We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



14

Group Tracking Algorithm for Crowded Scene

Mohd Rizal Arshad and Umair Soori
Underwater Robotics Research Group (URRG), Universiti Sains Malaysia
Malaysia

1. Introduction

Automatic visual monitoring systems gain more significance as the enhancement of high speed processors that are able to execute the prolonged algorithm efficiently. Video surveillance monitoring systems are typically applied to study human behavior in a crowded situation. It is used to observe posture, movement, trajectories and the interaction between people. For this purpose CCD cameras are installed at many crowded places. Most researchers' work however is focused on the people classification and the recognition of activities. Human identification is important because of a need of surveillance in the crowded scenes. Group motion tracking is applied to monitor the Groups of People (GOP) as single entities in the global flow rather than monitoring each person individually. It has been studied on the basis of dynamic movement and the surrounding of crowd density which will affect the speed of an individual/group. Large crowd flow perhaps causes deadly accidents if large numbers of GOP smash into each other in narrow passageways, streets or bridges. The system must identify multiple GOP within their global flow according to their Spatio-temporal characteristics. Figure 1 illustrates the density of large crowd flow. The Religious and Sport activities are the application examples. According to Crowd Dynamics Ltd, (Crowd Dynamics Ltd) in the year of 2006, 363 pilgrims lost their lives while performing the Hajj. There are a number of readily available wide area surveillance systems but most of the systems are not suitable for high density crowd monitoring. For safety surveillance of large crowd, there should be a system which can detect and track unstable GOP by careful monitoring of crowd and display results in real-time scenario. And, with the results being displayed in real-time, it will certainly assist the system's operator to make better decision.



(a) (b) (c) (d)

Fig. 1. Examples of large crowd flow, (a) New York City Marathon (b) Stoning stage in Makkah (c) Dense Traffic and (d) Fish Schooling

The literature on large crowd flow has got a less coverage from researchers. In the case of large crowd flow scenarios, such as huge flow of peoples as in religious ceremonies or in sport activates, where the flow of crowd have so many individuals or small numbers of groups. On this, commonly used processing techniques cannot be utilized for such uncertain conditions. Dense or huge crowded scenes have so many individuals and groups that pose considerably constraints on these current pre-processing techniques. Ali and Shah (2007) describes crowd flow segmentation.

The basic concept which they have used is the Lagrangian particle dynamics to uncover the spatial organization of the flow field generated by crowd flow. Their work is focussed on the crowd flow segmentation, rather than detection and tracking groups of people in a large crowd flow. Their technique is an off-line technique. They generated one mean field from sequence of images to perform the analysis.

The main objective of this research work is to develop an efficient monitoring system which can detect and track GOP within a huge crowd, with accurate and computationally efficient manner. A high density crowd flow may exhibits some basic properties of fluid flow. Techniques available in fluid mechanics can be applied to study the flow properties of GOP moving in a large crowd. Previously, crowd flow has been treated as fluid flow to study crowd behavior. The majority of research done in Computational Fluid Dynamics CFD is Eulerian and Lagrangian analysis. Eulerian analyses examine fixed points in space in a fluid flow, while Lagrangian analyses examine individual particles paths. Technically, global large crowd flow has many similar characteristics as the fluid flow. When coherent structures are studied in terms of its quantities as a consequence of crowd flow trajectories, they are named as Lagrangian Coherent Structures or LCS. The LCS separates the flow of distinct behavior by managing the boundaries where flow field experiencing the different dynamics. This flow boundary in unsteady flows was introduced in a series of papers by Haller (Haller (2001), Haller (2005a) and Haller (2005b)) the boundaries are referred as Lagrangian coherent structures (LCS). New method to evaluate LCS is Finite-Time Lyapunov Exponent field (FTLE) was developed by Shadden (Shadden *et al.* (2005, Shadden (2006)). Davies *et al.* (1995) presents the image processing techniques for monitoring crowd behavior. Another idea is to treat the crowd flow as a fluid dynamic, the crowd turbulence is used to study the physics of crowd disasters (Helbing *et al.* 2007). The density of crowd is an important isotropic quantity that appears to determine the probability of a deadly accident. The crowd flow contains two types of tragedies that is trampling and crushing of pedestrians (Lee and Roger 2005) Trampling occurs when pedestrians are in motion. Crushing usually occur when a moving crowd have a contact with a stationary crowd. The crowd motion organization phenomena such as the directional separation, lane formation, direction towards bottlenecks and stop-and-go type of flow also creates critical conditions. (Helbing *et al.* 2007b).

