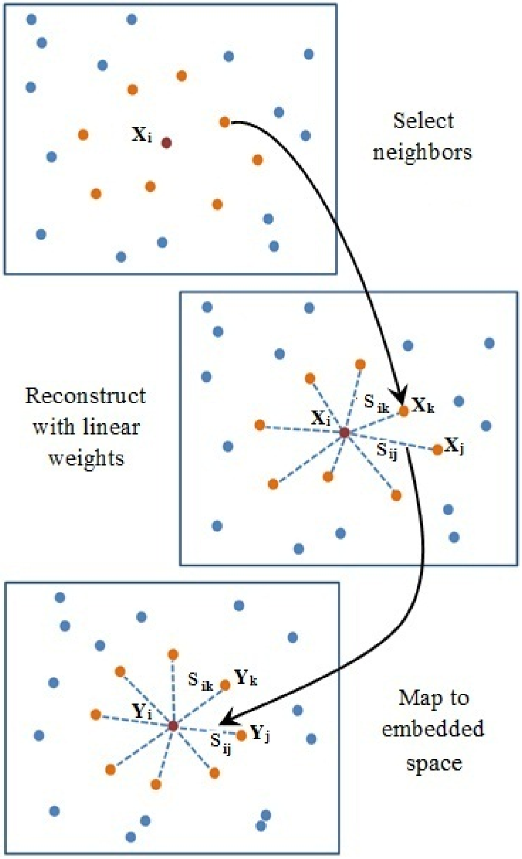


Figure 1. The cartoon representation of LLE model

data points in the transformed space (the transformed space in this paper is the sparse feature space).

This method assumes that the data is linear in each neighborhood, which means that any data point $p$ can be approximated by a weighted average of its neighbors.

Given a set of data points, this method constructs a $k$-nearest neighbor graph which models the relations between nearby data points. The algorithm finds the weights that minimize the cost of representing a point by its neighbors under the $l_2$-norm. The optimal weight matrix $S^* = [s^*_{ij}]$ providing minimal error for the linear reconstruction of data points from their neighbors is obtained according to (Figure 1 shows the cartoon representation of this model):

$$S^* = \underset{S=[s_{ij}]}{\operatorname{argmin}} \sum_{i=1}^{n} \|x_i - \sum_{x_j \in N_k(x_i)} s_{ij} x_j\|^2,$$
$$\text{s.t. } \forall i, \sum_{x_j \in N_k(x_i)} s_{ij} = 1, \tag{6}$$

where $N_k(x_i)$ shows the set of $k$ nearest neighbors of $x_i$. This problem can be solved as a constrained least-squares problem [1].

## 6 PROPOSED METHOD

In this section, we present our semi-supervised framework for 3D human pose estimation which takes into account both the labeled and unlabeled data. Here, the goal is to learn the dictionary so that the sparse codes of the similar poses are similar, and to preserve the geometrical structure of all data.

### 6.1 Problem Formulation

Let $X_L = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N_l}] \in \mathbb{R}^{M_x \times N_l}$, and $Y_L = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{N_l}] \in \mathbb{R}^{M_y \times N_l}$ be the labeled training set of $N_l$ visual input features and their corresponding pose features, respectively, and $X_U = [\boldsymbol{x}_{N_l+1}, \ldots, \boldsymbol{x}_N] \in \mathbb{R}^{M_x \times N_u}$ be the set of unlabeled data available for learning the dictionary, where $N_u = N - N_l$ and $N$ are the number of unlabeled and total samples, respectively.

Let $D = [d_1, \ldots, d_K] \in \mathbb{R}^{M_x \times K}$ be the input dictionary with $K$ atoms, and $A = [A_L, A_U]_{K \times N}$ be the matrix of the sparse codes, where $A_L = [\alpha_1, \ldots, \alpha_l]_{K \times N_l}$ and $A_U = [\alpha_{N_l+1}, \ldots, \alpha_N]_{K \times N_u}$ show the matrices of the sparse codes of the labeled and unlabeled input features, respectively.

Now, we propose the following optimization problem for learning the dictionary ($D$) and the sparse codes ($A$) based on both the labeled and the unlabeled data.

$$\left[\hat{A}, \hat{D}\right] = \underset{D,A}{\operatorname{argmin}} \|X_L - DA_L\|_2^2 + \|X_U - DA_U\|_2^2$$
$$+ \lambda_1 \sum_{i=1}^{N} \|\alpha_i\|_1 + \lambda_2 F_1(A_L) + \lambda_3 F_2(A_L, A_U) \tag{7}$$

where, $\|.\|_2^2$ denotes the reconstruction error term, $\|.\|_1$ denotes the sparsity constraint, $\lambda_1$, $\lambda_2$, $\lambda_3$ denote the regularization parameters, $F_1(A_L)$ denotes the discriminative term, and $F_2(A_L, A_U)$ denotes the topological structure preserving term. In the following, we discuss the design of $F_1$ and $F_2$ based on LFDA and LLE algorithms.

