Author/s:
Rao, AS;Gubbi, J;Marusic, S;Palaniswami, M

Title:
Crowd Event Detection on Optical Flow Manifolds

Date:
2016-07-01

Citation:
Rao, A. S., Gubbi, J., Marusic, S. & Palaniswami, M. (2016). Crowd Event Detection on Optical Flow Manifolds. IEEE Transactions on Cybernetics, 46 (7), pp.1524-1537. https://doi.org/10.1109/TCYB.2015.2451136.

Persistent Link:
https://hdl.handle.net/11343/302073

Crowd Event Detection on Optical Flow Manifolds

Aravinda S. Rao, *Student Member, IEEE,* Jayavardhana Gubbi, *Senior Member, IEEE,* Slaven Marusic, and Marimuthu Palaniswami, *Fellow, IEEE*

***Abstract*—Analyzing crowd events in a video is key to understanding the behavioral characteristics of people (humans). Detecting crowd events in videos are challenging because of articulated human movements and occlusions. The aim of this study is to detect the events in a probabilistic framework for automatically interpreting the visual crowd behavior. In this work, crowd event detection and classification in Optical Flow Manifolds (OFM) is addressed. A new algorithm to detect walking and running events has been proposed, which uses optical flow vector lengths in OFM. Furthermore, a new algorithm to detect merging and splitting events has been proposed, which uses Riemannian connections in the Optical Flow Bundle (OFB). The longest vector from the OFB provides a key feature for distinguishing walking and running events. Using a Riemannian connection, the optical flow vectors are parallel transported to localize the crowd groups. The geodesic lengths among the groups provide a criterion for merging and splitting events. Dispersion and evacuation events are jointly modeled from the walking/running and merging/splitting events. Our results show that the proposed approach delivers a comparable model to detect crowd events. Using the PETS 2009 dataset, the proposed method is shown to produce the best results in merging, splitting, and dispersion events, and comparable results in walking, running, and evacuation events when compared with other methods.**

***Index Terms*—Video surveillance, crowd monitoring, event detection, optical flow, Riemannian manifolds.**

I. Introduction

Video analytics is very helpful in learning the behavioral characteristics of humans from videos. Detecting and predicting events in the videos is both exacting and challenging. Individual object detection and tracking is a challenging task in multi-object scenarios, and the difficulty increases further in crowded scenes [1]. In particular, event detection in crowded scenarios becomes complex when faced with articulated human movements and occlusions [2], [3]. Because the human population is increasing steadily, the management and control of crowds have gained importance. Automatic analysis of crowd behavior is important in the following applications: (a) crowd management: strategies to evacuate buildings and premises in case of disaster events, ingress and egress route planning from sporting amphitheaters, guiding disabled and infirm citizens, etc.; (b) public space design: the output from crowd analysis provides a valuable input for architects and construction teams for careful space utilization and efficient engineering plans; and (c) video surveillance: addresses detecting and alerting unexpected events. The primary goal of this work is to provide an automated video surveillance mechanism to detect crowd events.

Optical flow estimates the motion between a pair of frames in a given video [4]. The underlying principle is to match the likelihood of apparent motion between frames with respect to changes in brightness (pixel value). The optical flow approach has been used in crowd motion analysis [5], detecting crowd anomalies [6], [7], and facial expressions [8]. The optical flow vectors are low-level features; interpreting them as high-level events are computationally expensive, and the results can be very noisy. Video data are voluminous and the data have to be reduced to create real-time video surveillance applications.

A manifold is a topological space and manifold learning algorithms aim at representing the data in high-dimensional space to low-dimensional space by finding the mapping functions. In doing so, the data dimensions are reduced while preserving certain properties of the data. The properties that are preserved are purely based on the objective function. Dimensionality reduction techniques can be broadly categorized into linear and nonlinear techniques. Linear techniques, such as Principal Component Analysis (PCA), assume a linear data subspace compared with a nonlinear subspace considered by nonlinear techniques, such as Isometric Mapping (ISOMAP) [9]. A Riemannian manifold utilizes the classical Riemannian geometry comprised of certain metrics, such as inner products, and the concept of lengths and differentiability on the manifolds [10]. Optical Flow Manifolds (OFM) explore the optical flow space for various operations based on the concepts of classical differentiable manifolds. OFM are a novel approach to finding the intrinsic dimensions for image and vision applications. A recent work by Nagaraj *et al.* [11], provide a detailed theory of these concepts. In this work, concepts of Riemannian manifolds have been applied to OFM for crowd event detection. In this work, an extension to OFM, called the Optical Flow Bundle (OFB) is introduced. In short, OFB is the disjoint union of tangent spaces defined by OFM. A detailed definition of relevant concepts has been provided in Section III.

The crowd events targeted are running, walking, crowd formation (merging), splitting, local dispersion and rapid dispersion (evacuation) as defined by Performance Evaluation of Tracking and Surveillance (PETS) [12] (refer to Fig. 1). An automated event detection system is proposed by defining Riemannian manifold concepts on OFM. In [13] (shorter version), we showed the following with respect to crowd event detection: (1) the length of optical flow features can be used for event detection, (2) crowd events can be detected using Riemannian manifolds and (3) events can be detected

Manuscript received April 16, 2015; revised June 23, 2015; accepted June 26, 2015. This work is partially supported by the Australian Research Council (ARC) linkage project (LP100200430), partnering The University of Melbourne, Melbourne Cricket Club (MCC) and ARUP. The authors would like to thank representatives and staff of ARUP and MCC.

The authors are with the ISSNIP, Department of Electrical and Electronic Engineering, The University of Melbourne, Parkville, Victoria - 3010, Australia (e-mail: aravinda@student.unimelb.edu.au; jgl@unimelb.edu.au; slaven@unimelb.edu.au; palani@unimelb.edu.au).

(a) Walking (b) Running (c) Crowd formation (Merging) (d) Splitting (e) Local dispersion (f) Rapid dispersion (Evacuation)

Fig. 1: Examples of crowd events from the PETS 2009 [12] dataset: (a) walking : people walk across the scene from right to left; (b) running: people running from right to left; (c) merging: people from three different directions approach and merge at the center; (d) splitting: people from right move towards the left and the split up into three different groups; (e) dispersion: people standing in the center of the scene disperse locally outwards; (f) evacuation: people run outwards from the center of the scene in all directions.

without using tracking algorithms. The main objective of this work (extended version) is to detect the events in a probabilistic framework on OFM derived from Riemannian manifolds. The proposed work provides a theoretical treatment for detecting crowd events. The main contributions of the work are summarized below:

1) Tracking groups in the crowds can be problematic due to the number of people and articulated movements. A motion-based, probabilistic crowd event detection framework has been proposed. Although some initial models (e.g., probabilistic models [14] and histogram models [15]) have been proposed in this direction, the approach to detection of crowd events in this work is from OFM perspective. The framework makes use of only optical flow vectors to detect crowd events, which is in contrast to other methods that use, appearance, shape, audio, individual tracking and other spatio-temporal information. The method is also semi-supervised: the parameters learned from a particular view apply to other video sequences.
2) Video manifolds offer many advantages for crowd analysis. Riemannian and OFM offer natural parametric spaces for the detection of crowd events (mainly spatio-temporal in nature). OFM have been primarily used in action recognition to model the parametric space [11]. In contrast, the detection of crowd events using OFM is addressed in this work. A new algorithm to detect walking and running events has been proposed, which uses optical flow vector lengths in OFM.
3) Localization of crowd groups is a difficult problem in crowd monitoring and is essential for finding the group events in crowds (such as merging, splitting, local dispersion, and evacuation). A new algorithm to detect merging and splitting events has been proposed, which uses Riemannian connections in the OFB.

II. Related Work

Video behavior analysis has grown from human action recognition to anomaly detection and eventually to event detection. The taxonomy for human behavior analysis described by Chaaraoui *et al.* [16] provides the relevance of motion, action, activity and behavior. The keywords (motion, action, and activity) are often interchanged in the literature. This section presents a consolidated view that identifies the critical developments in event detection analysis, specifically noting that crowd analysis is in its infancy. Furthermore, most of the methods are based on motion estimation [17] and optical flow [4].

A. Human Action Recognition

Examples of commonly defined human actions include running, walking, skipping, doing jumping-jacks, jumping forward, jumping in the same place, jumping sideways, waving two hands, waving one hand, boxing, hand clapping, hand waving, and jogging as defined by the two frequently used action datasets: Weizmann [18] and Kungliga Tekniska högskolan (KTH) [19]. Kinematic features provide a natural reference to modeling human actions [20]. Another way of identifying individuals' actions is by identifying the body parts. Feature point-based approaches, e.g., [21], [22], use key features to detect the action. Spatio-temporal invariant features (STIPs) have also been extensively used in action recognition [23], [24]; audio-visual features were also utilized in addition to STIPs [25]; multi-channel STIPs were incorporated into an Histogram of Gradients (HOG) based 3D descriptor [26]. These approaches require predetermined feature training and tracking for determining actions. It is clear that the self-occlusions and inter-object occlusions can reduce the effectiveness of these approaches. Silhouette-based method, such as [27], faces challenges in extracting the silhouettes because it is a critical step for feature extraction. Supervised learning-based approaches (e.g., [28], [29]), use discriminating features for training and Support Vector Machines (SVM) for classification. For automated surveillance, supervised approaches are less attractive because they require retraining if the view or the scene is changed. Manifold learning-based approaches are the most widely used technique for unsupervised action recognition [30], [31]. Unsupervised approaches have an edge over supervised methods in terms of practicality because of their straightforward readiness (zero or less training) for automated applications.