In crowded scenarios, the focus is on detecting and tracking groups in a global flow rather than in studying individual motion. In this circumstance, Lagrangian Coherent Structure is the most appropriate method. Lagrangian Coherent Structure or LCS is frequently used in fluid mechanics applications in order to distinguish the fluid flow and its associated characteristics. The basic inspiration to used LCS on crowd flow is to treat a flow as a single entity globally, and to track the unstable groups inside the flow. LCS can be computed in different ways, but Finite Time Lyapunov Exponent or FTLE field is the chosen approach (Shadden 2006) since it gives maximum stretching of the nearby particles.

2. Theoretical background

2.1 Lagrangian Coherent Structures LCS

The idea of Lagrangian coherent structures (LCS) stems from dynamical systems which have been extensively used in fluid dynamics analysis. LCS has also been used in the analysis of phase space of systems on the velocity vector field. Coherent motion is defined as a region at which at least one essential flow variable shows important correlation with another variable over a range of space and time Shadden (Shadden *et al.* 2005, Shadden 2006).

When coherent structures are studied in terms of its quantities as a consequence of crowd flow trajectories, they are named as Lagrangian Coherent Structures (LCS). The LCS is defined as ridges in the FTLE field. Ridges are the boundaries between the flows of distinct dynamics. The boundary can be determined by tracking flow particles and searching for the material lines that are named *separatrices*. It has effectively been utilized to partition the flow regions for different dynamics. The technical definition of LCS states that where the gradient of the FTLE field is normal to the eigenvector with the minimum eigenvalue of the Hessian matrix, a scalar field is formed that is the LCS Shadden (2006). The LCS is an invented boundary through which the amount of two or more fluids cannot pass to each other. All the particles within the divided region have similar behavior, known as coherent behavior.

In the case of time dependent systems, the crowd flow can be classified as stable and unstable manifolds of hyperbolic fixed points as coherent structures. But typically, one uses the notion of coherent structures in the context of more general flows. Modeling of a dynamical system in a LCS scene gives boundaries in the spatiotemporal separating region. This organization dynamically separates the distinct behavior in the crowd flow which is unseen in the velocity vector field. The boundaries of flow region can be determined by tracking the flow particles during advection which will indicate the material lines. Thus, flow regions with different characteristics would be highlighted. This flow boundary in unsteady flows was introduced in a series of papers by Haller (Haller 2001, Haller 2005a and Haller 2005b). These boundaries are referred to as Lagrangian Coherent Structures (LCS).

A vector field can be represented as a set of LCS. This LCS has sub regions in which each region contains similar behavior with respect to the crowd flow. The LCS reflects the large scale analysis of a vector field since they are based on the integration of trajectories. At each point in space, the LCS measures the rate of separation of neighboring particle and its trajectories. The LCS of unsteady fields usually distorts and shifts over time, but it still can be recognized. In fluid dynamics, LCS boundaries are the higher ridges of Finite Time Lyapunov Exponent Fields. This ridge refers to material surfaces that get advected with the crowd flow and move towards regions where the flow is separated in positive or negative time. The flow is separated along these boundaries surfaces.

The Lyapunov exponent measures the behavior of the flow particles in a dynamical system. It quantifies the exponential rate of convergence or divergence between particles of the neighboring trajectories in a global sense. A positive exponent involves divergence, and a negative one means convergence. Therefore, a system with positive exponents has positive information in that trajectories that are initially close together move at a distance over time. The more positive the exponent, the faster they separate. In the same way for negative exponents, the trajectories move together with each other. A system with both positive and negative Lyapunov exponents is said to be chaotic.

2.2 Finite-Time Lyapunov Exponent field (FTLE)

A new method to evaluate LCS is Finite-Time Lyapunov Exponent field (FTLE) was developed by Shadden (Shadden *et al.* 2005, Shadden 2006). The LCS can be calculated in various ways. One of these methods is based on the FTLE field. It evaluates a value that corresponds to how quickly two imaginary particles would separate from each other as the flow progresses. It measures the maximum linearized growth rate of the distance perturbation between the nearby flow particles over the time interval to its trajectories. In other words, it measures the rate at which two neighboring particles diverge from each other at a given location and time interval. The FTLE differentiates the amount of flow particle stretching about the trajectory over time. It is defined by the local maxima of the FTLE field which specifies regions with distinct dynamics in the flow Haller (2001).