### 6.2 Discriminative Term $F_1(A_L)$

In order to enhance the discriminativeness (by discriminativeness we mean that the sparse codes of the similar poses should be similar) of the dictionary based on the information of the pose features, we modify the LFDA algorithm by modifying the construction of the matrices $W_{i,j}^{(lb)}$, $W_{i,j}^{(lw)}$ as

$$W_{i,j}^{(lb)} = \begin{cases} P_{i,j}(1/N_l - 1/k) & \text{if } y_j \in \mathcal{N}_k(y_i), \\ 1/N_l & \text{othervise}, \end{cases} \tag{8}$$

$$W_{i,j}^{(lw)} = \begin{cases} P_{i,j}/k & \text{if } y_j \in \mathcal{N}_k(y_i), \\ 0 & \text{otherwise}, \end{cases} \tag{9}$$

where $\mathcal{N}_k(y_i)$ denotes the set of posed features which belong to the $k$ nearest neighbors of the pose feature $y_i$ (a heuristic choice of $k = 5$ was shown to be useful through experiments), and $P_{i,j}$ shows the affinity value between input samples $x_i$ and $x_j$ which is defined as

$$P_{i,j} = \exp\left(-\frac{\|y_i - y_j\|^2}{\gamma_i \gamma_j}\right) \tag{10}$$

where the parameter $\gamma_i$ represents the local scaling around $x_i$ as

$$\gamma_i = \|x_i - x_i^k\|, \tag{11}$$

and $x_i^k$ is the $k^{\text{th}}$ nearest neighbor of $x_i$.

Now, we define the discriminative term as

$$F_1(A_L) := tr(A_L L^{LW} A_L^T) - tr(A_L L^{LB} A_L^T) \tag{12}$$

where $tr$ denotes the trace operator, and $L^{LW}$ and $L^{LB}$ denote the graph Laplacian matrices which are defined as

$$L^{LW} = D^{LW} - W^{(lw)}, \quad L^{LB} = D^{LB} - W^{(lb)} \tag{13}$$

where $D^{LW}$ and $D^{LB}$ are diagonal $N_l \times N_l$ matrices with

$$D_{i,i}^{LW} = \sum_{j=1}^{N_l} W_{i,j}^{(lw)}, \quad D_{i,i}^{LB} = \sum_{j=1}^{N_l} W_{i,j}^{(lb)}. \tag{14}$$

The problem with the proposed discriminative term $F_1(A_L)$ is that it is not convex. In order to address this issue, we use the idea of [13], and add an elastic term $\|A_L\|_F^2$ into $F_1(A_L)$, where $\|.\|_F$ denotes the Frobenius norm (for more information about why $\|A_L\|_F^2$ can be used for convexification refere to [13]). So, $F_1(A_L)$ is defined as

$$F_1(A_L) := tr(A_L L^{LW} A_L^T) - tr(A_L L^{LB} A_L^T) + \|A_L\|_F^2. \tag{15}$$

### 6.3 Topological Structure Preserving Term $F_2(A_L, A_U)$

In order to preserve the intrinsic topological structure of both labeled and unlabeled data, we use the LLE algorithm which tries to preserve the geometrical structure of the data based on the notion of affinity preserving. More precisely, using LLE, the topological structure of the data is retained by maintaining locally linear relationships between sparse codes of close data points. Hence, after finding the optimal weight matrix $S^*$, we define the topological structure preserving term $F_2(A_L, A_U)$ as

$$\begin{aligned} F_2(A_L, A_U) &:= \sum_{i=1}^{N} \|\alpha_i - \sum_{x_j \in N_k(x_i)} s^*_{ij}\alpha_j\|^2 \\ &= tr\left(AEA^T\right) \end{aligned} \tag{16}$$

where

$$E = (I - S^*)^T (I - S^*) \tag{17}$$

and $I$ denotes the Identity matrix.

### 6.4 Proposed Model

By plugging Equations (16) and (15) into Equation (7), the proposed objective function can be expressed as

$$\begin{aligned} \left[\hat{A}, \hat{D}\right] = \underset{D,A}{\operatorname{argmin}} & \|X_L - DA_L\|_2^2 + \|X_U - DA_U\|_2^2 \\ & + \lambda_1 \sum_{i=1}^{N} \|\alpha_i\|_1 + \lambda_3 tr\left(AEA^T\right) + \lambda_2 \|A_L\|_F^2 \\ & + \lambda_2 \left(tr(A_L L^{LW} A_L^T) - tr(A_L L^{LB} A_L^T)\right), \\ & \text{s.t. } \forall k = 1, \ldots, K, \|d_k\|_2 \leq 1, \end{aligned} \tag{18}$$

where $d_k$ denotes the $k^{\text{th}}$ atom of the dictionary. Since the values of the atoms of $D$ can be arbitrary large, the values of the sparse codes can be relatively low. Therefore, it is common to normalize the atoms of the dictionary such that each column has $\|d_k\|_2 \leq 1$.

