Applied Soft Computing xxx (2015) xxx–xxx

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Shape feature encoding via Fisher Vector for efficient fall detection in depth-videos

Muzaffer Aslan [a], Abdulkadir Sengur [b,*], Yang Xiao [c], Haibo Wang [d], M. Cevdet Ince [b], Xin Ma [d]

[a] National Education Ministry, Gazi Industrial and Vocational High School, Elazig, Turkey
[b] Firat University, Technology Faculty, Electrical and Electronics Engineering Department, Elazig, Turkey
[c] National Key Laboratory of Science and Technology on Multi-spectral Information Processing, School of Automation, Huazhong University of Science and Technology, PR China
[d] School of Control Science and Engineering, Shandong University, Jinan, PR China

ARTICLE INFO

Article history:
Received 14 November 2014
Received in revised form
16 December 2014
Accepted 18 December 2014
Available online xxx

Keywords:
Fall detection
Shape contour
Curvature Scale Space
Fisher Vector encoding

ABSTRACT

Elderly people, who are living alone, are at great risk if a fall event occurred. Thus, automatic fall detection systems are in demand. Some of the early automatic fall detection systems such as wearable devices has a high cost and may cause inconvenience to the daily lives of the elderly people. In this paper, an improved depth-based fall detection system is presented. Our approach uses shape based fall characterization and a Support Vector Machines (SVM) classifier to classify falls from other daily actions. Shape based fall characterization is carried out with Curvature Scale Space (CSS) features and Fisher Vector (FV) encoding. FV encoding is used because it has several advantages against the Bag-of-Words (BoW) model. FV representation is robust and performs well even with simple linear classifiers. Extensive experiments on SDUFall dataset, which contains five daily activities and intentional falls from 20 subjects, show that encoding CSS features with FV encoding and a SVM classifier can achieve an up to 88.83% fall detection accuracy with a single depth camera. This classification rate is 2% more accurate than the compared approach. Moreover, an overall 64.67% accuracy is obtained for 6-class action recognition, which is about 10% more accurate than the compared approach.

1. Introduction

Falls are considered as an important public health problem, especially for the older adults whose ages are 65 and over. Statistics about the falls show that 33% of the older adults fall each year, which causes serious injuries such as hip fractures, lacerations and head traumas [1]. According to the emergency department's reports, 2.4 million elderly people who fall down were treated and more than one-third of these falls were hospitalized in 2012, which costs about $30 billion [1].

Elderly people, who are living alone, are at great risk if a fall event occurred and it is reported that about 3% of fallen elderly people are found helpless or dead at home [2]. Thus, fall detection at home or at other environments such as hospital room or aged care home has become an interesting research topic in the communities of computer vision and machine learning [3]. Such a system can ensure the safety of the elderly people who live alone.

In the last decade, a great number of automatic fall detection systems have been proposed [4–10]. The main aim of such systems is to monitor a fixed area where elderly people live and send a message to an emergency center or caregivers once a fall is detected. Some of the early works about the fall detection are based on the wearable sensors [11]. The main drawback of wearable sensors is the need to wear and carry some related devices such as batteries, during the day. The recent research trend about fall detection is vision-based approaches Most of the vision-based method use regular RGB cameras [12–15]. However, depth cameras, such as the Kinect sensor are recently popular and effective in fall detection based on its several advantages against regular RGB cameras. First of all, depth cameras have low price and have more functionality when compared with the regular RGB cameras [16,17]. The depth cameras preserve the privacy of the monitored person's life [17]. Moreover, changing lighting conditions does not affect the output of the depth cameras [18].

Up to now, there have been various depth sensor based automatic fall detection systems. In [19], the authors propose a two-staged fall detection approach based on Kinect sensor. Firstly, the observed individual's vertical state is modeled in the depth image frames and then segments on ground events from the vertical time series are obtained. The second stage uses an ensemble of decision tree classifier. Experimental comparisons show the superiority against its compared methods. In [20], Curvature Scale Space (CSS) features and Bag-of-Words (BoW) methods are combined to detect falls in depth videos. An improved extreme learning machine classifier is adapted to work and 86.83% fall detection accuracy is obtained. In [21], the authors propose an algorithm for fall detection using a ceiling-mounted 3D depth camera. Human area and shape's major length to width features are used to distinguish the lying pose and other common daily activities by using a k-NN classifier. The work presents a 0% error. Mean shift clustering (MSC) and depth connected components algorithms are used to detect fall in depth image sequences [22]. The human segmentation from the back-ground is achieved with the MSC. The presented experimental results show

