FALL DETECTION IN RGB-D VIDEOS BY COMBINING SHAPE AND MOTION FEATURES

Durga Priya Kumar, Yixiao Yun, Irene Yu-Hua Gu

Dept. of Signals and Systems, Chalmers University of Technology, Gothenburg, SE-41296, Sweden
Emails: durga@student.chalmers.se, {yixiao, irenegu}@chalmers.se

ABSTRACT

This paper addresses issues in fall detection from RGB-D videos. The study focuses on measuring the dynamics of shape and motion of the target person, based on the observation that a fall usually causes drastic large shape deformation and physical movement. The main novelties include: (a) forming contours of target persons in depth images based on morphological skeleton; (b) extracting local dynamic shape and motion features from target contours; (c) encoding global shape and motion in HOG and HOGOF features from RGB images; (d) combining various shape and motion features for enhanced fall detection. Experiments have been conducted on an RGB-D video dataset for fall detection. Results show the effectiveness of the proposed method.

Index Terms— Fall detection, shape feature, contour descriptor, RGB-D videos, elderly care

1. INTRODUCTION

Population aging is taking place in nearly all countries of the world, with a considerably high rate of growth, and many people in this age group choose to live alone [1]. Statistics show that falling is one of the most fatal threats for them, which may cause bone fracture, coma or even death [2]. It can often be difficult for themselves to seek help immediately after the fall, especially when severe injury or unconsciousness occur. Hence, there is an increasing demand for automatic surveillance systems that aim at automatically detecting falls and triggering timely alarms for emergency medical treatment.

Many existing solutions employ wearable devices with motion sensors, such as accelerometers and gyroscopes [2]. Despite the reasonable results achieved by wearables, they suffer from some vital user experience flaws, such as sync issues, limited battery life and uncomfortableness. Visual monitoring hence has some advantages, due to its non-invasive and less-disturbing nature.

Much effort has been made to detect human falls in videos. One way to address this problem is to analyze the 2-D bounding boxes containing the target person in each frame. Debard *et al.* [3] 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 falls using these features. Charfi *et al.* [4] define 14 features based on the bounding box such as height and width, aspect ratio, and centroid coordinates of the box. Transforms (Fourier, wavelet) are applied to these features before fall detection through SVM and AdaBoost classification. The major drawback of these methods is insufficient description of the shape or motion by using the rigid bounding box sorely, and the performance is also heavily dependent on view angles.

Another commonly adopted strategy is to represent the fall in 3-D settings. Auvinet *et al.* [5] 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. Mastorakis *et al.* [6] measure the velocity of target person based on the contraction or expansion of the width, height and depth of the 3-D bounding box, and detect a fall by thresholding the velocity. Stone and Skubic [7] 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 in 3-D modeling or multi-camera methods.

In this paper, we propose a novel scheme for human fall detection in RGB-D videos. Foreground human detection is done by RGB frame differencing, followed by using SURF keypoints to mark the blob boundary for defining the target bounding box. Instead of extracting structural features from rigid bounding boxes, we extract local shape and motion features from target contours, and fusing them with HOG- and HOGOF-based features encoding global shape and motion. This may lead to enhanced performance, without 3-D modeling or combining multiple cameras. The main contributions include: (a) forming contours of target persons in depth images based on morphological skeleton; (b) extracting local shape and motion features from target contours; (c) encoding global shape and motion in HOG and HOGOF features from RGB images; (d) combining various shape and motion features for enhanced fall detection. Experiments have been conducted on an RGB-D video dataset for fall detection.

The paper consists of following parts: Section 2 revisits some existing work to which our work is closely related. Section 3 describes each major step of the proposed method in detail. Section 4 shows some experimental results on an RGB-D video dataset. Finally, Section 5 concludes the paper.

2. REVIEW OF RELATED WORK

This section briefly reviews some previous work that our work is built upon, for conceptual and mathematical convenience in subsequent sections.