### 6.5 Optimization Procedure

In this section, we describe the optimization procedure for the proposed objective function (Equation (18)). Solving Equation (18) is a challenging task due to the fact that it is not jointly convex to $(D, A)$. However, it is convex with respect to each of $D$ and $A$ when the other is fixed. Hence, we resort to a coordinate descent method [17], in which unknown parameters are updated through an iterative process which updates each parameter by fixing the other parameter in each step.

### 6.5.1 Updating Dictionary $D$ with Fixed $A$:

Given $A$, the optimization problem for $D$ can be formulated as

$$\begin{aligned}\hat{D} &= \underset{D}{\operatorname{argmin}} \|X_L - DA_L\|_2^2 + \|X_U - DA_U\|_2^2, \\ &\text{s.t. } \forall k = 1, \ldots, K, \|d_k\|_2 \leq 1. \end{aligned} \tag{19}$$

The above problem is an constrained quadratic programming (QP), for which $D$ can be computed efficiently using QP solvers.

### 6.5.2 Updating the Sparse Codes $A$ with Fixed $D$:

We optimize each sparse code $\alpha_i (i = 1, \ldots, N)$ by fixing sparse codes $\alpha_j (j \neq i)$ of other signals. Hence, for each sparse code $\alpha_i$, if $x_i \in X_L$, we must solve

$$\begin{aligned}\hat{\alpha_i} = \underset{\alpha_i}{\operatorname{argmin}} &\|x_i - D\alpha_i\|_2^2 + \lambda_1\|\alpha_i\|_1 + \lambda_2\|\alpha_i\|_2^2 \\ &+ 2\lambda_2\alpha_i^T \left(A_L(L_i^{LW} - L_i^{LB})\right) - \lambda_2\alpha_i^T\alpha_i \left(L_{i,i}^{LW} - L_{i,i}^{LB}\right) \\ &+ 2\lambda_3\alpha_i^T (AE_i) - \lambda_3\alpha_i^T\alpha_i (E_{i,i}), \end{aligned} \tag{20}$$

and if $x_i \in X_U$, we must solve

$$\begin{aligned}\hat{\alpha_i} = \underset{\alpha_i}{\operatorname{argmin}} &\|x_i - D\alpha_i\|_2^2 + \lambda_1\|\alpha_i\|_1 + 2\lambda_3\alpha_i^T (AE_i) \\ &- \alpha_i^T\alpha_i (E_{i,i}) \end{aligned} \tag{21}$$

where $T_i$ is the $i^{\text{th}}$ column of the matrix $T$ and $T_{i,i}$ is the $(i, i)$ element of $T$.

In this paper, we adopt the feature-sign search algorithm [6] to solve (20) and (21). This algorithm proceeds in a series of feature-sign steps: in each step, it first estimate the coefficient sign $\theta_i$ of the sparse code $\alpha_i$ (for more information about how this algorithm estimate $\theta_i$ for $\alpha_i$, refer to [6]), then, it computes $\alpha_i$ by replacing $\|\alpha_i\|_1$ by $\theta_i\alpha_i$ and setting the derivative of the objective function of Equations (20) and (21) with respect to $\alpha_i$ equal to zero. Hence, the analytic solution of $\alpha_i$ can be obtained as

$$\begin{aligned}\hat{\alpha_i} = (2D^TD + 2\lambda_2 I - 2\lambda_2 &\sum_{j\neq i} \left(L_{ji}^{LW} - L_{ji}^{LB}\right)\alpha_j \\ &- 2\lambda_3 \sum_{j\neq i} E_{ji}\alpha_j)^{-1} \left(2D^Tx_i - \lambda_1\theta_i\right), \end{aligned} \tag{22}$$

**Algorithm 1** Semi-Supervised Sparse Representation

```
Require: $X_L, X_U, Y_L, K, k, \lambda_1, \lambda_2, \lambda_3$.
    Output: $D, A_L, A_U$.
    Initilization step: $A_L^{(0)} = 0; A_U^{(0)} = 0; D^{(0)} = 0;$
for $t = 0, 1, 2, \dots$ until convergence do
    for all $i \in V$ do
        Compute $D^{(t+1)}$ via Equation (19) using a QP solver.
        for $i = 1$ to $N$ do
            if $x_i \in X_L$
             compute $\alpha_i^{(t+1)}$ via Equation (22) using feature search sign algorithm.
            else
             compute $\alpha_i^{(t+1)}$ via Equation (23) using feature search sign algorithm.
            end if
        end for
    end for
end for
```

if $x_i \in X_L$, and

$$\hat{\alpha}_i = \left(2D^T D - 2\lambda_3 \sum_{j \neq i} E_{ji} \alpha_j\right)^{-1} \left(2D^T x_i - \lambda_1 \theta_i\right), \tag{23}$$

if $x_i \in X_U$. The overall algorithm is summarized in Algorithm 1.