B. Crowd Anomaly Detection

In general, anomaly detection operates on temporal domain data to identify the outliers or events [32]. There has been ongoing research on anomaly detection, where the system

classifies unusual events (hereafter unusual, abnormal events are called as anomalous) [33], [34], [35], [36]. Some of the crowd anomalies addressed in the literature include loitering about a particular place, a person collapsing, or a person running when the rest is walking. Anomalous events are contextual and are relative to the other objects in the scene [37], [32]. Emergency events, such as crowd panic or a threat to human life, create anomalies. Identifying these events is important for video surveillance applications. Spectral clustering-based approaches [38], social force model [39], hierarchical clustering [35], and K-means clustering [40] are widely used. Optical flow-based methods use spatio-temporal analysis where motion is used for the detection of anomalous events [34], [41], [42], [43], [15]. Region-based anomalous motion approaches [44], limit the motion information to particular regions. Sliding-window approaches [36], a combination of spatial, temporal and motion information, limit the anomalous events to structured and time-based events. As mentioned earlier, human action recognition involves recognizing the actions of an individual. However, anomaly detection attempts to detect the actions of an individual relative to the crowd.

C. Video Event Detection and Crowd Analysis

Video event detections are usually used to search for a specified action. Because this process involves detecting and matching actions, human action, anomalous events and crowd events are utilized. Visual (color, texture and shape) and audio (timbre, rhythm and pitch) features are normally used for video event detection [45]. Because the events have both spatial and temporal information, texture features from spatial information [46], motion features from spatio-temporal information and color [47], [48], and mixtures of texture and motion [49] are utilized. Volumetric analysis is a rising trend in video event detection [50], [51].

Chen *et al.* [52] applied an agent-based technique to detect queuing, gathering and dispersion events with the aid of tracking. It incorporates head features, template matching, Kalman filtering and SVM for object agent analysis. Five types of actions were defined using four people: walking, running, jumping, squatting and stopping on a locally collected dataset. From the review of the literature, this work appears to be the first of its type targeting event detection and behavior analysis. Almost all of the methods that have been proposed since then have been tested using the PETS 2009 dataset [12]. Li *et al.* [53] proposed a data-driven Discriminative Temporal Interaction Manifold (DTIM) framework to analyze group patterns as opposed to the parametric Bayesian framework. The framework generates probability densities that indicate the activities among the groups (of objects) with applications to a soccer game. Gárate *et al.* [54] used a reference frame to extract motion information and 2-D HOG descriptors as features. These features were tracked to categorize the crowd events. Occlusions in crowded scenarios make tracking infeasible. Benabbas *et al.* [14] used optical flow to extract motion patterns and build a direction and a magnitude model for crowd event detection. Furthermore, dominant directions are learned by circular clustering using a probabilistic model and the magnitude model is refined using online Gaussian Mixture Model (GMM). Blocks with similar magnitude and direction in a neighborhood are clustered and tracked for crowd event detection. Employment of group tracking to track a centroid in each frame makes this approach less attractive because tracking introduces (a) additional computation and grows significantly when the number of groups increases, and (b) tracking errors lead to overall system errors. Thida *et al.* [15] used blocks of Histogram of Optical Flow (HOOF) for each frame and compared this result with the neighboring frames. Based on this spatial and temporal information, crowd events were detected using Laplacian Eigenmaps. The main drawback of this approach is that the motion direction is assumed to provide information about various crowd events and also low-dimensional embedding is found using a time-controlled parameter. Li *et al.* [55] performed the crowd event detection using the intersection of motion vectors derived from Harris corner point and Kanade-Lucas-Tomasi (KLT) feature tracking. The events were classified based on the motion vector patterns at local intersection points in the space and membership event voting. The limitation of this approach is that tracking of feature points becomes cumbersome and often occluded due to crowd movements. It is clear from this discussion that limited research has been conducted in detecting and predicting events. It is also worth noting that most of the methods use training data to classify the events.

III. Video Manifolds

The reduction of dimensionality involves reducing the number of latent variables required to represent a point in a given space, and corresponds to the intrinsic dimensionality (structure) of the data [56]. In the context of video manifolds, given a set of frames as input, the objective is to identify the predefined human events in a given dataset. The hypothesis is that the events lie in a low-dimensional feature space; the video frame is a $\mathbb{R}^{5\times m\times n}$-dimensional data, where the pixel color information is $(r, g, b) \subset \mathbb{R}^3$. The spatial positioning $(x, y) \subset \mathbb{R}^2$, which is parameterized by the sampling interval of the frame, $t \subset \mathbb{R}$, and the number of rows and columns are indicated by the $m-$ and $n-$dimensions. Representation of a pixel for monocular vision can be generalized as a 5D vector, $I(r(t), g(t), b(t), x(t), y(t))$ and as a 6D vector, $I(r(t), g(t), b(t), d(t), x(t), y(t))$ for stereo vision. The input data consisting of events are analyzed in high-dimensional space and represented as low-dimensional data, such as probability outputs that are one-dimensional. In this work, there are three probabilistic models that generate outputs each in $\mathbb{R}^1$. Together, we can represent the entire system input-output as $\mathbb{R}^{5\times m\times n} \rightarrow \mathbb{R}^3$.

A. Manifolds

Intuitively, a manifold is a space that is Euclidean locally, i.e., a point in this space can be represented unequivocally, and appears to be in the Euclidean space [57]. For example, a three-tuple (x, y, z) that represents a point in the 3D space is threemanifold, where the intrinsic dimension of the space is three, which also implies that a point can be specified

without ambiguity. However, in the case of parameterized space, the combination of different parameters can represent the same point uniquely. For example, a twomanifold $(x, y) = (f_1(t), f_2(t))$ parameterized by t such that $x^2 + y^2 = 1$ may correspond to the same unit circle for a different t. Generalizing the notion of a manifold to higher-dimensional space, we denote an n-tuple in $\mathbb{R}^n$ as $n-$dimensional manifold or $n-$manifold. The following text provides some of the definitions (based on [10]) that are necessary for modeling event detection (readers familiar with manifold concepts can skip the definitions in this subsection).

Definition 1. *A real-valued function $f : \mathbb{R}^m \to \mathbb{R}^n$ is said to be differentiable at $p \in \mathbb{R}^m$ if there is a linear transformation T_p such that $T_p : \mathbb{R}^m \to \mathbb{R}$*

$$\lim_{\|t\|\to 0} \frac{\|f(p+t) - f(p) - T_p(t)\|}{\|t\|}. \tag{1}$$

The (linear) transformation T_p is called the derivative of f at p.

Definition 2 (Manifolds). *Let U be an open subset of the manifold $\mathcal{M}$ ($U \subset \mathcal{M}$). Let ϕ be a homeomorphism such that $\phi : U \to \mathcal{M}$. Then, (U, ϕ) forms the coordinate chart for the m-dimensional manifold $\mathcal{M}$.*

Definition 3 (Tangent Vector). *Let $\mathbb{R}^m$ be an m-dimensional manifold. Let p be a point in $\mathbb{R}^m$. Let $c = (x^1, x^2, \cdots, x^m)$ be a differentiable curve of class C^∞ such that $c : I \to \mathbb{R}^n$ with $c(0) = p$. Then, the Optical Flow Vector is given by $v_p = \dot{c(0)} = (\dot{x^1}, \dot{x^2}, \cdots, \dot{x^m})$.*

Definition 4 (Tangent Space (geometric)**).** *Let $\mathbb{R}^m$ be an m-dimensional manifold. Let p be a point in $\mathbb{R}^m$. Let c be a differentiable curve of class C^∞ such that $c : I \to R^n$ with $c(0) = p$. Then, the tangent vector of c at $p \in \mathbb{R}^n$ is*

$$Dc(0) = \dot{c}(0) = \lim_{t\to 0} \frac{\dot{c}(t) - c(0)}{t}. \tag{2}$$

Definition 5 (Tangent Space (analytic)**).** *Let (U, ϕ) be the coordinate chart with $p \in U$ for an m-dimensional manifold $\mathcal{M}$. Then, the tangent space $T_p\mathcal{M}$ is a derivation of $C^\infty(\mathcal{M})$ at a point $p \in \mathcal{M}$ such that $v : C^\infty(\mathcal{M}) \to \mathbb{R}$, where v is the vector at point p.*

B. Optical Flow Manifolds

A brief review of the popular optical flow methods has been provided here. Horn and Schunck [4] estimated the motion between images by applying brightness constancy, which is a dense approach but provides smooth flow vectors. Lucas and Kanade [58] considered the motion in the local neighborhood to be constant and the motion was computed using a least-squares approach, which is a sparse flow computation approach. Farnebäck [59] used a second-degree polynomial velocity estimation model. Zach *et al.* [60] applied the method of the total variations (TV) using the L_1 norm of penalizing the flow variations as opposed to the quadratic approach taken by Horn-Schunck [4]. Tao *et al.* [61] used a probabilistic approach based on local evidence (color constancy) to compute the motion vectors.

In the optical flow case, the horizontal and vertical velocities (v_x, v_y) naturally form the directional derivatives along x and y directions with some additional constraints being applied. The rationale behind this argument is that with a simple curve c (as shown in Fig. 2), the tangent vector v follows the direction along the curve. Thus, computation of optical flow is analogous to computing tangent vectors for a function(s) (belonging to objects) in a specified path (spatio-temporal volume). Optical flow can be regarded as a multivariate function $f(r, g, b, x, y)$ parameterized by the sampling interval t. The optical flow definitions with respect to the manifold definitions given above are as follows:

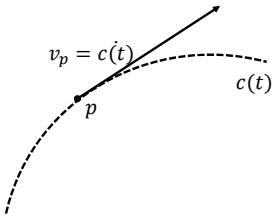

Fig. 2: A curve $c(t)$ with tangent vector at point $p = \dot{c}(t)$.

