HUMAN FALL DETECTION VIA SHAPE ANALYSIS ON RIEMANNIAN MANIFOLDS WITH APPLICATIONS TO ELDERLY CARE

Yixiao Yun, *Irene Yu-Hua Gu*

Dept. of Signals and Systems, Chalmers University of Technology, Sweden
{yixiao, irenegu}@chalmers.se

ABSTRACT

This paper addresses issues in fall detection from videos. The focus is on the analysis of human shapes which deform drastically in camera views while a person falls onto the ground. A novel approach is proposed that performs fall detection from an arbitrary view angle, via shape analysis on a unified Riemannian manifold for different camera views. The main novelties of this paper include: (a) representing dynamic shapes as points moving on a unit n-sphere, one of the simplest Riemannian manifolds; (b) characterizing the deformation of shapes by computing velocity statistics of their corresponding manifold points, based on geodesic distances on the manifold. Experiments have been conducted on two publicly available video datasets for fall detection. Test, evaluations and comparisons with 6 existing methods show the effectiveness of our proposed method.

***Index Terms*—** Human fall detection, shape analysis, Riemannian manifolds, elderly care, assisted living

1. INTRODUCTION

Recent decades have witnessed the rapid growth of aged population in most countries. According to statistics, falling on the ground is one of the most vital risks for this age group which may lead to bone fracture, coma, and even death [1]. In these cases, emergent medical attentions are necessary after the fall. Since many people in this age group live alone, it can be difficult for them to seek help immediately, especially when severe injury or unconsciousness occur due to the fall. Automatic surveillance systems have drawn increasing research interests recently, aiming at automatically detecting falls and triggering alarms.

Many existing methods are based on wearable devices with motion sensors, such as accelerometers and gyroscopes [1], which produce reasonable results in fall detection. However, users could feel uncomfortable after wearing the device for a long time, or forget to wear apart from battery charging issues. Visual monitoring hence has some advantages.

Much effort has been made to detect human falls in videos. One way to address this problem is to analyze the bounding boxes that encompass the person in question in each frame. Debard *et al.* [2] extract 4 features from the bounding box to describe a fall, including aspect ratio, torso angle, center speed and head speed. An SVM classifier is employed to detect a fall using these features. Charfi *et al.* [3] define 14 features based on the bounding box such as height and width, aspect ratio, and centroid coordinates of the box. Transforms (e.g., Fourier, wavelet) are applied to these features before fall detection through SVM and AdaBoost classification. The major drawback is insufficient description of the motion from using the bounding box, and the performance is also heavily dependent on view angles. Another commonly adopted strategy is to exploit the wide spatial coverage of multiple cameras, or the depth information from depth cameras. Auvinet *et al.* [4] reconstruct a 3D volume of the person from 8 cameras based on camera calibration, and a fall is indicated if a large portion of the body volume is found near the ground for a certain period of time. Ma *et al.* [5] obtain human silhouettes from depth images and learn curvature scale space (CSS) features from them. Actions are represented by a bag of CSS words, and classified by the extreme learning machine (ELM) into falls and other actions. Stone and Skubic [6] model the vertical state of a 3D object in each depth image frames, and segment the time series in on-ground state from those in vertical state. Then, an ensemble of decision trees is used to compute a confidence that a fall occurs before an on-ground state. It is worth noting the trade-off between the performance and complexity (or cost) in multi-camera or depth-camera methods.

In this paper, we propose a novel scheme adopting manifold-based shape analysis in single camera views, with the following motivations and contributions.

Motivations: Instead of using bounding boxes, the focus here is shifted to the analysis of human shape inside the box. Since it is a broadly accepted intuition that a human shape deforms drastically in camera views while the person falls onto the ground, better features could be obtained by studying the rate of shape change in a certain time interval. A suitable metric is preferred for measuring the rate. Riemannian geometry fulfills this requirement, given the assumption that many image features including shape naturally reside on Riemannian manifolds. By converting the analysis of human shape dynamics in an arbitrary camera view to the study of velocity

statistics on a unified manifold for different camera views, it is expected that these features are less sensitive to view angles. This could lead to a simple and effective solution, without combining multiple cameras.