2.1. Existing Feature Detectors and Descriptors

Speeded-Up Robust Features (SURF) is a local feature detector and descriptor, originally proposed by Bay *et al.* [8], that is inspired by SIFT [9]. SURF uses Hessian-matrix approximation operating on the integral images to locate interest points (keypoints) that are invariant to image scaling, translation, and rotation, and partially invariant to illumination change and affine or 3D projection.

Histogram of Oriented Gradients (HOG) is a feature descriptor for object detection and classification, originally proposed by Dalal and Triggs [10]. The basic idea is that object shape can often be characterized by the distribution of intensity gradients through voting dominant edge directions. The HOG is computed on a dense grid of uniformly spaced cells and uses local contrast normalization in overlapping blocks.

Optical flow is the pattern of apparent motion that is contained in a visual scene. Two common techniques for optical flow estimation are the Horn-Schunk method and Lucas-Kanade algorithm [11]. Given the optical flow between two consecutive video frames, motion features can be extracted such as *Histogram of Oriented Optical Flow (HOOF)* [12].

2.2. Support Vector Machine

Support Vector Machine (SVM) is a classification method, developed under the statistical learning theory, for supervised training. A most commonly discussed form is SVM for binary classes [13]. Given a set of labeled feature vectors $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ where $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \{-1, +1\}$, an SVM aims to find a classifier that has the minimum generalization error on the test set. This is related to finding maximum margin hyperplane, formulated by

$$\min \left(\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N} \xi_i \right), \quad (1)$$
$$\text{s.t.} \quad y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1 - \xi_i, \quad \forall i;$$
$$\xi_i \geq 0, \quad \forall i,$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product, $\mathbf{w}$ is a weight vector, b is a bias, $C > 0$ is a regularization coefficient, and ξ_i is a slack variable. This optimization problem can be formed by Lagrange multiplier, and solved by applying quadratic programming to its dual form.

For nonlinear separable classes, a mapping ($\phi : \mathbb{R}^d \mapsto \mathcal{H}$) is usually applied to map the feature vectors $\mathbf{x}_i \in \mathbb{R}^d$ to a higher dimensional space. This produces a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ with an inner product (kernel function) $\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_{\mathcal{H}}$. In this way, classes may become more close to linearly separable.

3. PROPOSED FALL DETECTION SCHEME

This section describes each major step of the proposed method in detail.

3.1. Encoding Shape and Motion in HOG and HOGOF Features from RGB Images

Foreground human detection is done by differencing consecutive RGB frames. SURF keypoint detector is applied to detect interest points in difference images that are used for fixing the target bounding box (BB) of size $w \times h$. The region of interest (ROI) of size $w \times h$ (defined by BB) is normalized to a fixed size of $\lambda \times \lambda$, by adopting the method from [14] and [15]. For each frame, the normalized ROI is used for extracting HOG features. For each pair of normalized ROIs from consecutive frames, optical flow is estimated and *Histogram of Oriented Gradients of Optical Flow* (HOGOF) is computed (illustrated in Fig. 1). The basic idea is similar to HOG that object motion can be represented by the distribution of optical flow as the votes for dominant directions of movement. By stacking the extracted HOG or HOGOF features temporally, the spatio-temporal features for shape or motion are obtained.

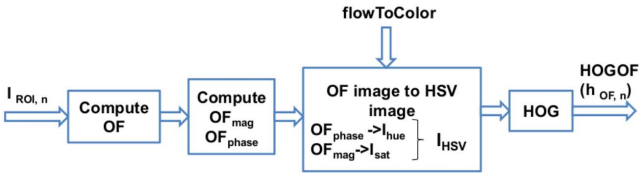

Fig. 1. Flow chart for the extraction of HOGOF features, where "OF" is the computed optical flow, "OF_{mag}" and "OF_{phase}" are the optical flow magnitude and direction, "I_{sat}" and "I_{hue}" are the saturation and hue images that are color-coded by MATLAB function flowtoColor() based on HSV model, "I_{HSV}" is the combined HSV image in RGB color space, "HOG" is the exactly the same process of extracting HOG features.