## 6.6 Pose Prediction

After computing the dictionary $D$ and the sparse codes $A$ by solving the optimization problem of the Equation (18), in order to determine the target pose $y_t$ of a given test instance $x_t$, we first compute the sparse code $\alpha_t$ of that test instance as

$$\hat{\alpha}_t = \underset{\alpha_t}{\operatorname{argmin}} \|x_t - D\alpha_t\|_2^2 + \lambda \|\alpha_t\|_1 \tag{24}$$

where $\lambda$ denotes the regularization parameter. Then, the target pose $y_t$ is estimated as

$$\hat{y}_t = \frac{\sum_{l=1}^{N_l} \gamma_l \boldsymbol{y}_l}{\gamma}, \tag{25}$$

where

$$\gamma_l = \|\alpha_t - \alpha_l\|_2^2, \quad l = 1, \dots, N_l, \quad \gamma = \sum_{l=1}^{N_l} \gamma_l. \tag{26}$$

## 7 EXPERIMENTAL RESULTS

In this section, we explain the experimental results on realistic sequences of human activities. We tested the performance of the proposed method with different types of activities using motion captured poses from CMU Mocap data base[1].

The CMU Mocap database consists of different activities such as "Acrobatic", "Navigate", "Throw and catch football", "Golf", "Laugh", "Boxing", "Cartweel", "Michael Jackson styled motions", "Swim", "Run", "Kick soccer ball", "Traffic", and "Walk".

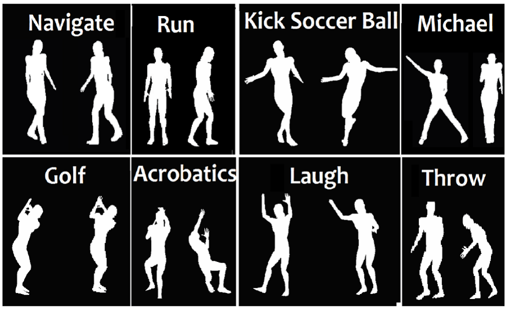


Figure 2. Sample visual inputs of the Mocap dataset

In our simulations, we use activities in bvh format to generate silhouettes for the realistic sequences. We also use the histograms of shape contexts [9] which encodes the visual input (silhouette) into a 100 dimensional descriptor as our input feature. Figure 2 shows some sample visual inputs of the Mocap dataset.

The human body pose is also encoded by 57 joint angles (three angles for each joint). To evaluate the 3D body configuration estimation, we used mean (over all pose dimensions) RMS error to compare the true and estimated body configuration, in degrees [9]:

$$D(\boldsymbol{y}, \boldsymbol{y}') = \frac{1}{57} \sum_{i=1}^{57} |(y_i - y_i') \bmod \pm 180^\circ|, \tag{27}$$

where $\boldsymbol{y} = [y_1, \ldots, y_{57}]$ and $\boldsymbol{y}' = [y_1', \ldots, y_{57}']$ are the true pose and the estimated pose, respectively.

We capture 1 000 frames from each sequence and use 500, 600, and 700 of them as training data (we use 100, 200, 300 of the training data as labeled data and the rest as unlabeled data), and the rest as the test data.

In order to determine an appropriate number of dictionary atoms $K$ and nearest neighbors of the data samples $k$ for LLE algorithm, the five-fold cross validation approach is performed to find the best pair $(K, k)$. The tested values for $K$ are

[1] `http://mocap.cs.cmu.edu/`

$\{64; 128; 196; 256\}$ and for $k$ are $\{3; 5; 7\}$. Since determining the values of the regularization parameters $\lambda, \lambda_1, \lambda_2$ and $\lambda_3$ using the cross validation technique is time consuming, we set them manually for all experiments as

$$\lambda = 0.005; \quad \lambda_1 = 0.005; \quad \lambda_2 = 0.005; \quad \lambda_3 = 0.05. \tag{28}$$

For comparison purposes, we compare the performance of our method with that of the relevance vector machine (RVM) [12] as a well known supervised regression method, the twin Gaussian process (TGP) [16] as a state-of-the-art method, and DDL [25] and SR [20] as two state-of-the-art sparse representation based 3D human pose estimation methods.

The average estimation accuracies (over 10 runs) together with the standard deviation for various activities are shown in Table 1 (in that table, "L-Tr. #" denotes the number of labeled data, and "U-Tr. #" denotes the number of unlabeled data), from which we can see that for all activities, the proposed method outperforms the other methods. This is due to the fact that the proposed method exploits the information of the pose space (the pose training data) as well as the information of the unlabeled data, but other methods ignore the information of the unlabeled data.

### 7.1 Sensitivity Analysis

To further analyse the impact of the unlabeled data on the pose estimation accuracy, we evaluate the effect of the unlabeled data regularizer term $\lambda_3$ (incorporating the geometrical structure) in the proposed method.

More precisely, we want to determine how much the information of the unlabeled training data could improve the pose estimation accuracy. To do so, in this section, we develop an experiment, in which we evaluate whether the information of unlabeled data for pose estimation is useful or not.