Contributions: A shape descriptor is introduced which represents shapes from different camera views as connect points lying on a unit n-sphere, one of the simplest Riemannian manifolds. Note that this is a unified manifold for all shapes from different views. Further, the characterization of shape deformations are formulated as velocity statistics of moving points on the manifold, based on geodesic distances.

The remainder of this paper is organized as follows: Section 2 briefly reviews the previous work that is closely related. Section 3 describes the proposed scheme in detail. Section 4 shows experimental results on two video datasets for human fall detection. Finally, Section 5 concludes the paper.

2. RELATED WORK

This section briefly reviews Riemannian geometry [7], and the unit n-sphere [8], for the sake of mathematical convenience in subsequent sections.

2.1. Riemannian Geometry

A *manifold* is a topological space as low dimensional subspaces embedded in a high dimensional space, which is locally similar to Euclidean space. In case of nonlinear manifolds which are not vector spaces, the usual Euclidean calculus and conventional statistics may not apply. However, a *differentiable manifold* equipped with a globally defined differential structure allows one to perform calculus on the manifold. Further, a *Riemannian manifold* is defined as a differentiable manifold where the tangent space at each point has an inner product that varies smoothly from point to point. That is, a Riemannian manifold possesses not only the differentiable structure that allows calculus to be done, but also a Riemannian metric that allows distances and angles to be measured on the manifold.

The *geodesic* is the shortest curve between two points on a manifold. Geodesics correspond to straight lines in Euclidean spaces. Hence, *geodesic distance*, the length of the geodesic, is considered as one of the most suitable distance measures between two points lying on a Riemannian manifold.

2.2. The Unit n-Sphere

The unit n-sphere, $\mathcal{S}^n$, is an n-dimensional sphere with a unit radius, centered at the origin of the $(n+1)$-dimensional Euclidean space. It can be considered as the simplest Riemannian manifold after the Euclidean space [9]. It inherits a Riemannian metric from embedding in $\mathbb{R}^{n+1}$. Under this metric, the geodesic distance $d(\mathbf{x}, \mathbf{y})$ between two manifold points $\mathbf{x}$, $\mathbf{y} \in \mathcal{S}^n$ is the great-circle distance between the two points:

$$d(\mathbf{x}, \mathbf{y}) = \arccos(\mathbf{x}^T \mathbf{y}) \quad (1)$$

where $\arccos(\cdot)$ is the inverse cosine function [8]. Note that the great-circle distance between two points is unique.

The unit n-sphere finds its connection to vision tasks where the extracted feature vectors are in most cases normalized by the L_2 norm. The descriptors thus lie on a unit n-sphere $\mathcal{S}^n$, for some n. In cases where feature vectors are normalized block-wisely, the radius of the underlying sphere may not be unit. However, since any n-dimensional sphere centered at the origin is homeomorphic to $\mathcal{S}^n$, it turns out that they share exactly the same geometry [8]. Examples of feature descriptors lying on $\mathcal{S}^n$ include SIFT [10], HOG [11], LBP [12] and other histogram-based representations that are L_2 normalized.

3. FALL DETECTION VIA SHAPE ANALYSIS ON RIEMANNIAN MANIFOLDS

This section describes the major steps of the proposed method and the rationale behind.

3.1. Shape Representation on the Manifold

The shape descriptor is based on the histogram of oriented gradient (HOG) [11]. The basic idea is that object shape can often be characterized by the distribution of intensity gradients through voting the dominant edge directions.