Definition 6 (Optical Flow Vector). *Let $\mathbb{R}^m$ be an m-dimensional manifold. Let p be a point in $\mathbb{R}^m$. Let $c = (x^1, x^2, \cdots, x^m)$ be a differentiable curve of class C^2, such that $c : I \to \mathbb{R}^n$ with $c(0) = p$. Then, the Optical Flow Vector is given by $v_p = \dot{c(0)} = (\dot{x^1}, \dot{x^2}, \cdots, \dot{x^m})$.*

Definition 7 (Optical Flow Tangent Space). *Let $\mathbb{R}^m$ be an m-dimensional manifold. Let p be a point in $\mathbb{R}^m$. Let c be a differentiable curve of class C^2 such that $c : I \to \mathbb{R}^n$ with $c(0) = p$. Then, the set of optical flow vectors at $p \in \mathbb{R}^m$ forms the Optical Flow Tangent Space $O_p\mathcal{M}$.*

In multi-variable analysis, the directional derivative at a point $p \in \mathbb{R}^n$ on a manifold provides a more generalized definition of the tangent vectors.

Definition 8. *Let f be a multi-variable, differentiable function defined in the neighborhood of point p. Let $c : (-\epsilon : \epsilon) \to \mathbb{R}^n$ ($\epsilon > 0$) be a differentiable curve with $c(0) = p$ and $\dot{c}(0) = v$. Then, the directional derivative Df of f in the direction of v is given by*

$$D(f \circ c) = \frac{d(f(c(t)))}{dt} \tag{3}$$

$$= Df_p(v). \tag{4}$$

Let $\mathcal{M}$ be an m-dimensional manifold. The tangent vector at a point p can be written as

$$v \in T_p\mathcal{M} = \sum_{i=1}^{m} v(x^i) \cdot (\frac{\partial}{\partial x^i}). \tag{5}$$

In case of Optical Flow Tangent Space $O_p\mathcal{M}$, we represent the function $f(r, g, b, x, y)$ in a generalized form as $f(r, g, b, x, y) \stackrel{\text{def}}{=\!=} f(x^1, x^2, x^3, x^4, x^5)$ where Einstein's summation convention is used.

Definition 9. *Let $f(x^1, x^2, x^3, x^4, x^5)$ represent the function of optical flow in the space $O_p\mathcal{M}$. The horizontal velocity v_{p_x}*

and vertical velocity v_{p_y} at $p \in \mathcal{M}$ are given by

$$v_{p_x} = f(x^1, x^2, x^3, x^4, x^5) \cdot v \left[0, 0, 0, \frac{\partial}{\partial x^4}, 0\right], \quad (6)$$

$$v_{p_y} = f(x^1, x^2, x^3, x^4, x^5) \cdot v \left[0, 0, 0, 0, \frac{\partial}{\partial x^5}\right]. \quad (7)$$

Definition 10. *An optical flow vector v_p at $p \in \mathbb{R}^m$ is an m-tuple vector with real components such that for a differentiable curve $c : I \to \mathbb{R}$, $\dot{c}(0) = v_p$.*

Theorem 1. *The optical flow space $O_p\mathcal{M}$ is a vector space.*

Proof. Let f_1 and f_2 be two optical flow functions on M. Let $p \in \mathbb{R}^m$ be a point on the optical flow space $O_p\mathcal{M}$. The functions f_1 and f_2 are nonlinear, and hence cannot be added or multiplied directly. In contrast, additional structure, such as componentwise addition and multiplication of the flow vectors at $p \in O_p\mathcal{M}$, can be achieved. Let (U, ϕ) be a coordinate chart around $p \in \mathcal{M}$. Then, $\phi \circ f_1(t)$ and $\phi \circ f_2(t)$ are curves in $\mathbb{R}^m$, where $\circ$ denotes the composition operation. Therefore, for $v_1, v_2 \in O_p\mathcal{M}$

(i). $v_1 + v_2 = \phi^{-1} \circ (\phi \circ f_1(t) + \phi \circ f_2(t))$.
(ii). $vs = \phi^{-1} \circ (r\phi \circ f_1(t))$, $v \in O_p\mathcal{M}$, $s \in \mathbb{R}$.

□

Definition 11 (Optical Flow Bundle (OFB)). *An optical flow bundle $O\mathcal{M}$ is the disjointed union of optical flow space $O_p\mathcal{M}$ such that $O\mathcal{M} : \bigsqcup_{p \in \mathcal{M}} O_p\mathcal{M}$, where $\bigsqcup$ indicates the union of optical flow tangent spaces ($O_p\mathcal{M}$).*

Definition 12 (Optical Flow Fields). *An optical flow field X on any $U \subset \mathbb{R}^m$ is the smooth assignment of an optical flow vector $v_p \in O_p\mathcal{M}$ for $f \in C^2(\mathcal{U})$ such that $Xf : \mathcal{M} \to \mathbb{R}$, with the following properties:*

(i). $X(f + g) = Xf + Xg$ *for all* $f, g \in C^\infty(M)$,
(ii). $X(fg) = fXg + gXf$ *for all* $f, g \in C^\infty(M)$,
(iii). $X(sf) = sXf$ *for all* $f \in C^\infty(M)$, $s \in \mathbb{R}$.

C. Riemannian Manifolds

Definition 13. *Let V be the vector space and V^* be the dual vector space. Then, a tensor of type (r, s) on V is a multilinear function map*

$$T : \underbrace{V^* \times \cdots \times V^*}_{r \text{ copies}} \times \underbrace{V \times \cdots \times V}_{s \text{ copies}} \to \mathbb{R} \quad (8)$$

Definition 14. *A Riemannian metric g_p at $p \in \mathbb{R}^m$ on $U \subset \mathbb{R}^m$ that varies smoothly on manifold M has the following properties:*

(i). *g is a bilinear function: (0,2)-tensor.*
(ii). *g is symmetric: for tangent vectors $X_p, Y_p \in T_p\mathcal{M}$ at point $p \in U$, $g_p(X_p, Y_p) = g_p(Y_p, X_p)$.*
(iii). *g is positive definite: $g_p(X_p, Y_p) \geq 0$ for all $X_p, Y_p \in T_pM$, $X_p \neq Y_p$, and $g_p(X_p, X_p) = 0$ iff $X_p = 0$.*

Definition 15. *A Riemannian manifold (M,g) is a smooth manifold M with a Riemannian metric g_p at $p \in \mathbb{R}^m$ on $U \subset \mathbb{R}^m$ that varies smoothly on the manifold M. The metric g can also be written as*

$$g = \sum_{i,j} g_{ij}(dx^i \otimes dx^j), \quad (9)$$

where $\otimes$ denotes the tensor product notation, vector space $V = \sum_i v^i \frac{\partial}{\partial x^i}$, set $\{\frac{\partial}{\partial x^1}\big|_p, \frac{\partial}{\partial x^2}\big|_p, \cdots, \frac{\partial}{\partial x^m}\big|_p\}$ is the basis for $T_p\mathcal{M}$ and set $\{dx^1, dx^2, \cdots, dx^m\}$ forms the dual basis to $(T_P\mathcal{M})^*$ where

$$(dx^i_p)\left(\frac{\partial}{\partial x^j}\bigg|_p\right) = \begin{cases} 1 \text{ if } i = j, \\ 0 \text{ if } i \neq j. \end{cases} \quad (10)$$

In the case of video (set of image frames) processing, if we consider only the intensity information, then the 5D parameterized space can be further parameterized as $f_i : I = \{(x(t), y(t))|x(t), y(t) = f(r(t), g(t), b(t)\}$. Therefore, the intensity space function f_i is given by $f_i : \mathbb{R}^5 \to \mathbb{R}^2$. For $\mathbb{R}^2$, assuming the standard basis vectors $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$ their dual dx and dy are independent, a constant Riemannian metric is given by $g_{ij} = dx \otimes dx + dy \otimes dy$.

Definition 16. *Let (M, g) be the Riemannian space. Let $c : [a, b] \to \mathcal{M}$ be a smooth, parameterized curve on a Riemannian manifold M, where $a, b \in \mathbb{R}$. The length of curve c is given by*

$$l(c) \stackrel{\text{def}}{=\!=} \int_a^b g_c\langle \dot{c(t)}, \dot{c(t)}\rangle dt. \quad (11)$$

Proposition 1. *Let (M, g) be the Riemannian space. The nonzero optical flow spaces indicate the presence of moving objects.*

Proof. The space $O_p\mathcal{M}$ indicates the presence of a vector field, and therefore, there must be derivatives in particular directions. Except for variations in the scene caused by noise, the majority of the vector field corresponds to the presence of an object or group of objects. This indicates the presence of moving objects. □

IV. Event Detection and Modeling

A. Events

We believe that crowd events, such as walking, running, crowd formation, splitting, local dispersion and rapid evacuation, are identified based on key human activities and movements that are normally perceived as fundamental events that we as humans perceive, and we believe these events are essential for visual surveillance. We use the keyword "event" synonymous to "activity." One of the main aims of the PETS dataset is to provide a common ground for measuring the performance of algorithms [62]. Therefore, the PETS 2009 dataset [12] is used in this body of work, and these events are defined as described below:

- **Walking** $(\mathcal{W})$ — is the *event* where objects move at a particular velocity collectively, which is less than the velocity of the events defined in running. Furthermore, *subevents* are defined, such as *standing* (W_s), *slow walking* (W_{sw})

and *fast walking* (W_{fw}) for efficient recognition and detection of events. Therefore, $\mathcal{W} = \{W_s, W_{sw}, W_{fw}\}$.

- **Running** ($\mathcal{R}$) — is the event where objects take spatio-temporal paths that are faster than those described in walking. Furthermore, *slow running* (R_{sr}) and *fast running* (R_{fr}) are defined as subevents of running. Therefore, $\mathcal{R} = \{R_{sr}, R_{fr}\}$.
- **Crowd formation** ($\mathcal{F} = \{F_f\}$) — is the event where the spatio-temporal analysis reveals that objects are converging to a single point or multiple points. Additionally, the tendencies of the objects exhibiting this phenomenon are categorized under this event.
- **Crowd splitting** ($\mathcal{S} = \{S_s\}$) — is the opposite of crowd formation. The objects in the scene would diverge from a single point or from multiple points.
- **Local dispersion** ($\mathcal{D} = \{d\}$) — is a conditional event where a walking event is recorded in association with crowd splitting.
- **Rapid dispersion** ($\mathcal{E} = \{E_e\}$) — is a conditional event where the running event is observed in conjunction with crowd splitting.