Let us suppose that the flow field experienced distinct dynamics in two or more different regions of the same flow. Within each sub region, FTLE contains a coherent motion in the flow that is all the particles inside each region contain a similar behavior which is indicated by the Eigenvalue of λ_{max} (Δ) close to zero. The boundary of two flow region contains different dynamics in which particles at these boundaries encloses as incoherent behavior. This condition creates higher Eigenvalue which rises as boundaries in FTLE field. The maximum stretching of particles is given by the square root of the largest eigenvalue. As the stretching is produced by velocity vector field, it would increase exponentially with the time series. The logarithm of the resulting value is computed and additionally normalized by the absolute advection time $|T|$. This leads to the explanation of the FTLE field. The notation (formulation) of the FTLE will be defined in section 2.3.

The FTLE indicates sections in the flow with different dynamics due to flow particle pairs straddling the edges of the boundary that separate them. This is also to indicate that they are faster than other arbitrary flow particle pairs. Area of maximum material stretching generates local maxima of the velocity field that may specify either local maximal stretching or local maximal shear. Trajectories of the flow particles can be integrated in negative or positive time. This is similar to the scalar parameters in a flow gathering in the coherent structures which can be achieved from the flow vector field. Consequently, FTLE can be calculated by integrating trajectories in backward time ($T < 0$). The ridges in the FTLE field indicate attracting material lines or attracting Lagrangian coherent structures (attracting LCS) Shadden (2006).Integrating trajectories in forward time ($T > 0$) produce FTLE ridges that spot the location of distinct flow as material lines, or repelling Lagrangian coherent structures (repelling LCS).

These positive-time and negative-time LCS are defined as the boundary between qualitatively different regions in the flow. The integration time can be increased or decreased depending on the amount of feature required from the computation. In addition, the location of the ridge indicating the boundary of the dissimilar flow of crowd does not change.

2.3 Formulation of FTLE

Let the open set $D \subset \mathbb{R}^2$ be the domain of the flow of crowd under study. Given a time dependent velocity field $\mathbf{v}(\mathbf{x},t)$ defined on D. Categorize a trajectory of $x(t;t_0,x_0)$ which is starting at point $\mathbf{x_0} \in D$ at time t_0 to be the solution of

$$\begin{cases} \dot{\mathbf{x}}(t;t_{0,}\mathbf{x_0}) = \mathbf{v}(\mathbf{x}(t;t_0,\mathbf{x_0}),t), \\ \mathbf{x}(t_0;t_{0,}\mathbf{x_0}) = \mathbf{x_0} \end{cases} \tag{1}$$

The velocity field $\mathbf{v}(\mathbf{x},t)$ is satisfying the continuity assumptions C^0 in time and C^2 in space, where C^0 assumption is responsible for smoothness of the flow field. A trajectory of a particle $\mathbf{x}(t;t_0,\mathbf{x}_0)$ depends on the initial position, x_0, and the initial time, t_0, to its final position at time T. From velocity field $\mathbf{v}(\mathbf{x},t)$ and its continuity assumptions, it follows that trajectory $\mathbf{x}(t;t_0,\mathbf{x}_0)$ is C^1 in time and C^3 in space. The solution of the dynamic system, as defined in equation (1), is viewed as a map from their position x_0 at time t_0 to their position at time t. This is referred to as a *flow map*, and denoted by $\phi_{t_0}^t$, and it satisfies equation (2)

$$\phi_{t_0}^t : D \to D : \mathbf{x_0} \mapsto \phi_{t_0}^t(\mathbf{x_0}) = \mathbf{x}(t;t_0,\mathbf{x_0}) \tag{2}$$

Furthermore, flow map $\phi_{t_0}^t$ satisfies the following properties as shown in equation (3) and (4)

$$\phi_{t_0}^t(\mathbf{x}) = \mathbf{x}, \tag{3}$$

$$\phi_{t_0}^{t+s}(\mathbf{x}) = \phi_s^{t+s}(\phi_{t_0}^s(\mathbf{x})) = \phi_s^{t+s}(\phi_{t_0}^t(\mathbf{x})) \tag{4}$$

here s defined as a arbitrary constant, but fixed $s \in \mathbb{R}$ (Theorem 3.1, [1]). Now, after a time interval T passes, advection of particles moves from $\phi_{t_0}^t$ to $\phi_{t_0}^{t_0+T}$ to calculate the amount of stretching to its trajectory, consider the evolution of the perturbed point illustrated in equation (5)

$$\mathbf{y} = \mathbf{x} + \delta\mathbf{x}(0) \tag{5}$$

Where $\delta\mathbf{x}(0)$ is infinitesimal and arbitrarily oriented. After a time interval T, this perturbation becomes Equation (6)

$$\delta\mathbf{x}(t_0+T) = \phi_{t_0}^{t_0+T}(\mathbf{y}) - \phi_{t_0}^{t_0+T}(\mathbf{x}) = \frac{d\phi_{t_0}^{t_0+T}(\mathbf{x})}{d\mathbf{x}}\delta\mathbf{x}(0) + O(\|\delta\mathbf{x}(0)\|^2) \tag{6}$$