As it was mentioned before, the regularization parameter $\lambda_3$ controls how much the information of the unlabeled data is incorporated into the proposed model. Hence, in Figure 3, we show the average estimation accuracy of the proposed method considering the geometrical structure (the information available in the unlabeled data), i.e. $\lambda_3 = 0.05$, and ignoring it, i.e. $\lambda_3 = 0$, for some activities. According to this figure, using the geometrical structure (via the regularizer term $\lambda_3$) significantly improves results of the proposed method.

## 8 DISCUSSION AND FUTURE WORK

In this paper, we proposed a semi-supervised method for sparse representation based 3D human pose estimation that can utilize the information of both labeled and unlabeled data.

Incorporating the unlabeled data into the proposed model is done using the Local Linear Embedding algorithm. Furthermore, the proposed method used the

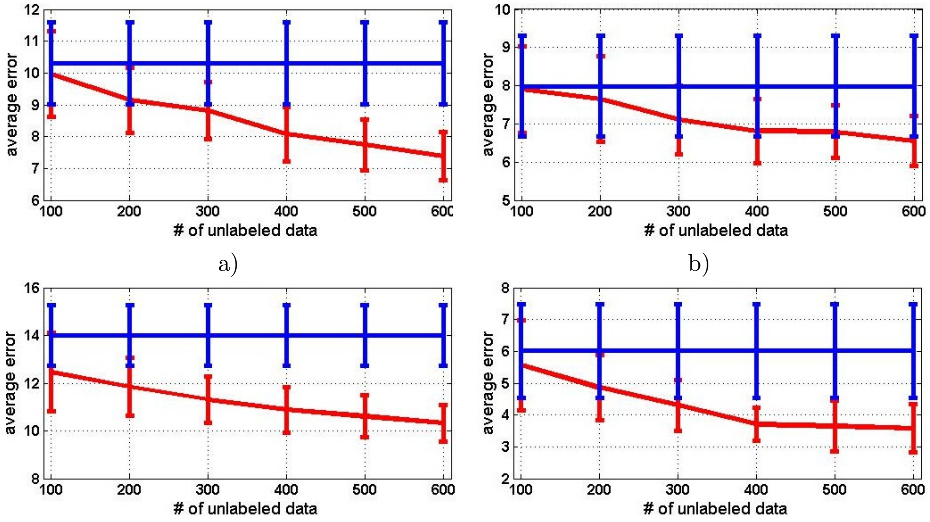
| Activity | L-Tr. # | U-Tr. # | RVM | TGP | SR | DDL | PM |
|---|---|---|---|---|---|---|---|
| Run | 100 | 400 | 9.3 ± 1.1 | 9.9 ± 0.7 | 9.0 ± 0.6 | 10.0 ± 0.6 | **7.4** ± 0.4 |
| | 200 | 400 | 6.6 ± 0.9 | 7.8 ± 0.4 | 6.8 ± 0.4 | 7.2 ± 0.6 | **5.6** ± 0.3 |
| | 300 | 400 | 5.7 ± 0.2 | 5.9 ± 0.3 | 5.8 ± 0.3 | 6.0 ± 0.3 | **4.3** ± 0.3 |
| Walk | 100 | 400 | 10.7 ± 1.6 | 10.0 ± 1.3 | 10.6 ± 0.9 | 11.1 ± 1.4 | **8.0** ± 0.6 |
| | 200 | 400 | 7.6 ± 0.7 | 6.6 ± 0.9 | 6.7 ± 0.6 | 7.1 ± 0.8 | **5.0** ± 0.5 |
| | 300 | 400 | 5.8 ± 0.7 | 5.7 ± 0.8 | 5.5 ± 0.5 | 5.9 ± 0.4 | **4.1** ± 0.4 |
| Throw. | 100 | 400 | 17.2 ± 2.9 | 11.9 ± 1.6 | 10.1 ± 0.8 | 11.4 ± 1.6 | **8.4** ± 0.9 |
| | 200 | 400 | 10.1 ± 1.5 | 8.9 ± 0.9 | 7.2 ± 0.4 | 8.8 ± 1.3 | **6.8** ± 0.4 |
| | 300 | 400 | 7.2 ± 0.7 | 6.8 ± 0.4 | 6.0 ± 0.4 | 6.7 ± 0.7 | **5.1** ± 0.1 |
| Michael. | 100 | 400 | 12.9 ± 0.8 | 12.7 ± 0.9 | 11.5 ± 1.3 | 12.9 ± 1.5 | **10.2** ± 0.8 |
| | 200 | 400 | 8.9 ± 0.6 | 10.8 ± 0.4 | 8.4 ± 0.5 | 8.9 ± 1.3 | **6.7** ± 0.3 |
| | 300 | 400 | 7.7 ± 0.4 | 8.4 ± 0.3 | 7.5 ± 0.4 | 7.7 ± 0.6 | **5.2** ± 0.1 |
| Kick. | 100 | 400 | 9.0 ± 0.9 | 10.8 ± 1.2 | 8.7 ± 1.5 | 9.7 ± 1.1 | **7.1** ± 0.7 |
| | 200 | 400 | 6.7 ± 0.6 | 7.9 ± 0.8 | 6.4 ± 0.8 | 7.2 ± 0.9 | **5.1** ± 0.3 |