B. Walking and running events

The flow of the proposed approach is summarized in Fig. 3. Walking and running events are based on the length of the curves in the nonzero regions of the flow space defined in Riemannian space (M, g). The underlying physical phenomenon is that the length of the optical flow tangent vectors at different optical flow tangent planes $\mathcal{O}_p\mathcal{M}$ associated with walking events will have a distribution that is different from running events. Let $\mathcal{N}(\mu_1, \sigma_1^2)$ represent the lengths associated with walking events and let $\mathcal{N}(\mu_2, \sigma_2^2)$ be the distribution of the lengths associated with running events. Then, the relationship $\mu_2 \geq \mu_1$ always holds.

Proposition 2. *Let (M, g) be the Riemannian space. Let $l(c)$ be the length of the optical flow vectors in $O\mathcal{M}$ associated with optical flow spaces. Then, the walking $\mathcal{W}$ and running $\mathcal{R}$ events can be determined by the lengths of the curves of the optical flow vectors.*

Proof. The lengths of the optical flow tangent vectors are determined using optical flow tangent vectors at $O_p\mathcal{M}$ for all points $p \in \mathcal{M}$ using Definition 16. In other words, the distribution of the length of the flow vectors of the optical flow bundle provides sufficient information about the current crowd events. Events are spatio-temporal processes, and so far only spatial information has been incorporated. The temporal information is derived from tracking the tangent bundle's state consecutively corresponding to video frames.

The key determining feature of walking and running events is the length of the optical flow tangent vectors. This has been clearly shown in our previous work (please refer to Fig. 2 of [13]). The tracking of every single optical flow vector pertinent to $\mathcal{O}_p\mathcal{M}$ becomes computationally expensive and noisy. Instead, for each frame, only $v_x = max(\mathcal{OM})$ and $v_y = max(\mathcal{OM})$ are tracked, which reduces both the computational time and noise that could be introduced by the optical flow calculation. Thus, for every frame we have two scalars corresponding to two optical flow vector spaces x and y, respectively.

The above procedure can be accomplished using the following. The tangent plane to the graph of parameterized space $f(r, g, b, x, y)$ with respect to the directional derivative $\nabla\gamma_x(t)$ of x and $\nabla\gamma_y(t)$ of y directions provide tangential vectors in local coordinates in the direction of x and y, where

$$\nabla\gamma_x(t) \stackrel{\text{def}}{=\!=} \langle\gamma, \mathbf{V}_x\rangle, \tag{12}$$

and

$$\nabla\gamma_y(t) \stackrel{\text{def}}{=\!=} \langle\gamma, \mathbf{V}_y\rangle, \tag{13}$$

where $\mathbf{V}_x = [0, 0, 0, 1, 0]$ and $\mathbf{V}_y = [0, 0, 0, 0, 1]$ are the unit vectors in the directions x and y. The directional derivatives are obtained from the dense optical flow [4]. The temporal gradients of the tangent vectors with respect to x and y and for time $t = \{1, 2, \ldots, N\}$ are

$$\nabla\dot{\gamma}_x(t) = \frac{\partial^2\gamma_x}{\partial t \partial x}(t) = \mathbf{T} \times \mathbf{I} \times \nabla\gamma_x(t), \tag{14}$$

and

$$\nabla\dot{\gamma}_y(t) = \frac{\partial^2\gamma_y}{\partial t \partial y}(t) = \mathbf{T} \times \mathbf{I} \times \nabla\gamma_y(t), \tag{15}$$

where

$$\mathbf{T} \stackrel{\text{def}}{=\!=} \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ & & \ddots & \ddots & \\ 0 & 0 & 0 & 1 & -1 \end{bmatrix} \in \mathbb{R}^{n-1 \times n}, \tag{16}$$

and $\mathbf{I} \in \mathbb{R}^{n \times n}$ is an identity matrix.

The *combinational tangential temporal gradient* provides the magnitude of the velocity vector in each direction. The contribution is weighted by the coefficients such that $\sum w_{i,j} = 1$. The weighted summation of the tangential temporal gradients $(\nabla\dot{\gamma}_x(t), \nabla\dot{\gamma}_y(t))$ is calculated as

$$\nabla\dot{\gamma}_{\mathcal{A}(t)} = \frac{1}{2}\sum_{i,j} w_{ij} \times \mathbf{I}_n \times \begin{bmatrix} \nabla\dot{\gamma}_x(t) & 0 \\ 0 & \nabla\dot{\gamma}_y(t) \end{bmatrix}, \tag{17}$$

$$= \frac{1}{2}\begin{bmatrix} w_{11}\nabla\dot{\gamma}_x(t) & 0 \\ 0 & w_{22}\nabla\dot{\gamma}_y(t) \end{bmatrix}, \tag{18}$$

$$\nabla\dot{\gamma}_{\mathcal{A}} = \frac{1}{2}\sum_{t=1,2,\ldots,N} \Big(w_{11}\nabla\dot{\gamma}_x(t) + w_{22}\nabla\dot{\gamma}_y(t)\Big), \tag{19}$$

where the scalar $\nabla\dot{\gamma}_{\mathcal{A}}$ represents the mean velocity corresponding to the weighted summation of x and y directions, respectively. The function $f_A : \nabla\dot{\gamma}_{\mathcal{A}} \subset \mathbb{R} \to \mathcal{A} = \{\mathcal{W} \cup \mathcal{R}\} \subset \mathbb{R}^5$ maps the combinational tangential temporal gradient to one of the subevents such that:

$$f_{\mathcal{A}}(t) = \begin{cases} \text{Standing}, & \nabla\dot{\gamma}_{\mathcal{A}} = 0 \\ \text{Slow Walking}, & 0 < \nabla\dot{\gamma}_{\mathcal{A}} \leq a_1 \\ \text{Fast Walking}, & a_1 < \nabla\dot{\gamma}_{\mathcal{A}} \leq a_2 \\ \text{Slow Running}, & a_2 < \nabla\dot{\gamma}_{\mathcal{A}} \leq a_3 \\ \text{Fast Running}, & a_3 < \nabla\dot{\gamma}_{\mathcal{A}}, \end{cases} \tag{20}$$

where $F_{\mathcal{A}}(t) = \{f_{\mathcal{A}}(1), f_{\mathcal{A}}(2), \ldots, f_{\mathcal{A}}(N) : a_i \in \mathbb{R}\}$.

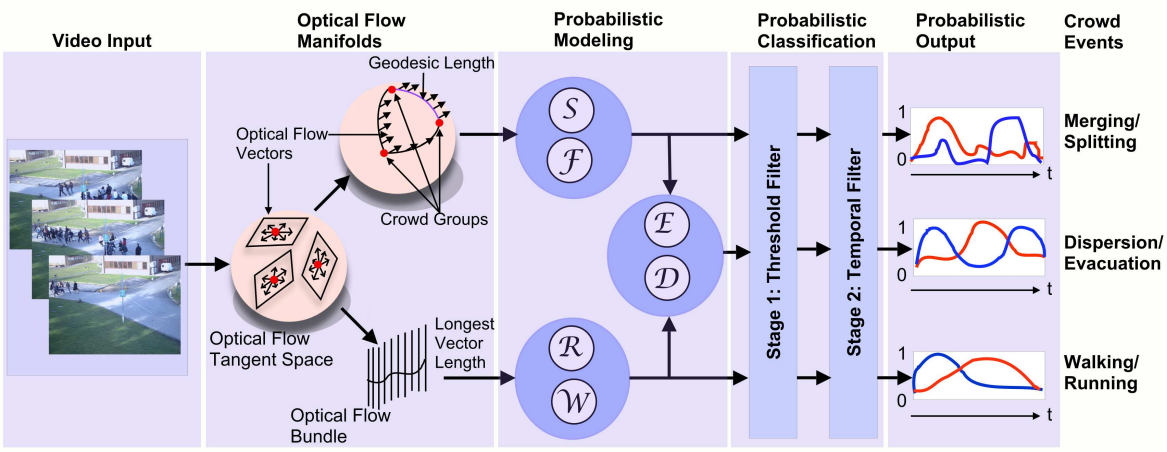

Fig. 3: Flow of the proposed crowd event detection approach. The crowd events from the input video are defined on OFM. The longest vector from the Flow Bundle is used for the detection of running $(\mathcal{R})$ and walking $(\mathcal{W})$ events. Simultaneously, groups are localized using centroids, and the direction of the group is found by parallel transporting of flow vectors. The geodesic lengths among the different groups indicate the tendency for the crowd to merge $(\mathcal{F})$ and split $(\mathcal{S})$. Dispersion $(\mathcal{D})$ and evacuation $(\mathcal{E})$ are jointly modeled from walking or running and merging or splitting events. The output at the Stage 1 of classification is based on the threshold filter (individually for event pairs). At Stage 2, the temporal filter screens the high-frequency output from Stage 1.

The probability of event $\mathcal{A}$ given time history $t = \{1, 2, \ldots, N\}$ is

$$\Pr(\mathcal{A}|t) = \begin{cases} \mathcal{W}, & \dfrac{|f_{\mathcal{A}=W_s}| + |f_{\mathcal{A}=W_{sw}}| + |f_{\mathcal{A}=W_{fw}}|}{|F_{\mathcal{A}}(t)|} \\ \mathcal{R}, & \dfrac{|f_{\mathcal{A}=R_{sr}}| + |f_{\mathcal{A}=R_{fr}}|}{|F_{\mathcal{A}}(t)|}, \end{cases} \tag{21}$$

where $|\cdot|$ is the cardinality of the set $F_{\mathcal{A}}(t)$. □

C. Merging and splitting events

The merging and splitting events are characterized by the movement of crowd groups and their intergroup distances. Let $C = \{1, 2, \cdots, N\}$ represent the current number of crowd groups. One can imagine the movement of tangent planes of a function in 5-D space. Furthermore, let each of the functions in 5-D space represents a crowd. Merging and splitting events are relative events in the sense that one group is moving away from the other, but the same group may be approaching another group. In this work, because the goal is to seek global information about the crowd, we report an overall tendency of groups to merge or split.