Since $\delta\mathbf{x}(t_0)$ is infinitesimal, the method can assume that the $O(\|\delta\mathbf{x}(t_0)\|^2)$ term is negligible, from which the growth of linearized perturbations are obtained. The magnitude of the perturbation is shown in Equation (7)

$$\|\delta\mathbf{x}(t_0+T)\| = \sqrt{\left\langle \delta\mathbf{x}(t_0), \frac{d\phi_{t_0}^{t_0+T}(\mathbf{x})}{d\mathbf{x}}^* \frac{d\phi_{t_0}^{t_0+T}(\mathbf{x})}{d\mathbf{x}}\delta\mathbf{x}(t_0)\right\rangle} \tag{7}$$

where * denotes the transpose of matrix of the matrix, while $\frac{d\phi_{t_0}^{t_0+T}(\mathbf{x})}{d\mathbf{x}}$ is the spatial gradient of the flow map.

$$\Delta = \frac{d\phi_{t_0}^{t_0+T}(\mathbf{x})}{d\mathbf{x}}^* \frac{d\phi_{t_0}^{t_0+T}(\mathbf{x})}{d\mathbf{x}} \tag{8}$$

In equation (8) Δ is a *finite-time* version of the Cauchy Green deformation tensor. Maximum separation of advected particles initially located at $\mathbf{x}_0$ t_0 when $\delta\mathbf{x}(0)$ is aligned with the eigenvector which is associated with the maximum eigenvalue of Δ , shown in Equation (9)

$$\max_{\delta\mathbf{x}(0)} \left\| \delta\mathbf{x}(T) \right\| = \sqrt{\lambda_{\max}(\Delta)} \quad \left\| \overline{\delta\mathbf{x}}(0) \right\| \tag{9}$$

Now introducing the finite-time Lyapunov exponent (FTLE) $\sigma_t^T(\mathbf{x})$ in Equation (10)

$$\sigma_{t_0}^T(\mathbf{x}) = \frac{1}{|T|} \ln \sqrt{\lambda_{\max}(\Delta)} \tag{10}$$

Equation (10) is of Finite-time Lyapunov exponent (FTLE) with a finite integration time T, which is related to point $\mathbf{x}_0 \in D$ at time t_0. This technical and mathematical explanation of FTLE field with respect to eigenvalue is first given by Shadden [2005, 2006]. The FTLE can be described as the flow region experiences distinct dynamics in two or more different areas of the same flow. Within this each region, we have a coherent motion in the flow i.e. all the particles inside each region contain similar behavior which is indicated by the eigenvalue of $\lambda_{\max}(\Delta)$ close to zero. While at the boundary of two regions, having different dynamics, particles at these boundaries having an incoherent behavior, creates higher eigenvalue which are rise as boundaries in FTLE field. The absolute value |T| is used instead of T in Equation (10) for the reason that FTLE can be calculated for T > 0 and T < 0. The material line is called a repelling LCS when (T>0) over the time interval in forward time. On the other hand the material line is called an attracting LCS (T < 0) over the interval in backward time. Repelling and attracting LCS reveals stable and unstable manifolds of a dynamic system.

2.4 Eigenvector parallelism method

Eigenvalues play an important role in conditions where the matrix is a transformation from one vector space onto itself. Systems of linear ordinary differential equations are the primary examples. The values can correspond to frequencies of vibration, or critical values of stability parameters, or energy levels of atoms it has been discussed in more detail by Moler, C. (2008). As mentioned earlier, LCS can be defined as ridges on the flow field where particles show incoherent behavior. This phenomenon can be explained by the existence of higher eigenvalues at certain region of the flow field. These boundaries separate two regions, and are known as separatrices. Moreover the method requires an approach to extract the information from FTLE field, and identify the precise unstable GOP with respect to its surroundings.