* Corresponding author. Tel.: +90 424237000 4343.
E-mail address: ksengur@firat.edu.tr (A. Sengur).

http://dx.doi.org/10.1016/j.asoc.2014.12.035
1568-4946/

high accuracy. In [23], a fall detection system is proposed based on anthropometric relationships in a complex indoor environment. The obtained features are used to determine multiple human subjects between the detected segments. The experiments show promising results even in more complex indoor scenarios. Another depth-based fall detection system by considering the human skeleton is presented in [24]. The authors use two features such as distance between human skeleton joins and the floor and the join velocity. The first feature is considered for fall detection in depth frames due to the fact that fallen human skeleton joins are close to the floor level. The second feature is used to distinguish the natural falls from the lying activity.

There are also other fall detection approaches based on RGB cameras [12–15]. In [12], authors propose an algorithm based on body posture changes. Fall is detected if the body posture changes from standing to lying in a short period. The proposed approach can recognize standing, bending, sitting and lying actions with a fuzzy neural network classifier. In [13], fall detection is conducted based the human body postures. The human body silhouette bounding box parameters are used as features and a k-nearest neighbor (k-NN) approach is used for classification purposes. A fuzzy logic based fall detection approach is proposed in [14]. The authors recognize static postures such as lying, squatting, sitting and standing with Support Vector Machines (SVM) classifier. 74.29% recognition accuracy is reported. Another SVM based fall detection approach is proposed in [15]. Both using the shape and context information, the accuracy of fall detection system is highly improved. The reported accuracy is 97.08%.

In this paper, we present a novel approach for improving the depth based fall detection by encoding the CSS features with Fisher Vector (FV) representation. Especially, we aim to improve the detection performance of the approach that is proposed in [20] where CSS features and BoW are combined (BoCSS) to characterize the fall action. The CSS features are known to be invariant against the translation, rotation and scaling [25]. BoW is a popular image representation approach for classification task which consists of two stages such as local descriptor extraction and assigning each descriptor to the closest dictionary [26]. The dictionary is constructed with the popular k-means clustering algorithm. In this work, we propose to use FV encoding instead of the BoW for further performance improvement. FV can be regarded as the extension of BoW. In FV, the Gaussian Mixture Model (GMM) is employed as the visual dictionary. The gradients of the log likelihood w.r.t. the parameters of GMM are computed to model the generative process of data samples. As a result, FV can be obtained by concatenating the partial derivatives w.r.t. all the Gaussians. Compared to BoW, FV can further leverage the performance, generally. The FV representation further has the following advantages [27]:

1) FV can be considered as a generalization of the BoW. In other words, BoW is a particular case of the FV. The additional gradients improve the FV's performance greatly.
2) Smaller codebooks can be used to construct the FV, which yields lower computational cost.
3) FV performs well even with simple linear classifiers.

In [20], the authors also improve an ELM structure to classify falls from the other daily activities such as bending, sitting, walking, squatting and lying. On the contrary, we do not try to improve any classifier based on the fact that FV leads to excellent results even with linear classifiers. Thus, we used a binary SVM to distinguish fall action with the other actions. Various experiments are conducted on the SDUFall dataset [28]. In [20], it is reported that the SVM obtain the worst classification result and average accuracy is 63.12%. On the contrary, we obtain 88.83% average accuracy with the SVM classifier.

The paper is organized as follows: In Section 2, we briefly introduced the concept of the CSS features, FV and SVM. In Section 3, details about the proposed method are given. In Section 4, dataset and experimental works are described. The conclusions are drawn in Section 5.

2. Theoretical background

In this section, the theoretical background of the proposed work is given. The concepts of the CSS, FV and SVM are briefly reviewed. For their comprehensive descriptions, please refer to the related references.