Initially, groups in the crowd are identified using the nonzero optical flow vectors in the OFB. The connectivity of the nonzero tangent vectors in the neighborhood extends in all directions until the tangent vectors are zero, which creates contour-like boundaries around the groups. The center of mass of each group is located using a centroid. Using the Riemannian connection on the OFM, the tangent vectors at different points in the tangent bundle corresponding to that particular group are parallel transported to the centroid location. This parallel transport allows the optical flow tangent vectors to be moved from one tangent plane to another without affecting the properties of the vectors. The resultant vector at the centroid location provides the principal direction of the group and its velocity, which provides localized information about the group and its tendency.

The global group tendency is then detected using all of the groups in a given frame. The distance between groups can be measured using the geodesic distance. The geodesic distance between the points (assuming the curves are *admissible*) is given by

$$\mathcal{L}_a^b(\gamma)_{ij} \stackrel{\text{def}}{=\!=} \int_a^b g_{\gamma_{ij}} \langle \dot{\gamma}_i(t), \dot{\gamma}_j(t) \rangle \tag{22}$$

Proposition 3. *Let (M, g) be the Riemannian space. Let $l(c)$ be the length of the optical flow vectors in OM associated with optical flow spaces. Let $\mathcal{L}_a^b(\gamma)_{ij}$ be the geodesic distance between two tangent points. Then, the global events merging F and splitting S can be determined by the geodesic lengths of the curves of optical flow vectors.*

Proof. The geodesic distance matrix provides the geodesic distance between groups. Thus, the temporal evolution of group locations can be measured by tracking the variational changes in the positions of the tangent vectors provided by the geodesic distance matrix (**G**).

$$\mathbf{G}(t) = \dot{\boldsymbol{\gamma}}(t)^T \dot{\boldsymbol{\gamma}}(t) \tag{23}$$

$$= \begin{bmatrix} \mathcal{L}(\gamma)_{11} & \mathcal{L}(\gamma)_{12} & \cdots & \mathcal{L}(\gamma)_{1N} \\ \mathcal{L}(\gamma)_{21} & \mathcal{L}(\gamma)_{22} & \cdots & \mathcal{L}(\gamma)_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ \mathcal{L}(\gamma)_{N1} & \mathcal{L}(\gamma)_{N2} & \cdots & \mathcal{L}(\gamma)_{NN}. \end{bmatrix} \tag{24}$$

For each of the groups in the crowd, the mean relative probability of merging or splitting is given by $\sum_i \mathcal{L}(\gamma)_{Ni}$ or $\sum_i \mathcal{L}(\gamma)_{iN}$ because **G** is symmetric. The overall tendency of the crowd at any given instance t is $\sum_{i,j} \mathcal{L}(\gamma)$. Then, the probability of an event B given the temporal variations

$t = \{1, 2, \ldots, N\}$ is $f_{\mathcal{B}}(t)$. The function $f_{\mathcal{B}}(t) : \mathbf{G}(t) \subset \mathbb{R} \rightarrow \mathcal{B} = \{\mathcal{F} \cup \mathcal{S}\} \subset \mathbb{R}$ maps the *variations of geodesic distances* to one of the subevents such that

$$f_{\mathcal{B}}(t) = \begin{cases} \text{Splitting}, & \sum_t \mathbf{G}(t) > 0 \\ \text{Merging}, & \sum_t \mathbf{G}(t) < 0 \\ \text{Neither}, & \sum_t \mathbf{G}(t) = 0, \end{cases} \tag{25}$$

where $F_{\mathcal{B}}(t) = \{f_{\mathcal{B}}(1), f_{\mathcal{B}}(2), \ldots, f_{\mathcal{B}}(N) : f_{\mathcal{B}}(t) \in \mathbb{R}^2\}$. The probability of event $\mathcal{B}$ given time history $t = \{1, 2, \ldots, N\}$ is

$$\Pr(\mathcal{B}|t) = \begin{cases} \mathcal{F}, & \frac{|f_{\mathcal{B}=F_f}|}{|F_{\mathcal{B}}(t)|} \\ \mathcal{S}, & \frac{|f_{\mathcal{B}=S_s}|}{|F_{\mathcal{B}}(t)|}, \end{cases} \tag{26}$$

where $|\cdot|$ is the cardinality of the set $F_{\mathcal{B}}(t)$ and

$$\Pr(\mathcal{B} = \mathcal{F}) + \Pr(\mathcal{B} = \mathcal{S}) \leq 1 \tag{27}$$

A positive variational change between two tangent planes infers the possibility of a splitting event and a negative variational indicates the possibility of merging. A combination of these possibilities leads to the overall crowd formation or splitting events. □

D. Dispersion and evacuation events

The local dispersion ($\mathcal{D}$) and evacuation ($\mathcal{E}$) events are derived from the joint probability distribution of events $\mathcal{A}$ and $\mathcal{B}$. The probability that the event $\mathcal{C}$ is a local dispersion is given by

$$\Pr(\mathcal{C} = \mathcal{D}|t) = \Pr(\mathcal{A} = \mathcal{W}|t) \cdot \Pr(\mathcal{B} = \mathcal{S}|t), \tag{28}$$

and the probability that the event $\mathcal{C}$ is an evacuation event is given by

$$\Pr(\mathcal{C} = \mathcal{E}|t) = \Pr(\mathcal{A} = \mathcal{R}|t) \cdot \Pr(\mathcal{B} = \mathcal{S}|t) \tag{29}$$

V. Evaluation

The proposed method was implemented in OpenCV 2.4 on a Virtual Box Linux machine (64-bit Ubuntu 14.04 LTS) equipped with 1.5 GB RAM and Intel® i7−2600 CPU running at 3.4 GHz.

A. Dataset

The PETS 2009 [12] dataset was used to evaluate the proposed method. The crowd events are categorized under Dataset S3 with four different timings ($14-16, 14-27, 14-31$ and $14-33$,) and for each timing, there are four different views ($001, 002, 003,$ and 004). The timing, for example, $14-16$ denotes the time (in the format hh-mm, where "hh" means hour and "mm" means minutes) when the data were collected. To the best of our knowledge, only the PETS 2009 [12] dataset has events where all six crowd events can be clearly evaluated based on human analysis. Therefore, the proposed approach was evaluated on a total of 16 different sequences. All of the sequences were manually annotated into three event groups. The preprocessing of the video frames is based on [63].

B. Implementation Details

We used a binary classifier to classify the events based on the probability density function generated by the events as described in Section IV.

Walking and running events: The parameters w_{11} and w_{22} in (19) were set to 0.5. For an event to be considered running, the probability of the function in (20) is considered to be greater than T_1 i.e.,

$$\Pr(\mathcal{A}|t) = \begin{cases} \mathcal{W}, & \leq T_1 \\ \mathcal{R}, & > T_1. \end{cases} \tag{30}$$

Merging and splitting events: In the case of merging and splitting, we found significant overlap between $\Pr(\mathcal{F})$ and $\Pr(\text{Neither})$. Therefore, to classify the events accurately, we used

$$f_{\mathcal{B}}(t) = \begin{cases} \text{Splitting}, & \sum_t \mathbf{G}(t) > 0 \\ \text{Merging}, & \sum_t \mathbf{G}(t) \leq 0. \end{cases} \tag{31}$$

and

$$\Pr(\mathcal{B}|t) = \begin{cases} \mathcal{F}, & > T_1 \wedge \sum_t \mathbf{G}(t) \leq 0 \\ \mathcal{S}, & > T_1 \wedge \sum_t \mathbf{G}(t) > 0. \end{cases} \tag{32}$$

Dispersion and evacuation events: The results for dispersion and evacuation events are jointly modeled, and the classification of events is given by

$$\Pr(\mathcal{C}|t) = \begin{cases} \mathcal{E}, & > T_2 \\ \mathcal{D}, & \leq T_2. \end{cases} \tag{33}$$

C. Calculating Parameters and Thresholds

The temporal parameter t was determined using the training dataset. The t was selected such that the least squares classification error was minimized. In the PETS2009 dataset, this was found to be 5 in all of the experiments. The temporal filter at Stage 2 uses a convolution operation to smooth the transient signals from Stage 1. The convolution operation uses a 1D Gaussian kernel with kernel size equals $t = 5$. The output from the Stage 1 to Stage 2 are scalar values. Let S_1 denote the 1D signal from Stage 1, S_2 at Stage 2, and G denote the Gaussian kernel. The temporal filtering operation at Stage 2 can be written as

$$S_2(i) = \sum_{j=1}^{n} G(j) \cdot S_1(i - j + n/2), \tag{34}$$

where i and j are the indices used to perform convolution, and $n \in [1, t]$.

The parameters ($a_1, a_2,$ and a_3) in (20) were determined by modeling Mixture of Gaussians (MoGs). The initial values for parameters a_i were determined by using the K-means approach [64] with $K = 3$. These parameters are specific to camera views and the dataset. K-means uses least-squares error to partition the training data into K clusters and determine the K centroids. Later, MoGs were modeled using with Gaussian means equal to K means.