3.2. Extracting Shape and Motion Features from Target Contours in Depth Images

Skeletons of the target person are formed from the corresponding ROIs in each depth image frame using morphological operations. Target contours are obtained by taking endpoints of the skeleton, resulting in 8 extrema points (blue dots in Fig. 2). Based on target contours and their bounding boxes, local shape and motion features can be extracted.

For each frame, the local shape feature vector has the following form:

$$\mathbf{f}_s = \left[\mathbf{g}, \{\mathbf{E}_i\}_{i=1}^8, \{d_i\}_{i=1}^8, \theta, AR, Ecc\right]^T, \quad (2)$$

where $\mathbf{g}$ is the coordinate of contour centroid (yellow dot in Fig. 2), $\mathbf{E_i}$ are the coordinates of extrema points, d_i is the distance (green line in Fig. 2) from each extrema point to the centroid, θ and AR are the orientation and aspect ratio of the box, and Ecc is the eccentricity.

For each pair of consecutive frames, the local motion feature vector is defined as

$$\mathbf{f}_m = \left[\{\nabla d_i\}_{i=1}^8, \{k_i\}_{i=1}^8, k_g\right]^T, \quad (3)$$

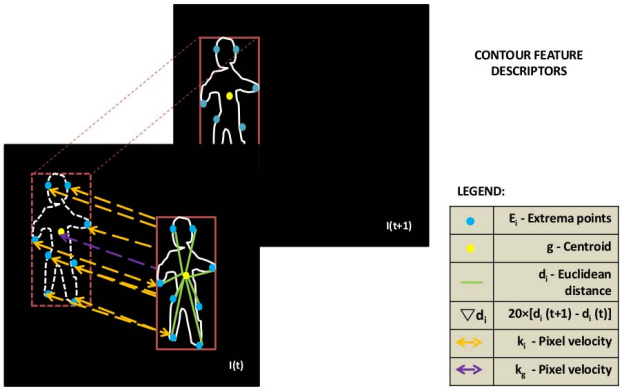

Fig. 2. The illustration of feature extraction from target contour. I_t and I_{t+1} are two consecutive frames.

where $\nabla d_i = \eta \times \dfrac{d_i(t+1) - d_i(t)}{\Delta t}$ is the gradient of distance d_i, $k_i = \eta \times \dfrac{\text{dist}[\mathbf{E}_i(t+1), \mathbf{E}_i(t)]}{\Delta t}$ and $k_g = \eta \times \dfrac{\text{dist}[\mathbf{g}(t+1), \mathbf{g}(t)]}{\Delta t}$ are the instantaneous velocity of extrema points and the centroid respectively, η is the frame rate, Δt is the time step ($\Delta t = 1$), and $\text{dist}(\cdot, \cdot)$ is the Euclidean distance between two points.

3.3. Combining Shape and Motion Features

To combine the discriminative power of different features, a simple feature-level fusion scheme is adopted here by concatenating all features into an augmented feature vector

$$\mathbf{f} = \left[\mathbf{f}_{\text{HOG}}^T, \mathbf{f}_{\text{HOGOF}}^T, \mathbf{f}_{\text{Contour}}^T\right]^T, \tag{4}$$

where $\mathbf{f}_{\text{HOG}}$ and $\mathbf{f}_{\text{HOGOF}}$ are the HOG and HOGOF features encoding global shape and motion using RGB images, and $\mathbf{f}_{\text{Contour}} = [\mathbf{f}_s^T, \mathbf{f}_m^T]^T$ is the concatenation of local shape and motion features extracted from target contours in depth images. Fusing various shape and motion features may lead to enhanced fall detection.