In recent years, valuable developments have been made in image segmentation. Several algorithms have been proposed in which Normalized cut algorithm, is one of the method used for image segmentation first proposed by Shi and Malik (2000). This Normalized cut algorithm is also applicable to perform segmentation on the boundaries of FTLE field. One weakness of this algorithm is that, it process at high computational time to produce results. But in this application, the prime goal is to make the algorithm efficient enough to execute in real-time in real world scenarios. This requires fast running algorithm. In order to overcome this problem generalized eigenvalue is utilized to measure vector parallelism, named this phenomenon as Eigenvector Parallelism. This technique calculates eigenvalue divergence on

the FTLE boundaries to identify/segment the exact unstable GOP with respect to their global flow. Consequently, it will differentiate the unstable or hazardous GOP's boundaries from its flow boundaries with a low computational time.

2.5 Formulation of Eigenvector Parallelism Method

In Eigenvector Parallelism Method, the technique first correlates the two constitutive mean flow fields RM_u^{u+T} and holds the peak-mean value denoted as PM_u^{u+T} of the vector field F_u^{u+T}. The characteristic Equation of eigenvalues and eigenvectors is given as:

$$\boldsymbol{Ax} = \lambda \boldsymbol{x} \tag{11}$$

In Equation (11), λ is an *eigenvalue*, $\boldsymbol{x}$ is an *eigenvector* and the pair ($\boldsymbol{\lambda}$, $\boldsymbol{x}$) is called an *eigenpair* for matrix $\boldsymbol{A}$. All eigenvalues and eigenvectors satisfy the Equation (11) for a given matrix, $\boldsymbol{A}$. The generalized eigenvalue is used in this application to determine eigenvalues between PM_u^{u+T} and VR_u^{u+T}. The nontrivial solutions of generalized eigenvalue can be written as

$$\boldsymbol{Ax} = \lambda \boldsymbol{Bx} \tag{12}$$

In Equation (12), $\boldsymbol{A}$ *and* $\boldsymbol{B}$ are matrices and λ is a scalar eigenvalue. The values of λ satisfy the equation is the generalized eigenvalues and the corresponding values of x are the generalized eigenvectors.

$$\boldsymbol{PM_u^{u+T} x} = \lambda\, \boldsymbol{VR_u^{u+T} x} \tag{13}$$

Equation (13) corresponds to the peak-mean value and the vector's position where the boundaries arise.

$$(\boldsymbol{VR_u^{u+T}})^{-1} (\boldsymbol{PM_u^{u+T}} - \boldsymbol{VR_u^{u+T}} \lambda)\, \boldsymbol{x} = 0 \tag{14}$$

Equation (14) is the simply form of Equation (13). It is a common form of generalized eigenvalue problem.

$$det\ [\ (\boldsymbol{VR_u^{u+T}})^{-1} (\boldsymbol{PM_u^{u+T}} - \boldsymbol{VR_u^{u+T}} \lambda)\ ] = 0 \tag{15}$$

Equation (15) reveals the Eigenvector Parallelism method. It finds the largest generalized eigenvalue between PM_u^{u+T} and VR_u^{u+T}. If λ, eigenvalue, is greater than high-threshold or less than low-threshold indicates unstable flow whereas if λ lies between the high and low thresholds it indicates the stable flow.

3. Implementation

This section is focused on the implementation of LCS. It requires optical flow which estimates the flow vectors field of the crowd. This flow vectors need filtered data for fine LCS implementation. Advection of particles is calculated by numerical integration method (Forth-order Runge-Kutta-Fehlberg). Flow map is used to map the advection of flow particles. Further on proposed method eigenvector parallelism is applied on the FTLE boundaries to identify the unstable flow. Figure 2 represents a brief overview of the methodology of the system. Steps of implementation process have been briefly described below:

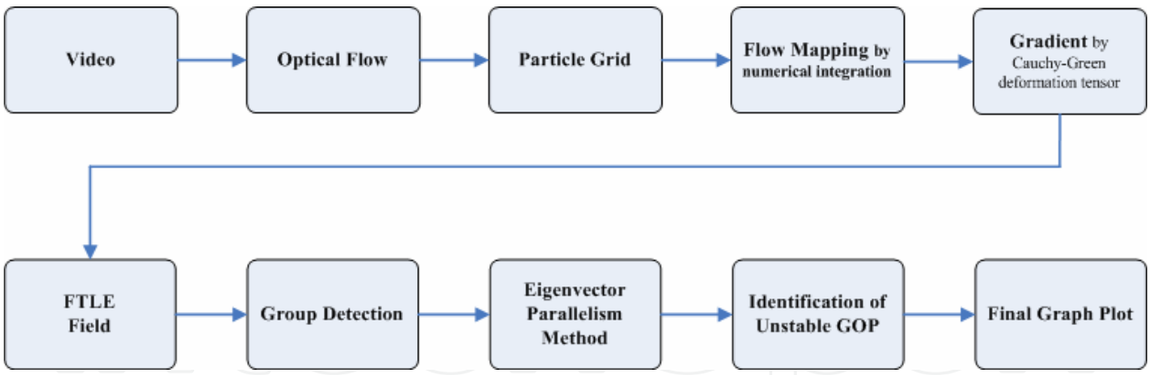


Fig. 2. Conceptual flow chart of the algorithm.

3.1 Optical flow

The input prerequisite for FTLE analysis is the velocity vector field. Optical flow, Lucas-Kanade (Lucas & Kanade 1981) technique is used. This is a two frame differential method to estimate motion in a moving object. This method aims to calculate the motion between two image frames which are taken at times t and $t + 1$ at every pixel location. This method is also frequently known as the differential methods, since it is based on local Taylor series approximations of the image signal. It also uses partial derivatives with respect to the spatial and temporal coordinates.

The term optic flow refers to visual phenomenon that has apparent visual motion that can be experienced as an object move through the world. There is a directly mathematical relationship between the magnitudes of the optical flow. If the speed of motion is doubled, the optic flow will also be double. The optical flow also depends on the angle between the direction of view and the moving objects. Advantages of the optical flow algorithm include that it yields a high density of flow vectors. That is, if the flow information missing in inner parts of homogeneous objects is filled-in from the motion boundaries

3.2 Smoothing of data

Data smoothing is an important process which contributes to simplify the LCS implementation. It is based on Noise Filtering and Mean value. Figure 3 shows the block diagram of data smoothing process. It is very useful technique which exposes the clear picture of obtained results from optical flow. The optical flow field of crowd motion carries

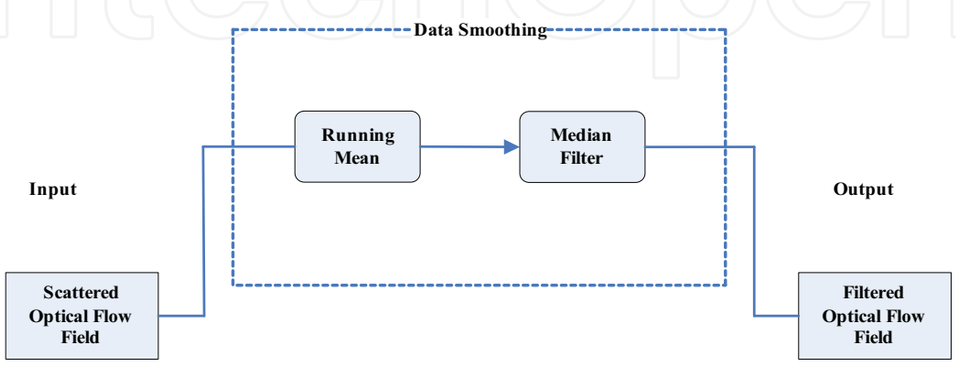


Fig. 3. Block diagram of data smoothing process.

scattered resulting vectors. This is shown in Figures 4 where it is observed that flow vectors of dense crowd have scattered direction and magnitude which is unable to define distinct group behaviors. Data smoothing helps to expose different groups present in the crowd flow by noise reduction and mean flow field.
The task for noise reduction is performed by median filtering, which is a non-linear spatial signal enhancement technique. While mean flow field is calculated by taking mean in time series or running mean. It is commonly defined as the continuing calculation. The degree of smoothness can be controlled by adjustment of the neighborhood range or the fitting weights. F_u denoted as optical flow field under observation and its running mean as RM_u, here u represents the number of frame in real time video sequence. Running means is programmed in this methodology to reset after a few frames, so that FTLE field can exactly identify the true position of the unstable group.

3.3 Advection of particles

In order to detect and track the movement of the flow of crowd, the system launched a Cartesian grid of particles. This movement is known as the advection of particles. These particles are placed on the running mean RM_u of flow field F_u, where particles have certain (constant) distance between them. Initially, particle position is $\mathbf{x}_0$, at time t. As time T passes, flow of crowd change its position therefore, the optical flow field becomes F_u^{u+T} and its running mean RM_u^{u+T}. The final position of particle turn out to be to $\mathbf{x}(t+T;t,\mathbf{x}_0)$. Each particle advection is computed with a forth-order Runge-Kutta-Fehlberg algorithm.