The event pairs (walking-running, dispersion-evacuation) are modelled using a mixture of two probability density functions (PDFs). Let us consider the walking and running event. The procedure outlined in [65] is followed. Let $p(a) = P_1 p_1(a) + P_2 p_2(a)$, where P_1 and P_2 correspond to $\mathcal{W}$ and $\mathcal{R}$ events, and $p_1 a$ and $p_2(a)$ are the corresponding PDFs, respectively. The PDFs p_1 and p_2 are assumed to be Gaussian i.e.,

$$p(a) = \frac{P_1}{\sqrt{2\pi\sigma_1}} e^{-\frac{a-\mu_1}{2\sigma^2}} + \frac{P_2}{\sqrt{2\pi\sigma_2}} e^{-\frac{a-\mu_2}{2\sigma^2}}. \quad (35)$$

Then, the probability of error in classifying $\mathcal{W}$ and $\mathcal{R}$ are

$$E_1(T_1) = \int_0^{T_1} p_2(a)da, \qquad E_2(T_1) = \int_{T_1}^{\infty} p_1(a)da \quad (36)$$

Then, the total error probability is

$$E(T) = P_2 E_1(T_1) + P_1 E_2(T_1). \quad (37)$$

The minimal error is found by differentiating (37) and equating it to zero, which results in

$$P_1 p_1(T_1) = P_2 p_2(T_1). \quad (38)$$

The analytical solution to (38) is given by

$$(\sigma_1^2 - \sigma_2^2)(T_1) + 2(\mu_1\sigma_2^2 - \mu_2\sigma_2^2)T_1 + (\sigma_1^2\mu_2^2 - \sigma_2^2\mu_1^2 + 2\sigma_1^2\sigma_2^2 \ln(\frac{\sigma_2 P_1}{\sigma_1 P_2})) \quad (39)$$

Assuming equal variances for PDFs, i.e., $\sigma_1 = \sigma_2 = \sigma$, the threshold T_1 is given by

$$T_1 = \frac{\mu_1 - \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \ln(\frac{P_2}{P_1}) \quad (40)$$

The value T_1 in (40) is used in (30) and (32). Similarly, T_2 is found for the dispersion-evacuation event and is used in (33).

D. Results and Discussion

The results of the proposed crowd event detection approach are discussed at three different levels. First, the events are fundamentally pairwise: walking-running, merging-splitting and dispersion-evacuation. Stage 1 and Stage 2 confusion matrices for all three pairwise crowd events [13] are provided in Tables I and II. From Table I(a) the walking events were detected as walking 76% of the time. In contrast, running events were detected as walking 37% of the time and as running 63% of the time. Merging events were correctly detected 88% of the time and splitting 60% of the time, as shown in Table I(b). The highest correct detection rate (94%) was achieved in detecting dispersion events and, 65% correct detection was achieved for evacuation. If we consider an actual event of the frame i to be x_i and the detected event to be y_i, then the error in detection for the frame i will be either $T_i = 1$ if the event detection is correct or else $T_i = 0$. Consequently, the percentage error (p_{error}) accumulated over the model delay during detection will be $p_{error} = \sum_1^t \frac{(|T_i|)}{t} \times 100$. This result is related to the large error rates in the confusion matrices. The threshold parameters for running and walking vary from method to method, for example, Benabbas *et al.* [14] chose 0.95 for running based on Gaussian model and Gárate *et al.* [54] specified t_1 and t_2 based on motion vectors. The confusion matrices (Table II) at Stage 2 indicate that walking events were correctly classified 91% of the times, whereas running events were detected with 84% accuracy. Classification of merging events equalled walking (91%) and splitting events were incorrectly classified as merging at an average of 7%. Dispersion events were efficiently classified (94%) as opposed to evacuation events (86%).

TABLE I: Confusion matrices for the crowd events detected [13] using the PETS 2009 dataset [12] at Stage 1: (a) walking and running events, (b) merging and splitting events, (c) dispersion and evacuation events.

(a) Confusion matrix for walking and running events

	$\mathcal{W}$	$\mathcal{R}$
$\mathcal{W}$	0.76	0.24
$\mathcal{R}$	0.37	0.63

(b) Confusion matrix for merging and splitting events

	$\mathcal{F}$	$\mathcal{S}$
$\mathcal{F}$	0.88	0.12
$\mathcal{S}$	0.4	0.60

(c) Confusion matrix for local dispersion and evacuation events

	$\mathcal{D}$	$\mathcal{E}$
$\mathcal{D}$	0.94	0.06
$\mathcal{E}$	0.35	0.65

TABLE II: Confusion matrices for the crowd events detected using the PETS 2009 dataset [12] at Stage 2: (a) walking and running events, (b) merging and splitting events, (c) dispersion and evacuation events.

(a) Confusion matrix for walking and running events

	$\mathcal{W}$	$\mathcal{R}$
$\mathcal{W}$	0.91	0.09
$\mathcal{R}$	0.16	0.84

(b) Confusion matrix for merging and splitting events

	$\mathcal{F}$	$\mathcal{S}$
$\mathcal{F}$	0.91	0.08
$\mathcal{S}$	0.07	0.93

(c) Confusion matrix for local dispersion and evacuation events

	$\mathcal{D}$	$\mathcal{E}$
$\mathcal{D}$	0.94	0.06
$\mathcal{E}$	0.14	0.86

TABLE III: Comparison of the detection of the start and end timings (in seconds, fps=7) of crowd events with the ground truth from the selected video samples [63]. This table highlights the model delay in detecting particular events. The maximum delay was observed to be 4 seconds.

Video	**Event**	**Ground Truth** Start-End (sec)	**Detected** Start-End (sec)
14-16, View-001	Walking	0-6 13-24	0-7 17-28
14-16, View-001	Running	6-15 24-31	7-17 28-31
14-33, View-001	Merging	0-29	0-27
14-33, View-001	Splitting	48-53	49-53
14-33, View-001	Dispersion	0-48	0-49
14-33, View-001	Evacuation	48-53	49-53

At the second level, the results are reported in terms of event detection as a time series. The results in Table III provide a comparison of the detection of the start and end timings of the crowd events. Fig. 4 shows the corresponding temporal output for walking and running event. Here, we showed the output for View-001 of the PETS 2009 dataset. We conducted experimental evaluations of event detection from different views and found that View-001 best captures the crowd events. The same events result in different events when viewed from

TABLE IV: Comparison of crowd event detection results. The last two columns of the table indicate the results of the proposed approach at Stage 1 and Stage 2 respectively. The bolded text indicates where the proposed approach has better performance.

Crowd Event	Measure	Statistical Filters [66]	Holistic Approach [67]	Random Forest [14]	Motion Pattern [14]	Stage 1 Results	Stage 2 Results
Walking	Precision	-	0.87	0.96	0.97	0.61	0.82
	Recall	-	-	0.99	0.96	0.75	0.97
	F-score	-	-	0.97	0.96	0.67	0.88
Running	Precision	0.99	0.75	0.86	0.75	0.78	0.93
	Recall	0.99	-	0.68	0.81	0.63	0.84
	F-score	0.99	-	0.75	0.77	0.69	0.89
Merging	Precision	-	0.68	0.65	0.59	**0.85**	**0.92**
	Recall	-	-	0.46	0.45	**0.88**	**0.89**
	F-score	-	-	0.53	0.51	**0.86**	**0.9**
Splitting	Precision	0.65	0.74	0.73	0.47	0.66	**0.93**
	Recall	1	-	0.92	0.47	0.6	**0.95**
	F-score	0.78	-	0.81	0.47	0.62	**0.94**
Dispersion	Precision	-	0.8	0.58	0.67	**0.9**	**0.94**
	Recall	-	-	0.48	0.45	**0.94**	**0.98**
	f-score	-	-	0.52	0.53	**0.91**	**0.96**
Evacuation	Precision	-	0.94	0.83	0.69	0.75	0.85
	Recall	-	-	1.0	0.82	0.65	0.84
	F-score	-	-	0.90	0.74	0.69	0.85

different views. In the proposed method, empirically chosen $t = 5$ was used for crowd event detection, and is the main contributor to accurate detection as well as detection delay. The detection delay is the delay incurred by the model (time-window) and not the computation delay, which has not been reported in the literature. From Table III, we observe that there is a maximum delay of 4 seconds between the actual start of an event and correct detection, which is same for all cases across different camera views (View-001—View-004). The start of an event may be slightly delayed due to camera views and occlusion.

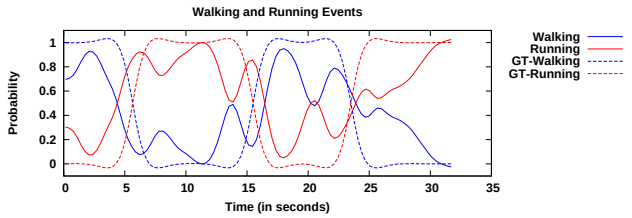

Fig. 4: A sample probabilistic output obtained for walking–running event (dataset: PETS 2009, 14-16, View-001) along with ground truth (GT) at Stage 2.

At the third level, a detailed comparison with different methods is provided in Table IV. The comparison is based on View-001 of our approach. In [14], the test was conducted on 1000 frames. Therefore, the events have been divided into two groups. The first class described to be either running or walking. The second class contained the remaining events. In this study, we separated the second division, resulting in three categories for six events. The rationale behind this approach is that the merging and splitting events can be separated from local dispersion and evacuation events as described earlier. The results are provided in two stages. As shown in Table IV, at Stage 1, the probabilities are built using the definitions of the sub-events and events. The events were then classified based on the thresholds determined using the mixture of probability density functions at every time instance. In Stage 2, a temporal filter with window size corresponding to delay in processing ($t = 5$ seconds) was added to refine the results. This eliminated the transient probability outputs and allowed the smooth transition of events. The justification for the addition of this filter is that the abrupt movements in the scene due to human movements cause the length of the flow vectors to overshadow actual events. Further, we observed that in the PETS 2009 dataset [12], the crowd movements abruptly changed. For the results at Stage 1, dispersion (precision: 0.9, recall: 0.94 and $F-$score: 0.91) and merging (precision: 0.85, recall: 0.88 and $F-$score: 0.86) events have the highest accuracies.