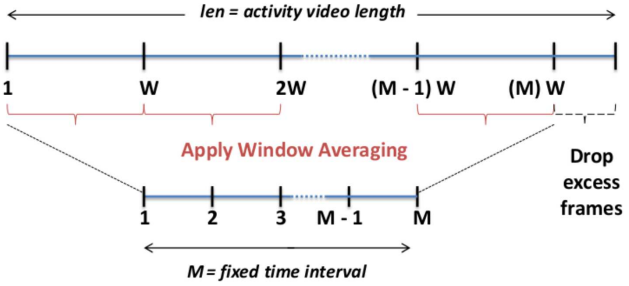

Fig. 3. The illustration of temporal window averaging, where each video segment of variable length len is divided into M (fixed value) non-overlapping windows of equal size $W = \text{round}(len/M)$. After window averaging, each resulting video segment will have a fixed length of M.

The final feature vector for each video segment is obtained by temporally stacking the feature vectors extracted from each frame. However, each video segment may have varying number of frames. Temporal window averaging is performed to keep all video segment outputting feature vectors of the same length, as shown in Fig. 3. That is, given a video segment containing len frames, feature vectors in the form of (4) from every $W = \text{round}(len/M)$ frames are averaged, followed by feature normalization, yielding $\mathbf{f}_{\text{avg}}^j$, $j = 1, \cdots, M$. Thus, the final feature vector for each video segment will have fixed length of M:

$$\mathbf{x} = \left[\mathbf{f}_{\text{avg}}^1, \mathbf{f}_{\text{avg}}^2, \cdots, \mathbf{f}_{\text{avg}}^M\right]^T. \tag{5}$$

3.4. Detecting Falls by SVM Classifier

In this work, fall detection is formulated as a binary classification problem (total number of classes $K = 2$) that distinguishes the fall from other activities. That is, all remaining activities are treated as one negative class. Given a training set $\mathcal{X} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$, where $\mathbf{x}_i$ is the feature vector for the i-th video segment in the form of (5), $y_i \in \{+1, -1\}$ is the corresponding class label, and N is the total number of video segments in the training set. A binary SVM classifier [13] is trained with $\mathcal{X}$. For each feature vector $\mathbf{x}$ representing a testing video segment, its class label $\hat{y} = \text{sgn}(a)$, where $\text{sgn}(\cdot)$ is a sign function, and a is the output margin of the SVM classifier, where a fall is indicated as $\hat{y} = +1$.

4. EXPERIMENTAL RESULTS

This section shows the experimental results on an RGB-D video dataset for fall detection using the proposed method.

4.1. RGB-D Video Dataset on Fall Detection

Experiments have been conducted on an RGB-D video dataset built by ourselves at Chalmers University of Technology using a *Kinect* sensor. A total of 20 participants are involved to perform the actions of falling and lying down. Lying down is considered as it is visually more confusing thus more difficult to be distinguished from a fall than any other activities (e.g., walking, running, getting seated, crouching down). For both RGB and depth streams, the frame rate is 20 FPS and the resolution is 640×480 pixels. The average length of video is approximately 300-400 frames ($\approx$ 10 seconds). Detailed information on the dataset is given below.

Table 1. Quantitative specifications on the RGB-D video dataset.

Class#	Activity	#Subjects	#RGB Video	#Depth Video
1	Falling down	20	400	400
2	Lying down	20	400	400

As shown in Table 1, our dataset uses 800 RGB videos and 800 corresponding depth videos. Each video is pre-processed to only include the segment of activity. In our tests, the video events containing falls are selected as positive samples, while those containing lying down activities are selected as negative samples.