2.1. CSS representation and features extraction

CSS is considered as an efficient presentation of the invariant geometric features of a given planar curve at various scales [25]. In order to compute the CSS, let's define a curve with a parametric vector equation as given in Eq. (1);

$$\Gamma(u) = (x(u), y(u)) \tag{1}$$

here, u is assigned arbitrarily. Then, the curvature function $\kappa(u)$ is defined as;

$$\kappa(u) = \frac{\dot{x}(u)\ddot{y}(u) - \ddot{x}(u)\dot{y}(u)}{(\dot{x}^2(u) + \dot{y}(u))^{3/2}} \tag{2}$$

If each component of Γ is convolved with a one dimensional Gaussian kernel $g(u,\sigma)$ of width σ, then $X(u,\sigma)$ and $Y(u,\sigma)$ represent the components of the resulting curve Γ_σ;

$$X(u,\sigma) = x(u) * g(u,\sigma) \tag{3}$$

$$Y(u,\sigma) = y(u) * g(u,\sigma) \tag{4}$$

where $*$ shows the convolution operator. The derivatives of each component can be computed as;

$$X_u(u,\sigma) = x(u) * g_u(u,\sigma) \tag{5}$$

$$X_{uu}(u,\sigma) = x(u) * g_{uu}(u,\sigma) \tag{6}$$

where $g_u(u,\sigma)$ and $g_{uu}(u,\sigma)$ are the first and second derivatives of $g(u,\sigma)$ with respect to u. We need to compute the similar derivatives for $Y(u,\sigma)$ as $Y_u(u,\sigma)$ and $Y_{uu}(u,\sigma)$. Based on the extracted equations, the curvature on Γ_σ can be defined as;

$$\kappa(u,\sigma) = \frac{X_u(u,\sigma)Y_{uu}(u,\sigma) - X_{uu}(u,\sigma)Y_u(u,\sigma)}{(X_u(u,\sigma) + Y_u(u,\sigma)^2)^{3/2}} \tag{7}$$

The CSS image of Γ is then defined at $\kappa(u,\sigma) = 0$, the zero crossing point of all Γ. Zero crossing point is invariant to rotation, translation and scaling because the curvature is calculated for several scales. On the CSS image, the (u,σ) coordinates of all zero crossing construct several continuous curves. The maxima point of each curve is concatenated to construct the CSS features [20].

2.2. Fisher Vector

Let's define a set of local features $X = \{x_t, t = 1, \ldots, T\}$ which is extracted from an image, where T indicates the number of features. In addition, let's assume that the generation process of X can be modeled by a probability density function u_λ with parameters λ [27,29]. X can be defined by the gradient vector [29];

$$G_\lambda^X = \frac{\nabla_\lambda \log u_\lambda(X)}{T} \tag{8}$$

The generation process can be described by the gradient of the log-likelihood. The number of parameters in λ directly determines the dimensionality of this vector and the number of patches T does not affect the dimensionality of the vector. A kernel on these gradients is [29]:

$$K(X,Y) = G_\lambda^X F_\lambda^{-1} F_\lambda^Y \tag{9}$$

where F_λ is the Fisher information matrix of u_λ;

$$F_\lambda = E_{x\sim u_\lambda}[\nabla_\lambda \log u_\lambda(x) \log u_\lambda(x)^{'}] \tag{10}$$

Since F_λ is positive semi-definite, using the Cholesky decomposition $F_\lambda = L'_\lambda L_\lambda$ and $K(X,Y)$ can be written as;

$$G_\lambda^X = L_\lambda G_\lambda^X \tag{11}$$

G_λ^X is the Fisher Vector of X. When probability density function u_λ is assumed to be modeled by GMM, $u_\lambda(x) = \sum_{i=1}^{K} w_i u_i$ and $\lambda = \{w_i, \mu_i, \Sigma_i, \quad i = 1, \ldots, K\}$ where w_i, μ_i, Σ_i are parameters of Gaussian u_i.

2.3. SVM

Let's consider a training pair with N samples, defined by $\{x_i, y_i\}$, $i = 1, \ldots, N$, with the data $x_i \in R^d$ and label $y_i \in \{-1, +1\}$. SVM tries

Please cite this article in press as: M. Aslan, et al., Shape feature encoding via Fisher Vector for efficient fall detection in depth-videos, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2014.12.035