| | 300 | 400 | 5.5 ± 0.4 | 5.8 ± 0.5 | 5.4 ± 0.3 | 6.1 ± 0.5 | **4.0** ± 0.2 |
| Traffic | 100 | 400 | 7.8 ± 0.7 | 8.5 ± 1.4 | 9.2 ± 1.1 | 9.7 ± 1.4 | **6.8** ± 0.8 |
| | 200 | 400 | 4.5 ± 0.6 | 4.3 ± 0.6 | 4.4 ± 0.4 | 4.6 ± 0.7 | **2.8** ± 0.3 |
| | 300 | 400 | 4.4 ± 0.3 | 3.8 ± 0.4 | 3.5 ± 0.2 | 3.8 ± 0.5 | **2.5** ± 0.1 |
| Swim | 100 | 400 | 14.4 ± 1.7 | 14.0 ± 1.5 | 13.7 ± 1.6 | 14.5 ± 1.9 | **10.8** ± 0.9 |
| | 200 | 400 | 12.6 ± 1.9 | 12.1 ± 0.9 | 11.4 ± 1.1 | 12.4 ± 1.4 | **9.6** ± 0.7 |
| | 300 | 400 | 11.0 ± 0.9 | 10.5 ± 0.5 | 10.3 ± 0.6 | 10.8 ± 0.8 | **8.8** ± 0.6 |
| Acro. | 100 | 400 | 10.9 ± 1.7 | 9.9 ± 1.9 | 8.4 ± 1.5 | 10.7 ± 2.2 | **6.2** ± 0.9 |
| | 200 | 400 | 7.6 ± 1.5 | 7.0 ± 1.2 | 5.7 ± 1.0 | 7.1 ± 1.6 | **4.3** ± 0.7 |
| | 300 | 400 | 6.1 ± 1.3 | 6.1 ± 0.9 | 4.9 ± 0.9 | 6.2 ± 1.3 | **3.7** ± 0.5 |
| Navi. | 100 | 400 | 4.9 ± 0.4 | 5.6 ± 0.6 | 5.7 ± 0.6 | 6.0 ± 0.9 | **3.0** ± 0.4 |
| | 200 | 400 | 3.9 ± 0.3 | 4.4 ± 0.5 | 3.9 ± 0.4 | 4.7 ± 0.8 | **2.8** ± 0.3 |
| | 300 | 400 | 3.5 ± 0.2 | 3.8 ± 0.3 | 3.5 ± 0.2 | 3.8 ± 0.6 | **2.5** ± 0.1 |
| Golf | 100 | 400 | 7.9 ± 1.5 | 7.8 ± 0.9 | 5.8 ± 0.7 | 7.9 ± 0.9 | **3.7** ± 0.5 |
| | 200 | 400 | 5.6 ± 1.6 | 5.6 ± 0.8 | 4.6 ± 0.6 | 5.6 ± 0.8 | **2.8** ± 0.3 |
| | 300 | 400 | 4.5 ± 1.4 | 4.7 ± 0.4 | 3.9 ± 0.4 | 4.5 ± 0.6 | **2.5** ± 0.2 |
| Laugh | 100 | 400 | 7.5 ± 0.8 | 9.9 ± 1.3 | 8.7 ± 1.2 | 9.2 ± 1.7 | **6.3** ± 0.6 |
| | 200 | 400 | 5.8 ± 0.7 | 6.5 ± 0.8 | 5.6 ± 0.8 | 5.9 ± 0.9 | **4.8** ± 0.3 |
| | 300 | 400 | 4.9 ± 0.3 | 5.8 ± 0.6 | 4.5 ± 0.5 | 4.6 ± 0.7 | **3.9** ± 0.2 |
| Boxing | 100 | 400 | 11.2 ± 1.7 | 10.9 ± 1.6 | 10.4 ± 1.2 | 11.3 ± 1.3 | **8.8** ± 1.0 |
| | 200 | 400 | 8.5 ± 1.1 | 7.8 ± 0.9 | 7.4 ± 0.8 | 8.8 ± 0.8 | **6.0** ± 0.5 |
| | 300 | 400 | 8.0 ± 0.9 | 7.1 ± 0.8 | 6.9 ± 0.7 | 7.3 ± 0.5 | **5.2** ± 0.3 |

Table 1. Average error (in degrees) with standard deviation for different methods

information of the pose data to enhance the pose estimation accuracy. This is done by incorporating the modified Local Fisher Discriminant Analysis into the our model. Empirical results on several activities of the benchmark Mocap dataset showed the superiority of the proposed model over other state of the art 3D human pose estimation methods.