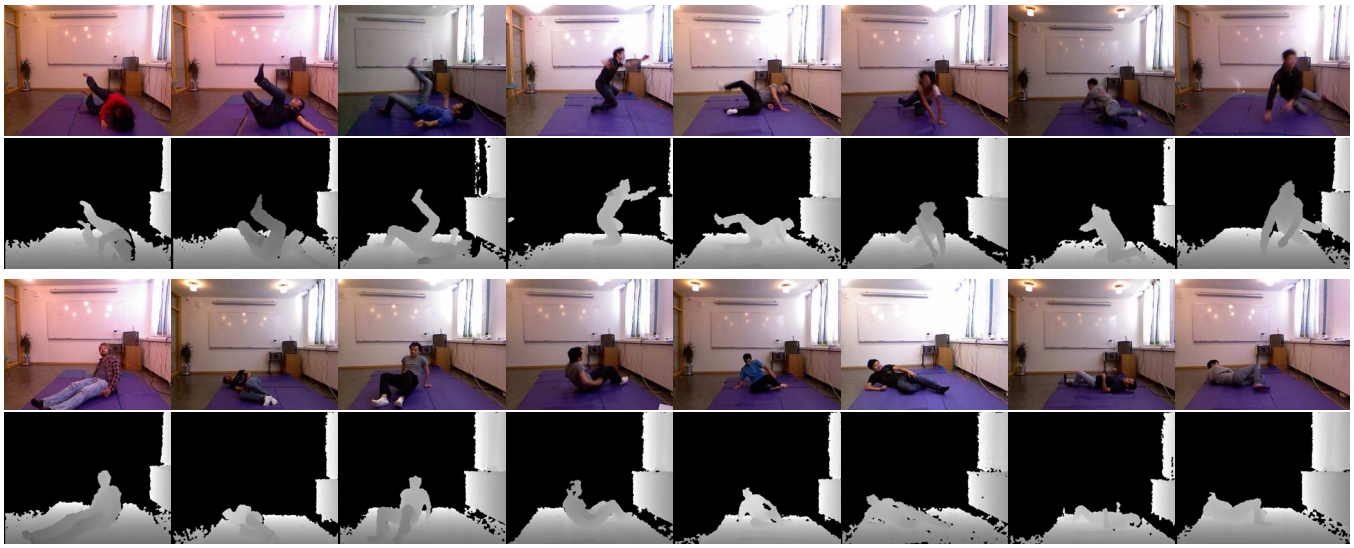

Fig. 4. Keyframes from the RGB-D video dataset on fall detection. Row 1-2: human falls in various ways in RGB and depth images. Row 3-4: other activities mainly containing lying down in RGB and depth images.

Fig. 4 depicts some keyframes of videos from our dataset. It can be observed that lying down activities can appear quite confusing in comparison with human falls.

4.2. Experimental Setup

Parameter Settings: The SURF thresholds for RGB and depth images are 50 and 1000, respectively. For ROI normalization, $\lambda = 32$. For HOG features, all ROIs are normalized and divided into non-overlapping cells of size 8×8. The number of histogram bins is 9 (unsigned). Blocks are formed by grouping 2×2 adjacent cells with overlapping rate 50%. HOGOF features are extracted based on the code from [16] and [17]. For temporal window averaging, $M = 10$.

Dataset Splitting for Training/Testing: A C-SVM classifier using RBF kernels [13] is employed, using libSVM [18], with the regularization coefficient C and the kernel parameter γ tuned by 10-fold cross-validation. Two case studies are carried out: (i) for case study-1, the SVM classifier is trained on 200 falls and 200 lying down activities (50%), and the remaining ones (50%) are used for testing; (ii) for case study-2, 320 falls and 320 lying down activities (80%) are used for training, and the remaining ones (20%) are used for testing. Note that subjects used for training are avoided in the test set.

4.3. Tests, Comparisons and Evaluations

The performance of the proposed fall detection scheme is evaluated according to detection accuracy, false negative rate and false positive rate (also known as false alarm rate) [19] [20] on the test set, as shown in Table 2. In Table 2, the proposed scheme shows high detection accuracy of human falls while maintaining small false negatives and false alarms on the test set, despite the confusion caused by lying down activities. In addition, comparisons are made with classification using standalone features in Table 2(a), i.e., $\mathbf{f}_{\text{HOG}}$, $\mathbf{f}_{\text{HOGOF}}$, $\mathbf{f}_{\text{Contour}}$, where the proposed fusion scheme outperforms in detection rate and false negatives. Further, it is observed in Table 2(b) that larger size of training set leads to better performance on the testing set.