Table II shows the confusion matrices for classification at stage 2. Clearly, it surpasses the Stage 1 results. As shown in Table IV, it can be seen that merging events (precision: 0.92, recall: 0.89 and $F-$score: 0.9) have performed better than others. Similarly, the results of a splitting event (precision: 0.93, recall: 0.95 and $F-$score: 0.94) and dispersion events (precision: 0.94, recall: 0.98 and $F-$score: 0.96) are better than others. The remaining events, i.e., the walking (precision: 0.82, recall: 0.97 and $F-$score: 0.88), running (precision: 0.93, recall: 0.84 and $F-$score: 0.89) and evacuation events (precision: 0.85, recall: 0.84 and $F-$score: 0.85) are comparable to others.

In [66], the high running event classification measure (precision: 0.99, recall: 0.99, $F-$score: 0.99) is attributed to the problem formulation. The classification was formulated between running and splitting instead of walking and running. However, in the proposed and remaining approaches, the classification is between walking and running. In [67], crowd events were classified using Dynamic Texture (DT) features along with Nearest Neighbor and SVM classifier. The classification threshold was set to 0.5 and 75% of the dataset was used for training. In the proposed approach, the training dataset (with respect to a particular view) is used for determination of temporal parameter t. In [14], two classifiers were used: (1) walking/running events, and (2) merging, splitting, dispersion and evacuation events. For detection and classification of re-

sults, about 5 different parameters are required. Comparatively, the proposed approach requires temporal parameter t is to be determined.

The parameter t is empirically selected. In [14], an equivalent of t was set to 4, whereas we have selected as $t = 5$. A smaller t results in less number of data points to build the probability distribution in which case the results will be instantaneous, i.e., the output could fluctuate arbitrarily. On the other hand, a larger t result in influencing the probability distribution of the large proportion of elapsed events. As a rule of thumb in finding the t, one can count the number of frames an event occurs and calculate time using the frame rate. This rule can be applied for finding shorter events if required. For longer events, the effect of influence of t is relatively less compared with shorter events. In case of longer events, a smaller t would yield accurate results, whereas a larger t would delay the event detection. In case of shorter event following a longer event, smaller t is preferable to avoid delayed detection of smaller event, which otherwise would result in high classification errors. In general, keeping t smaller is the best way to avoid detection and eventually the classification errors.

The chief goal of designing an automated event detection system by reducing human intervention is achieved using the proposed method. From a video surveillance perspective, merging and dispersion events are more important for behavioral analysis than walking and running events, which are usually dependent on multiple factors. For example, in the event of crowd panic in response to possible injury or threat to human life at a stadium, the proposed probabilistic model indicates the merging and dispersion (indicators of panic) immediately, which is an indispensable model compared with existing methods. We separated the merging/splitting events from local dispersion/evacuation events to facilitate the detection of exact events in video surveillance applications. Further improvement was made by combining the regular event with local dispersion, because we found a significant overlap between them.

One of the potential reasons for low detection rates is that during occlusion, the tangent vectors estimated are indistinguishable for walking and running events. The result can be ameliorated with the utilization of group tracking techniques to estimate the group velocity. Likewise, if the tracking algorithms are lightweight and sufficiently fast, region-based optical flow can be implemented to improve running and walking events. The proposed approach performs better than the existing methods in merging, splitting and dispersion because of the inclusion of localized group detection using the Riemannian connection in the OFB, which is one of the novel aspects of the proposed approach. The parallel transport of flow vectors provides us with a method to transport the vectors from one tangent space to another. In this way, the localization of the crowd group and its direction is invariant to the location of the center of the group mass, which is the second novel aspect of the proposed approach. Some approaches use crowd movement direction vectors that are inadequate at many times because of the inability to capture the localization features, and thus, there is the possibility of incorrect detection. This result may not occur in all instances, but it can never be ruled out at critical junctures.

In this work, the PETS 2009 dataset [12] was used and the threshold parameters were selected as described in Section V-C using the dataset (14-16, View-001) for all of the crowd events. There are datasets, such as [68], where only abnormal events are present. The crowd events are only available in the PETS 2009 dataset [12]. Two things should be noted here: (1) in this work, we used only one camera view (View-001, 14-16)—because the literature in crowd monitoring adopts View-001 as an optimal camera position for visual surveillance. The results for the other views and timings were obtained using the same model parameters. Therefore, we call the proposed method to be semi-supervised; (2) when we view an event from different cameras, the features used will have an impact on the detection of events. For example, splitting an event along the x–plane may not appear to be splitting at all from another view. Likewise, the features for the other events will change.

In [69], [15], optical flow features were used for anomaly detection using histograms of optical flow. However, the optical flow values vanish for a static crowd in the scene. We used the GMM [70] for background modeling followed by optical flow for crowd detection. Future work in this direction includes derivation of efficient velocity vectors in crowded scenes without tracking in Riemannian manifolds. A further improvement in processing and feature space can be introduced with the help of manifold learning while detecting the events. Nevertheless, OFM can still be utilized for probabilistic estimation of crowd events in almost real-time.

Several optical flow based methods have tried to address this issue by assuming that the vectors will be inconsistent or undefined. Methods, such as [71], [72], [73], have tried to address the occlusion problem. These methods have been proposed based on the assumption that occluded pixels will be visible in the next frame. The occlusion involving humans is different compared with objects, such as, cars that are rigid with a uniform motion. Unlike these, crowd motion includes body parts moving in different directions and at different velocities. Secondly, because the velocity of a rigid object is constant most of the times, the flow vectors can be calculated even if the occluded pixels are not observed using the consecutive frame. However, in case of humans this is not possible because of nonuniform body movements. During occlusions, there is no defined pattern of vectors that could be used for classifying the subevents. Therefore, it is not possible exactly to determine the occlusion and its effects in terms of subevents.

The walking and running events directly use the probability distribution of the length of tangent vectors. These vectors are inconsistent at the boundaries of the occlusions. Therefore, the events will be misrepresented. However, for events like merging and splitting, the geodesic distance between tangent planes and the crowd direction (derived by using Riemannian connections and parallel transport) are used. The geodesic distances, estimate the distances between patches (groups of people), which is not affected by the occlusion. Similarly, the crowd direction estimates the principal direction of the crowd, where vector directions are used as opposed to the length.

This will not be affected by occlusion because when there are occlusions in a group of people, the majority of the vectors will be pointing in the crowd direction.

In this work, we assume that only humans are present. The PETS 2009 dataset [12] contains only humans and it is the only dataset where crowd events have been procedurally acquired. The classification of people, animals, vehicles, etc. has not been included. Future work will involve incorporating this aspect. Furthermore, when the crowd approaches the camera, the length of the tangent vectors increase, affecting the performance. To an extent, this limitation is overcome by the threshold selection (as described in Section V-C) technique, and the inclusion of the temporal filter in the Stage 2, but an automated view normalization technique would be an ideal solution. In addition, automated crowd event detection with adaptive learning has potential in video surveillance applications.

VI. Conclusion

Crowd event detection and classification is key to understanding behavioral characteristics of a crowd. In this regard, we developed a probabilistic detection of crowd events (running, walking, merging, splitting, local dispersion and evacuation) on OFM using Riemannian manifolds. A motion-based, probabilistic framework for detection of crowd events has been proposed. In particular, a new algorithm to detect walking and running events has been reported, which uses optical flow vector lengths in OFM. Additionally, the framework delivers a system to detect merging and splitting events, which uses Riemannian connections in the OFB. The algorithm resulted in excellent performance in the detection of all events and outperformed other algorithms in merging, splitting and dispersion.

References

[1] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 34, no. 3, pp. 334–352, 2004.

[2] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Computing Surveys*, vol. 38, no. 4, Dec. 2006.

[3] L. Li, W. Huang, I. Y. H. Gu, L. Ruijiang, and Q. Tian, "An efficient sequential approach to tracking multiple objects through crowds for real-time intelligent cctv systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 38, no. 5, pp. 1254–1269, 2008.

[4] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[5] S. Wu and H. S. Wong, "Crowd motion partitioning in a scattered motion field," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 42, no. 5, pp. 1443–1454, 2012.

[6] M. Thida, H.-L. Eng, D. N. Monekosso, and P. Remagnino, "Learning video manifolds for content analysis of crowded scenes," *IPSJ Transactions on Computer Vision and Applications*, vol. 4, pp. 71–77, 2012.

[7] Y. Yuan, J. Fang, and Q. Wang, "Online anomaly detection in crowd scenes via structure analysis," *IEEE Transactions on Cybernetics*, vol. 45, no. 3, pp. 562–575, 2015.

[8] K. Anderson and P. W. McOwan, "A real-time automated system for the recognition of human facial expressions," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 36, no. 1, pp. 96–105, 2006.

[9] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[10] A. McInerney, *First Steps in Differential Geometry: Riemannian, Contact, Symplectic*. Springer New York, 2013.

[11] S. Nagaraj, C. Hegde, A. Sankaranarayanan, and R. Baraniuk, "Optical flow-based transport on image manifolds," *Applied and Computational Harmonic Analysis*, vol. 36, no. 2, pp. 280–301, 2014.

[12] J. Ferryman, "PETS 2009 benchmark data," http://www.cvg.reading.ac.uk/PETS2009/a.html, 2009, [Online; verified on 28-June-2015].

[13] A. S. Rao, J. Gubbi, S. Marusic, and M. Palaniswami, "Probabilistic detection of crowd events on riemannian manifolds," in *2014 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, 2014, pp. 1–8.

[14] Y. Benabbas, N. Ihaddadene, and C. Djeraba, "Motion pattern extraction and event detection for automatic visual surveillance," *Journal on Image and Video Processing*, vol. 2011, pp. 1–15, 2011.

[15] M. Thida, E. How-Lung, and P. Remagnino, "Laplacian eigenmap with temporal constraints for local abnormality detection in crowded scenes," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 2147–2156, 2013.