Although the empirical results have shown the superiority of our method over other state of the art methods, there is one important issue with the proposed model that we are going to address in the future work:

In order to compute the LLE and LFDA, we need to determine the nearest neighbors of the input features. For doing so, we use the well-known Euclidean distance as the distance measure. However, it was shown that in 3D pose estimation domain, the high dimensional input and output data points lie on a non-linear low dimensional manifold [19]. Hence, using the Euclidean distance may not be realistic in this situation. A better idea may be to use the Geodesic distance [29] instead of the Euclidean distance.



a)

b)

c)

d)

Figure 3. Average error (with standard deviation) on some activities of the Mocap dataset for both cases of considering the geometrical structure (red curve) and ignoring it (blue curve). a) "Walk" b) "Traffic" c) "Swim" d) "Golf"

## REFERENCES

[1] Roweis, S. T.—Saul, L. K.: Nonlinear Dimensionality Reduction by Locally Linear Embedding. Science, Vol. 290, 2000, No. 5500, pp. 2323–2326.

[2] Chen, S. S.—Donoho, D. L.—Saunders, M. A.: Atomic Decomposition by Basis Pursuit. SIAM Journal on Scientific Computing, Vol. 20, 1998, No. 1, pp. 33–61.

[3] Mallat S.—Zhang, Z.: Matching Pursuits with Time-Frequency Dictionaries. IEEE Transactions on Signal Processing, Vol. 41, 1993, No. 12, pp. 3397–3415.

[4] Aharon, M.—Elad, M.—Bruckstein, A.: k-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. IEEE Transactions on Signal Processing, Vol. 54, 2006, No. 11, pp. 4311–4322.

[5] Gavrila, D. M.: The Visual Analysis of Human Movement: A Survey. Journal of Computer Vision and Image Understanding (CVIU), Vol. 73, 1999, No. 1, pp. 82–98.

[6] Lee, H.—Battle, A.—Raina, R.—Ng, A. Y.: Efficient Sparse Coding Algorithms. Proceedings of the 19th International Conference on Neural Information Processing Systems (NIPS '06), 2006, pp. 801–808.

[7] Sidenbladh, H.—Black, M. J.—Fleet.—D. J.: Stochastic Tracking of 3S Human Figures Using 2d Image Motion. Proceedings of the 6th European Conference on Computer Vision (ECCV '00), 2000, pp. 702–718.

[8] Shakhnarovich, G.—Viola, P.—Darrell, T.: Pose Estimation with Parameter Sensitive Hashing. Proceedings of the Ninth IEEE Conference on Computer Vision (ICCV '03), 2003, pp. 432–440.

[9] Agarwal, A.—Triggs, B.: Recovering 3D Human Pose from Monocular Images. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 28, 2006, No. 1, pp. 44–58.

[10] Rehg, J. M.—Kanade, T.: Model-Based Tracking of Selfoccluding Articulated Objects. Proceedings of 5th International Conference on Computer Vision (ICCV), 1995, pp. 612–617.

[11] Kreutz-Delgado, K.—Murray, J.—Rao, B.—Engan, K.—Lee, T.-W.—Sejnowski, T.: Dictionary Learning Algorithms for Sparse Representation. Neural Computation, Vol. 15, 2003, No. 2, pp. 349–396.

[12] Agarwal, A.—Triggs, B.: 3D Human Pose from Silhouettes by Relevance Vector Regression. Proceedings of the 2004 IEEE International Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04), 2004, pp. 882–888.

[13] Yang, M.—Zhang, L.—Feng, X.—Zhang, D.: Fisher Discrimination Dictionary Learning for Sparse Representation. Proceedings of the 2011 IEEE Conference on Computer Vision (ICCV '11), 2011, pp. 543–550.

[14] Mori, G.—Malik, J.: Recovering 3D Human Body Configurations Using Shape Contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 28, 2006, No. 7, pp. 1052–1062.

[15] Andriluka, M.—Roth, S.—Schiele, B.: Monocular 3D Pose Estimation and Tracking by Detection. 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 623–630.

[16] Bo, L.—Sminchisescu, C.: Twin Gaussian Processes for Structured Prediction. International Journal of Computer Vision (IJCV), Vol. 87, 2010, No. 1-2, pp. 28–52.

[17] Chang, K.-W.—Hsieh, C.-J.—Lin, C.-J.: Coordinate Descent Method for Large-Scale l2-Loss Linear Support Vector Machines. Journal of Machine Learning Research, Vol. 9, 2008, pp. 1369–1398.

[18] Sugiyama, M.—Ide, T.—Nakajima, S.—Sese, J.: Semi-Supervised Local Fisher Discriminant Analysis for Dimensionality Reduction. Machine Learning, Vol. 78, 2010, No. 1-2, pp. 35–61.

[19] Lee, M.—Nevatia, R.: Human Pose Tracking in Monocular Sequence Using Multilevel Structured Models. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 31, 2009, No. 1, pp. 27–38.