3.3.1 Forth-Order Runge-Kutta-Fehlberg

The Runge–Kutta method is an important tool. It is used to perform the approximation of solutions of ordinary differential equations. There are different types of Runge–Kutta methods, classified by how many points are used within each time step. The method applied in the FTLE computation is the 4th Order Runge–Kutta method. This is the most often used method of the Runge-Kutta family. It extends the idea of the mid-point method by using the information.
The fourth order Runge-Kutta algorithm requires four gradient or ''k'' terms (as given in equation 16), which can calculate from following equations (17), (18), (19) and (20)

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{16}$$

$$k_1 = f(y_n, t_n)\Delta t \tag{17}$$

$$k_2 = f(y_n + \frac{k_1}{2}, t_n + \frac{\Delta t}{2})\Delta t \tag{18}$$

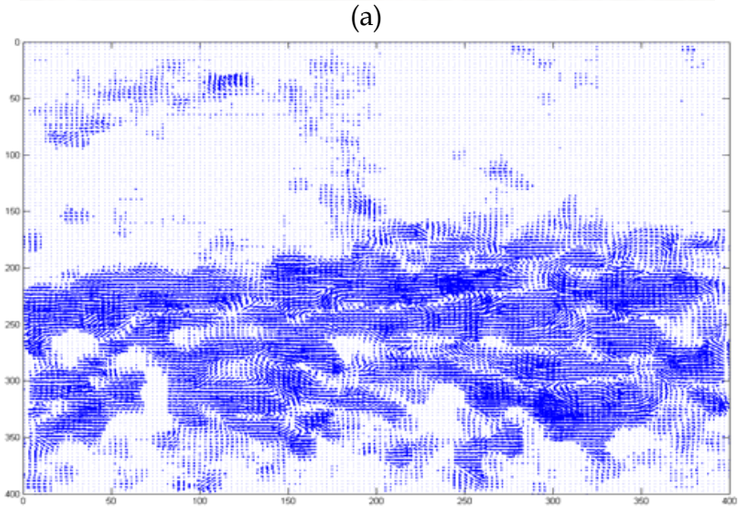
$$k_3 = f(y_n + \frac{k_2}{2}, t_n + \frac{\Delta t}{2})\Delta t \tag{19}$$

$$k_4 = f(y_n + k_3, t_n + \Delta t)\Delta t \tag{20}$$

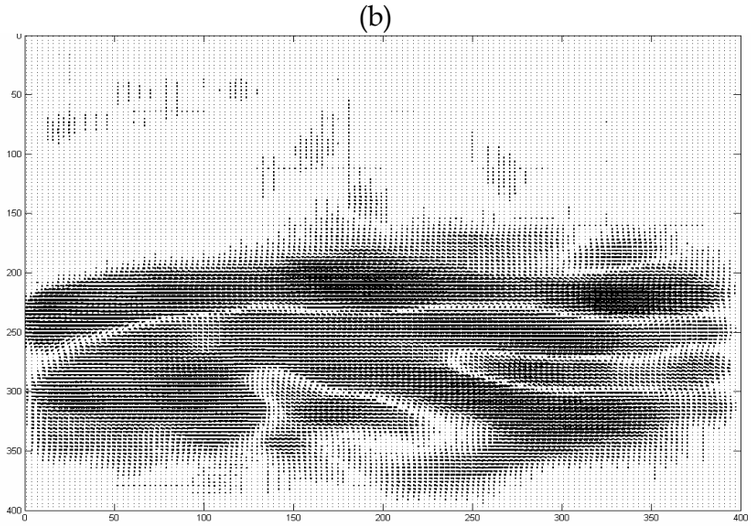
Where k1, k2, k3 and k4 are the gradient of fourth-order Runge-Kutta algorithm. This advection of particles is mapped by flow map. Figure 5 demonstrate the Cartesian grid of particles.



(a)



(b)



(c)

Fig. 4. The of optical flow result. (a) Video frame of the scene. (b) Optical flow field. (c) Running mean and filtered optical flow field.

For a clear illustration of particle advection, consider the two nearby flow particles, as shown in Figure 5. Initially, at position $\mathbf{x}_0$ $\mathbf{y}_0$, these two arbitrary flow particles are at the sides of the boundary where the dynamical change occurs. As the flow progress to time interval $\boldsymbol{T}$, these particles quickly separate and diverge from each other. As a result, a boundary is formed. This boundary is referred to as ridges in the FTLE field.