For each input video, it is assumed that the person in question is tracked in tight bounding boxes through all frames. Based on a tracked image region $\mathbf{R}$ at time t that is of size $w \times h$ and centered at (x_0, y_0), our region of interest (ROI) is modified to be of size $l \times l$ but still centered at (x_0, y_0), where $l = \max(w, h)$. Then, our ROI is normalized to $\lambda \times \lambda$, where λ is a predefined length that is usually smaller than l for computational efficiency. In the normalized ROI, the corresponding tracked image region $\mathbf{R}'$ becomes of size $(w \cdot \lambda/l) \times (h \cdot \lambda/l)$. In this way, normalization of image size does not impact the aspect ratio of $\mathbf{R}$ inside our ROI.

Given the normalized ROI, histograms of oriented gradient are formed, similarly to HOG [11]. However, the difference here is that image gradients are only collected inside the tracked image region $\mathbf{R}'$. Thus, the background noise is effectively suppressed while keeping the aspect ratio of the foreground object. For a cell completely outside $\mathbf{R}'$, the histogram bins for that cell are assigned equal unit votes.

Finally, our shape descriptor is an n-dimensional vector $\mathbf{x}$ concatenating all elements of normalized histograms from all blocks. This vector $\mathbf{x}$ is a point residing on the unit n-sphere $\mathcal{S}^n$, i.e., $\mathbf{x} \in \mathcal{S}^n$.

3.2. Statistics on the Shape Manifold

Since the shape of a person tracked in each frames is represented as connected points on a Riemannian manifold, i.e., $\mathcal{S}^n$, the analysis of human shape dynamics are thus converted to the study of statistics on that manifold. Intuitively, the

more drastically the shape deforms, the more rapidly the corresponding manifold point moves.

To analyze the dynamics of points on the manifold, a temporal sliding window $\mathcal{X}$ of maximum length L is introduced, where a set of manifold points are collected, which correspond to the shapes of the person tracked in each frames up to time T:

$$\mathcal{X} = \begin{cases} \{\mathbf{x}_t\}_{t=1}^{T} & \text{if} \quad T \leq L \\ \{\mathbf{x}_t\}_{t=T-L+1}^{T} & \text{if} \quad T > L \end{cases} \tag{2}$$

where $\mathbf{x}_t \in \mathcal{S}^n$.

The instantaneous velocity v of a point $\mathbf{x} \in \mathcal{S}^n$ moving on the manifold is defined as

$$v = \frac{d(\mathbf{x}_{t-1}, \mathbf{x}_t)}{\Delta t} \tag{3}$$

where $\mathbf{x}_{t-1}$ and $\mathbf{x}_t$ are the samples of $\mathbf{x}$ at time $(t-1)$ and t, $d(\mathbf{x}_{t-1}, \mathbf{x}_t)$ is the geodesic distance between them defined in Eq. (1), and Δt is the time step. By assuming unit time step between consecutive frames, the velocity simply becomes the geodesic distance between two neighboring points in $\mathcal{X}$.

The feature vector $\mathbf{f}$ for each temporal window $\mathcal{X}$ is a concatenation of simple statistics for the velocity of moving points in $\mathcal{X}$:

$$\mathbf{f} = [\, \mathbb{E}[V],\ \text{Var}[V],\ \max(V),\ \min(V),\ R(V),\ \tau \,]^T \tag{4}$$

where V is a velocity variable that takes value v according to Eq. (3), $\mathbb{E}[V]$ and $\text{Var}[V]$ are the mean and variance of velocities measuring the average speed and dispersion of shape deformations, $\max(V)$ and $\min(V)$ are the maximum and minimum velocities, $R(V) = \max(V) - \min(V)$ is the velocity range in $\mathcal{X}$, and τ is the *Kendall rank correlation coefficient* [15] between $\{V\}$ and its sorted copy in ascending order, which estimates the likelihood of escalated shape deformations.