[16] A. A. Chaaraoui, P. Climent-Pérez, and F. Flórez-Revuelta, "A review on vision techniques applied to human behaviour analysis for ambient-assisted living," *Expert Systems with Applications*, vol. 39, no. 12, pp. 10 873–10 888, 2012.

[17] J. Bergen, P. Anandan, K. Hanna, and R. Hingorani, *Hierarchical model-based motion estimation*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 1992, vol. 588, book section 27, pp. 237–252.

[18] E. S. M. I. Lena Gorelick, Moshe Blank and R. Basri, "Actions as space-time shapes," http://www.wisdom.weizmann.ac.il/~vision/SpaceTimeActions.html, 2007, [Online; verified on 28-June-2015].

[19] I. Laptev and B. Caputo, "Recognition of human actions," http://www.nada.kth.se/cvap/actions/, 2005, [Online; verified on 28-June-2015].

[20] S. Ali and M. Shah, "Human action recognition in videos using kinematic features and multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 2, pp. 288–303, 2010.

[21] A. Gilbert, J. Illingworth, and R. Bowden, "Action recognition using mined hierarchical compound features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 883–897, 2011.

[22] K. Huang, Y. Zhang, and T. Tan, "A discriminative model of motion and cross ratio for view-invariant action recognition," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2187–2197, 2012.

[23] A. Haq, I. Gondal, and M. Murshed, "On temporal order invariance for view-invariant action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 2, pp. 203–211, 2013.

[24] G. Yu, J. Yuan, and Z. Liu, "Action search by example using randomized visual vocabularies," *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 377–390, 2013.

[25] Q. Wu, Z. Wang, F. Deng, Z. Chi, and D. D. Feng, "Realistic human action recognition with multimodal feature selection and fusion," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 4, pp. 875–885, 2013.

[26] I. Everts, J. C. van Gemert, and T. Gevers, "Evaluation of color spatio-temporal interest points for human action recognition," *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1569–1580, 2014.

[27] D. Wu and L. Shao, "Silhouette analysis-based action recognition via exploiting human poses," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 2, pp. 236–243, 2013.

[28] X. Wu, D. Xu, L. Duan, J. Luo, and Y. Jia, "Action recognition using multilevel features and latent structural svm," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 8, pp. 1422–1431, 2013.

[29] Z. Zhang and D. Tao, "Slow feature analysis for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 436–450, 2012.

[30] Y. M. Lui, "Tangent bundles on special manifolds for action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 6, pp. 930–942, 2012.

[31] A. J. Ma, P. C. Yuen, W. W. W. Zou, and L. Jian-Huang, "Supervised spatio-temporal neighborhood topology learning for action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 8, pp. 1447–1460, 2013.

[32] O. P. Popoola and W. Kejun, "Video-based abnormal human behavior recognition—a review," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 42, no. 6, pp. 865–878, 2012.

[33] D. Zhang, D. Gatica-Perez, S. Bengio, and I. McCowan, "Semi-supervised adapted hmms for unusual event detection," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1. IEEE, 2005, pp. 611–618.

[34] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, "Robust real-time unusual event detection using multiple fixed-location monitors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 3, pp. 555–560, 2008.

[35] F. Jiang, W. Ying, and A. K. Katsaggelos, "A dynamic hierarchical clustering method for trajectory-based unusual video event detection," *IEEE Transactions on Image Processing*, vol. 18, no. 4, pp. 907–913, 2009.

[36] B. Zhao, L. Fei-Fei, and E. P. Xing, "Online detection of unusual events in videos via dynamic sparse coding," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011, pp. 3313–3320.

[37] V. Mahadevan, L. Weixin, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 1975–1981.

[38] E. L. Andrade, S. Blunsden, and R. B. Fisher, "Modelling crowd scenes for event detection," in *18th International Conference on Pattern Recognition (ICPR 2006)*, vol. 1. IEEE, 2006, pp. 175–178.

[39] R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009, pp. 935–942.

[40] M. Andersson, F. Gustafsson, L. St-Laurent, and D. Prevost, "Recognition of anomalous motion patterns in urban surveillance," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 1, pp. 102–110, 2013.

[41] D.-Y. Chen and P.-C. Huang, "Dynamic human crowd modeling and its application to anomalous events detcetion," in *2010 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2010, pp. 1582–1587.

[42] I. Tziakos, A. Cavallaro, and L.-Q. Xu, "Event monitoring via local motion abnormality detection in non-linear subspace," *Neurocomputing*, vol. 73, no. 1012, pp. 1881–1891, 2010.

[43] H. Liao, J. Xiang, W. Sun, Q. Feng, and J. Dai, "An abnormal event recognition in crowd scene," in *2011 Sixth International Conference on Image and Graphics (ICIG)*. IEEE, 2011, pp. 731–736.

[44] Y. Cong, J. Yuan, and Y. Tang, "Video anomaly search in crowded scenes via spatio-temporal motion context," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 10, pp. 1590–1599, 2013.

[45] J. Shen, D. Tao, and X. Li, "Modality mixture projections for semantic video event detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1587–1596, 2008.

[46] N. Ben Aoun, H. Elghazel, and C. Ben Amar, "Graph modeling based video event detection," in *2011 International Conference on Innovations in Information Technology (IIT)*. IEEE, 2011, pp. 114–117.

[47] G. Cámara-Chávez and A. de Albuquerque Araújo, "Harris-sift descriptor for video event detection based on a machine learning approach," in *11th IEEE International Symposium on Multimedia (ISM)*. IEEE, 2009, pp. 153–158.

[48] S. Kwak, B. Han, and J. Han, "On-line video event detection by constraint flow," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2013.

[49] X. Qian, X. Hou, Y. Y. Tang, H. Wang, and Z. Li, "Hidden conditional random field-based soccer video events detection," *IET Image Processing*, vol. 6, no. 9, pp. 1338–1347, 2012.

[50] Y. Ke, R. Sukthankar, and M. Hebert, "Volumetric features for video event detection," *Interantional Journal of Computer Vision*, vol. 88, no. 3, pp. 339–362, 2010.

[51] D. Tran, J. Yuan, and D. Forsyth, "Video event detection: From subvolume localization to spatiotemporal path search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 2, pp. 404–416, 2014.

[52] Y. Chen, Z. Zhong, L. Ka Keung, and X. Yangsheng, "Multi-agent based surveillance," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 2810–2815.

[53] R. Li, R. Chellappa, and S. K. Zhou, "Learning multi-modal densities on discriminative temporal interaction manifold for group activity recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009, pp. 2450–2457.

[54] C. Gárate, P. Bilinsky, and F. Bremond, "Crowd event recognition using hog tracker," in *2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-Winter)*. IEEE, 2009, pp. 1–6.

[55] G. Li, J. Chen, B. Sun, and H. Liang, "Crowd event detection based on motion vector intersection points," in *2012 International Conference on Computer Science and Information Processing (CSIP)*. IEEE, 2012, pp. 411–415.

[56] J. J. A. Lee and M. Verleysen, *Nonlinear dimensionality reduction*. Springer, 2007.

[57] J. M. Lee, *Riemannian Manifolds: An Introduction to Curvature*. Springer, 1997, vol. 176.

[58] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th international joint conference on Artificial intelligence*. Morgan Kaufmann Publishers Inc., 1981, pp. 674–679.

[59] G. Farnebäck, *Two-Frame Motion Estimation Based on Polynomial Expansion*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2003, vol. 2749, ch. 50, pp. 363–370.

[60] C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime tv-l^1 optical flow," in *Pattern Recognition*, ser. Lecture Notes in Computer Science, F. A. Hamprecht, C. Schnörr, and B. Jähne, Eds., vol. 4713. Springer Berlin Heidelberg, 2007, pp. 214–223.

[61] M. Tao, J. Bai, P. Kohli, and S. Paris, "Simpleflow: A non-iterative, sublinear optical flow algorithm," *Computer Graphics Forum*, vol. 31, no. 2, pp. 345–353, 2012.

[62] J. Ferryman and A. Shahrokni, "Pets2009: Dataset and challenge," in *2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-Winter)*. IEEE, 2009, pp. 1–6.

[63] A. S. Rao, J. Gubbi, S. Marusic, and M. Palaniswami, "Estimation of crowd density by clustering motion cues," *The Visual Computer*, pp. 1–20., 2014.

[64] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14. Oakland, CA, USA., 1967, pp. 281–297.

[65] R. Gonzalez and R. Woods, *Digital Image Processing (3rd Edition)*. Prentice Hall, 2007.

[66] Á. Utasi, Á. Kiss, and T. Szirányi, "Statistical filters for crowd image analysis," in *Proceedings of the 11th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*. IEEE, 2009.

[67] A. B. Chan, M. Morrow, and N. Vasconcelos, "Analysis of crowded scenes using holistic properties," in *Performance Evaluation of Tracking and Surveillance workshop at CVPR*. IEEE, 2009, pp. 101–108.

[68] University of Minnesota, "Detection of unusual crowd activity," http://mha.cs.umn.edu/proj_events.shtml, [Online; verfified on 28-Jun-2015].

[69] M. Thida, H.-L. Eng, M. Dorothy, and P. Remagnino, *Learning Video Manifold for Segmenting Crowd Events and Abnormality Detection*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2011, vol. 6492, book section 34, pp. 439–449.

[70] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2. IEEE, 1999, pp. 246–252.

[71] S. Ince and J. Konrad, "Occlusion-aware optical flow estimation," *IEEE Transactions on Image Processing*, vol. 17, no. 8, pp. 1443–1451, 2008.

[72] C. Ballester, L. Garrido, V. Lazcano, and V. Caselles, *A TV-L1 Optical Flow Method with Occlusion Detection*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, vol. 7476, book section 4, pp. 31–40.

[73] S. Raza, A. Humayun, I. Essa, M. Grundmann, and D. Anderson, "Finding temporally consistent occlusion boundaries in videos using geometric context," in *2015 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, Jan 2015, pp. 1022–1029.