[20] Huang, J.-B.—Yang, M.-H.: Fast Sparse Representation with Prototypes. 2010 IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 3618–3625.

[21] Salzmann, M.—Urtasun, R.: Implicitly Constrained Gaussian Process Regression for Monocular Non-Rigid Pose Estimation. Advances in Neural Information Processing Systems (NIPS), Vol. 23, 2010, pp. 2065–2073.

[22] Huang, J. B.—Yang, M. H.: Estimating Human Pose from Occluded Images. Proceedings of the 9th Asian Conference on Computer Vision (ACCV '09), Part I, 2009, pp. 48–60.

[23] Jaume-i-Capó, A.—Varona, J.—González-Hidalgo, M.—Perales, F.: Adding Image Constraints to Inverse Kinematics for Human Motion Capture. EURASIP Journal on Advances in Signal Processing, Vol. 2010, 2010, Article No. 142354.

[24] Hara, K.—Kurokawa, T.: Human Pose Estimation Using Patch-Based Candidate Generation and Model-Based Verification. IEEE International Conference on Automatic Face Gesture Recognition and Workshops (FG 2011), 2011, pp. 687–693.

[25] Ji, H.—Su, F.: Robust 3D Human Pose Estimation Via Dual Dictionaries Learning. 2012 21st International Conference on Pattern Recognition (ICPR), 2012, pp. 58–61.

[26] Babagholami-Mohamadabadi, B.—Jourabloo, A.—Zarghami, A.—Kasaei, S.: A Bayesian Framework for Sparse Representation-Based 3D Human Pose Estimation. IEEE Signal Processing Letters, Vol. 21, 2014, No. 3, pp. 297–300.

[27] Babagholami-Mohamadabadi, B.—Jourabloo, A.—Zarghami, A.—Soleymani Baghshah, M.: Supervised Dictionary Learning Using Distance Dependent Indian Buffet Process. 2013 IEEE International Workshop on Machine Learning for Signal Processing (MLSP), 2013, pp. 1–6.

[28] Babagholami-Mohamadabadi, B.—Zarghami, A.—Zolfaghari, M.—Soleymani Baghshah, M.: PSSDL: Probabilistic Semi-supervised Dictionary Learning. In: Blockeel, H., Kersting, K., Nijssen, S., Železný, F. (Eds.): Machine Learning and Knowledge Discovery in Databases. Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD 2013). Springer, Lecture Notes in Computer Science, Vol. 8190, 2013, pp. 192–207.

[29] Handrich, S.—Al-Hamadi, A.: A Robust Method for Human Pose Estimation Based on Geodesic Distance Features. 2013 IEEE International Conference on Systems, Man, and Cybernetics (SMC '13), 2013, pp. 906–911.

[30] Zolfaghari, M.—Jourabloo, A.—Ghareh Gozlou, S.—Pedrood, B.—Manzuri-Shalmani, M. T.: 3D Human Pose Estimation from Image Using Couple Sparse Coding. Machine Vision and Applications, Vol. 25, 2014, No. 6, pp. 1489–1499.

[31] Vera-Perez, O. L.—Mesejo-Chiong, A.—Jaume-i-Capó, A.—González-Hidalgo, M.: Automatic Parameter Configuration: A Case Study on a Rehabilitation Oriented Human Limb Tracking Algorithm. Studies in Informatics and Control, Vol. 23, 2014, No. 1, pp. 87–96.

**Azam Andalib** received her B.Sc. degree in software engineering from Azad University of Lahijan in 2008 and her M.Sc. degree in information technology (IT) engineering from the University of Guilan in 2011. Now, she is Ph.D. student in software engineering at the University of Kashan, Iran. Her current research interests include software testing, formal methods, web services, decision making multi criteria, and artificial intelligence.

**Seyed Morteza Babamir** received his B.Sc. degree in software engineering from Ferdowsi University of Meshhad and his M.Sc. and Ph.D. degrees in software engineering from Tarbiat Modares University in 2002 and 2007, respectively. He was a researcher at Iran Aircraft Industries, Tehran, Iran, from 1987 to 1993, Head of Computer Center at the University of Kashan, Iran, from 1997 to 1999, and Head of Computer Engineering Department at the University of Kashan from 2002 to 2005. Now, he is Associate Professor at the Department of Computer Engineering at University of Kashan, Iran. He has authored one book on software testing, four book chapters, 20 journal papers and more than 40 international and internal conference papers (`http://ce.kashanu.ac.ir/babamir/Publication.htm`). He is the managing director of the Soft Computing Journal, which is publishing under auspices of University of Kashan, Iran.

**Alireza Faraji** is Assistant Professor of control engineering at the Electrical and Computer Engineering Department at the University of Kashan, Iran. He received his B.E. degree at Sharif University in Tehran, Iran and his M.E. and Ph.D. degrees in electrical engineering at Ferdowsi University of Mashhad, Iran. His current research interests include artificial intelligence, expert systems, intelligent and nonlinear systems and control.