3.3. Fall Detection by Classification

Fall detection is formulated as a binary classification problem, that distinguishes the fall from other activities. That is, all remaining activities are treated as one negative class. The negative class consists of walking, crouching down (squatting), getting seated, lying on sofa, etc. Walking is considered since it is one of the most common activities in daily lives. Activities such as crouching down (squatting), getting seated and lying on sofa are chosen as they are close to falls which may cause confusion to the classifier. This would lead to better evaluations of the robustness of the proposed scheme.

Given a feature vector $\mathbf{f}$ in Eq. (4) which characterizes the shape deformation of a tracked person within a temporal window $\mathcal{X}$, its class label $c \in \{-1, +1\}$ is determined according to the decision rule as follows:

$$c = \text{sgn}(\alpha) \tag{5}$$

where $\text{sgn}(\cdot)$ is a sign function, and α is the output margin of a binary SVM classifier [16] by taking $\mathbf{f}$ as the input, where a fall is indicated as $c = +1$.

4. EXPERIMENTS AND RESULTS

4.1. Fall Detection Datasets

Experiments have been conducted based on two video datasets [17] [18] that are publicly available.

Dataset-A is built upon "multiple cameras fall dataset" [17] that contains 24 scenarios recorded by 8 IP cameras. In our tests, videos from different camera views are mixed, where temporal windows of video (video segments) containing human falls are selected as positive samples while those containing other activities are selected as negative samples.

Dataset-B is collected from "UR Fall Detection Dataset" [18] that contains 30 fall scenarios measured by two Kinect sensors (parallel to the floor and ceiling mounted, respectively) and an accelerometer, as well as 40 other activities measured by one Kinect sensor parallel to the floor. In our tests, only RGB data in *Dataset-B* is used, corresponding to 60 videos containing human falls (positive samples) and 40 videos containing other activities (negative samples).

Quantitative specifications on *Dataset-A* and *Dataset-B* are given in Tables 1 and 2. Fig. 1 and 2 show some keyframes of videos from each dataset.

Table 1. Quantitative specifications on *Dataset-A*.

Class #	Activity	# Video Segments	Total # Video Segments
1	Human falls	184	184
2	Crouching down/Squatting	40	216
	Getting seated	48	
	Lying down	48	
	Other	80	

Table 2. Quantitative specifications on *Dataset-B*.

Class #	Activity	# Video Segments	Total # Video Segments
1	Human falls	60	60
2	Crouching down/Squatting	8	40
	Bending over	7	
	Getting seated	9	
	Lying down	16	

4.2. Setup

All image sequences are temporally downsampled by 7. To exclude the impact of possible tracking drifts, bounding boxes containing the person in question are manually marked in each frame. All marked image regions are normalized to 32×32, i.e., $\lambda = 32$. The maximum length of each temporal sliding window (video segment after temporal downsampling) $L = 5$. A C-SVM classifier using RBF kernels [16] is employed, with the regularization coefficient $C = 4$ and the kernel parameter $\gamma = 0.0625$ (obtained via cross-validation on the training set). For each dataset, all video segments are partitioned into 2 parts, where approximately 50% in each class are used for training, and the remaining ones are used for testing.

Fig. 1. Keyframes from *Dataset-A* [17]. Upper row: human falls in various ways. Lower row: other activities including walking, crouching down (squatting), getting seated and lying down on sofa. Note that all images are zoomed in for better inspection of the person.