Table 2. Performance evaluation: detection rate, false negative rate (FNR), and false positive rate (FPR) on the test set.

(a) Comparison between the proposed feature fusion and standalone features for case study-1.

Feature	Detection rate (%)	FNR (%)	FPR (%)
HOG	93.75	6.25	5.00
HOGOF	94.00	6.00	**4.00**
Contour	92.75	7.25	9.00
Fusion	**95.25**	**4.75**	5.00

(b) Comparison between case study-1 and case study-2.

Case	Detection rate (%)	FNR (%)	FPR (%)
Case study-1	95.25	4.75	5.00
Case study-2	**97.50**	**2.50**	**2.50**

Discussion: Video segments were manually chosen instead of automatically done. Such a setting is used for examining whether the proposed method is effective, without the impact of other parts that could cause overall performance degradation. If videos are automatically segmented, the performance of classification is expected to have some degradation.

5. CONCLUSION

The proposed fall detection scheme characterizes falls by measuring the dynamics of shape and motion of the target person, based on global features encoded in HOG and HOGOF using RGB images, and local features extracted from target contours in depth images. The fusion of these features is shown to be effective in obtaining high detection rate with small false alarms in our experiments on an RGB-D video dataset. Further study is planned on testing more datasets, extending to more activities and comparing with state-of-the-art methods.

6. REFERENCES

[1] United Nations, "World Population Ageing 2013," *Population Division, Department of Economic and Social Affairs (DESA), United Nations*, pp. 1–95, 2013.

[2] M. Mubashir, L. Shao, L. Seed, "A survey on fall detection: principles and approaches," *Neurocomputing*, vol. 100, pp. 144–152, 2013.

[3] G. Debard *et al.*, "Camera-based fall detection on real world data," *International Workshop on Theoretical Foundations of Computer Vision*, pp. 356–375, 2012.

[4] 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.

[5] 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.

[6] G. Mastorakis, D. Makris, "Fall detection system using Kinects infrared sensor," *Journal of Real-Time Image Processing*, vol. 9, no. 4, pp. 635–646, 2014.

[7] 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.

[8] H. Bay, T. Tuytelaars, L.V. Gool, "SURF: speeded up robust features," *European Conference on Computer Vision (ECCV)*, 2006.

[9] D.G. Lowe, "Object recognition from local scale-invariant features," *IEEE International Conference on Computer Vision (ICCV)*, vol. 2, pp. 1150–1157, 1999.

[10] 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.

[11] A. Bruhn, J. Weickert, "Lucas/Kanade meets Horn/Schunck: combining local and global optic flow methods," *International Journal of Computer Vision (IJCV)*, vol. 61, no. 3, pp. 211–231, 2005.

[12] R. Chaudhry *et al.*, "Histograms of oriented optical flow and Binet-Cauchy kernels on nonlinear dynamical systems for the recognition of human actions," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1932–1939, 2009.

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

[14] Y. Yun, I.Y.H. Gu, "Human fall detection via shape analysis on Riemannian manifolds with applications to elderly care," *IEEE International Conference on Image Processing (ICIP)*, 2015.

[15] Y. Yun, I.Y.H. Gu, "Human fall detection in videos via boosting and fusing statistical features of appearance, shape and motion dynamics on Riemannian manifolds with applications to assisted living," *Computer Vision and Image Understanding (CVIU)*, doi:10.1016/j.cviu.2015.12.002.

[16] C. Liu, "Beyond pixels: exploring new representations and applications for motion analysis," *Doctoral Thesis*, Massachusetts Institute of Technology, 2009.

[17] S. Baker *et al.*, "A database and evaluation methodology for optical flow," *Technical Report, Microsoft Research*, MSR-TR-2009-179, 2009.

[18] C.C. Chang, C.J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 27:1–27:27, 2011.

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

[20] D.M.W. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness & correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.