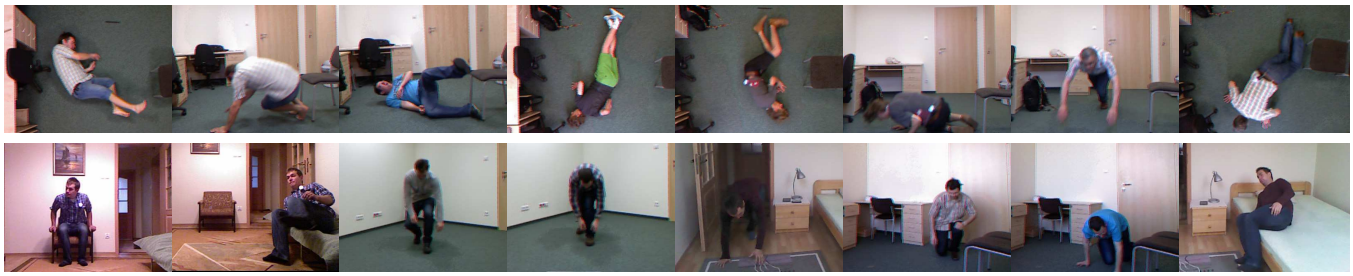

Fig. 2. Keyframes from *Dataset-B* [18]. Upper row: human falls in various ways. Lower row: other activities including crouching down (squatting), bending over, getting seated and lying down. Note that all images are zoomed in for better inspection of the person.

4.3. Test Results and Comparisons

The performance of the proposed fall detection scheme is evaluated according to detection rate (true positive rate, TPR) and false alarm rate (false positive rate, FPR) [19] on the test set of each dataset, as shown in Table 3.

Table 3. Performance of proposed fall detection scheme: detection rate and false alarm rate on the test set of each dataset.

Method	Dataset	#Video Segments	Detection Rate	False Alarm Rate
Proposed	*A*	200	91.30%	8.33%
	B	50	96.77%	10.26%

Further, comparisons are made with 6 existing methods in terms of sensitivity (TPR) and specificity (true negative rate, TNR) [20], as shown in Table 4.

Table 4. Comparison of different methods: sensitivity (Sens) and specificity (Spec) on the test set of each dataset. *Accel.*: Accelerometer. *Arbitrary RGB View*: using single RGB camera from arbitrary view angles, where videos from different camera views are mixed.

(a) *Dataset-A*

Method	Sensor Type	Sens (%)	Spec (%)
Auvinet *et al.* [4]	Multi. RGB Views	80.6	100
Rougier *et al.* [21]	Multi. RGB Views	95.4	95.8
Hung *et al.* [22]	Multi. RGB Views	95.8	100
Ma *et al.* [5]	RGB + Depth	99.93	91.97
Proposed	Arbitrary RGB View	91.30	91.67

(b) *Dataset-B*

Method	Sensor Type	Sens (%)	Spec (%)
Kepski *et al.* [18]	Depth + Accel.	100	96.67
Bourke *et al.* [23]	Accel.	100	90.00
Proposed	Arbitrary RGB View	96.77	89.74

In Table 3, the proposed scheme shows a high detection rate while maintaining small false alarm, despite the mixture of videos from different camera views. In Table 4, although our method does not consistently outperform the other methods that are based on multiple camera calibration or multiple modality information, it still produces comparable results, especially considering the fact that only single arbitrary camera view is employed in our method, and that videos from different camera views are mixed in our tests.

Discussion:

(i) This paper uses one of the simplest Riemannian manifolds, n-sphere, for shape representation. However, it is not limited to n-sphere. The geodesic distance $d(\cdot,\cdot)$ in Eq. (3) can be replaced accordingly (instead of Eq. (1)), if other types of Riemannian manifolds are considered.

(ii) Geodesic distance is employed since it suffices our purpose. Nevertheless, it would also be interesting to test on other distance measures, e.g., biharmonic distance [24], Stein divergence [25].

(iii) Considering the fact that only single arbitrary camera view is employed in our method, and that videos from different camera views are mixed in our tests, it may not be a fair comparison in Table 4 with methods that are based on multiple camera calibration or multiple modality information.

5. CONCLUSION

The proposed fall detection scheme characterizes human falls by shape features on a Riemannian manifold with simple velocity statistics. It is shown to be effective in obtaining high detection rate with low false positive rate in our experiments on two video datasets. Comparisons with 6 existing methods have provided further support to the robustness of the proposed scheme. Further study is planned on classification of more activities from videos.

6. REFERENCES

[1] X. Yu, "Approaches and principles of fall detection for elderly and patient," *IEEE International Conference on e-Health Networking, Applications and Services (HealthCom)*, pp. 42–47, 2008.

[2] G. Debard *et al.*, "Camera-based fall detection on real world data," *Outdoor and Large-Scale Real-World Scene Analysis*, pp. 356–375, 2012.

[3] I. Charfi *et al.*, "Optimized spatio-temporal descriptors for real-time fall detection: comparison of support vector machine and Adaboost-based classification," *Journal of Electronic Imaging*, vol. 22, no. 4, 041106, pp. 1–17, 2013.

[4] E. Auvinet *et al.*, "Fall detection with multiple cameras: an occlusion-resistant method based on 3-D silhouette vertical distribution," *IEEE Transactions on Information Technology in Biomedicine (T-ITB)*, vol. 15, no. 2, pp. 290–300, 2011.

[5] X. Ma *et al.*, "Depth-based human fall detection via shape features and improved extreme learning machine," *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 6, pp. 1915–1922, 2014.

[6] E.E. Stone, M. Skubic, "Fall detection in homes of older adults using the Microsoft Kinect," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 1, pp. 290–301, 2015.

[7] J.M. Lee, "Introduction to Smooth Manifolds," *Springer*, 2006.

[8] S. Jayasumana *et al.*, "Combining multiple manifold-valued descriptors for improved object recognition," *IEEE International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pp. 1–6, 2013.

[9] S.T. Lovett, "Differential Geometry of Manifolds," *A K Peters/CRC Press*, 1st edition, 2010.

[10] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision (IJCV)*, vol. 60, no. 2, pp. 91–110, 2004.

[11] N. Dadal, B. Triggs, "Histograms of oriented gradients for human detection," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 886–893, 2005.

[12] T. Ojala, M. Pietikäinen, T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, vol. 24, no. 7, pp. 971–987, 2002.

[13] D. Comaniciu, V. Ramesh, P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, vol. 25, no. 5, pp. 564–577, 2003.

[14] M.S. Arulampalam *et al.*, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing (T-SP)*, vol. 50, no. 2, pp. 174–188, 2002.

[15] M. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30, no. 1/2, pp. 81–93, 1938.

[16] C.J.C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[17] E. Auvinet *et al.*, "Multiple cameras fall dataset," *Technical Report*, no. 1350, Department of Computer Science and Operations Research (DIRO), University of Montreal, 2010.

[18] B. Kwolek, M. Kepski, "Human fall detection on embedded platform using depth maps and wireless accelerometer," *Computer Methods and Programs in Biomedicine*, vol. 117, no. 3, pp. 489–501, 2014.

[19] T.K. Moon, W.C. Stirling, "Mathematical Methods and Algorithms for Signal Processing," *Prentice Hall*, 1999.

[20] N.A. Macmillan, C.D. Creelman, "Detection Theory: A User's Guide," *Taylor & Francis*, 2004.

[21] C. Rougier *et al.*, "Robust video surveillance for fall detection based on human shape deformation," *IEEE Transactions on Circuits and Systems for Video Technology (T-CSVT)*, vol. 21, no. 5, pp. 611–622, 2011.

[22] D.H. Hung, H. Saito, "Fall detection with two cameras based on occupied area," *Japan-Korea Joint Workshop on Frontiers in Computer Vision (FCV)*, pp. 33–39, 2012.

[23] A.K. Bourke, J.V. O'Brien, G.M. Lyons, "Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm," *Gait & Posture*, vol. 26, no. 2, pp. 194–199, 2007.

[24] Y. Lipman, R. Rustamov, T. Funkhouser, "Biharmonic distance," *ACM Transactions on Graphics*, vol. 29, no. 3, 2010.

[25] S. Sra, "Positive definite matrices and the S-divergence," Preprint:(http://arxiv.org/abs/1110.1773), 2012.