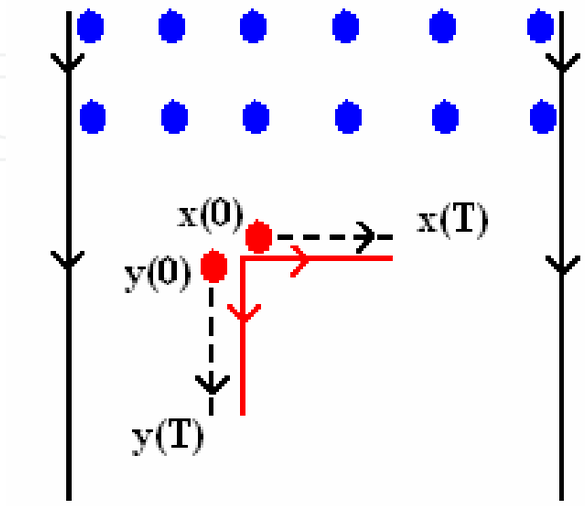


Fig. 5. Two flow particles (red in color) diverge from each other, as a result boundary formed (red in color). While blue circles represents as a grid of particles

3.4 Flow mapping

The flow map is used to map flow particles from its initial position at time t, to their final position at time $(t+T)$. Consequently flow map ϕ_x,ϕ_y tracks the positions of x and y particles respectively, due to the flow field generated by the movement of crowd. Note that the method reset the running mean after few frames in order to get the exact information of moving objects. In this formulation, the time interval T represents the next frame in a real-time video sequence. Figure 6 shows the illustration of flow mapping.

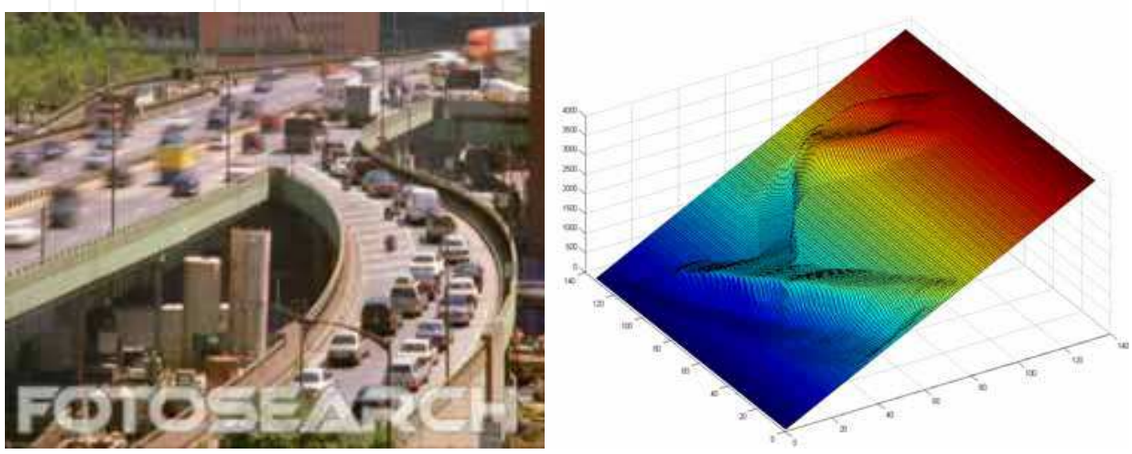


Fig. 6. Illustration of flow mapping. (a) Video frame of the scene. (b) Flow mapping results.

3.5 FTLE field

In order to obtain the FTLE field, the gradient of the flow map has to be calculated. $\frac{d\phi_x}{dx}, \frac{d\phi_x}{dy}, \frac{d\phi_y}{dx}$ and $\frac{d\phi_y}{dy}$ are refer to as spatial gradients. Gradient of flow map can be calculated by Equation (21)

$$\frac{d\phi_t^T}{dt} = \begin{bmatrix} \frac{d\phi_x}{dx} & \frac{d\phi_x}{dy} \\ \frac{d\phi_y}{dx} & \frac{d\phi_y}{dy} \end{bmatrix} \tag{21}$$

$$\frac{d\phi_t^T}{dt}\Big|_{(x_{i,j}(t),y_{i,j}(t))} = \begin{bmatrix} \frac{x_{i+1,j}(t+T)-x_{i-1,j}(t+T)}{x_{i+1,j}(t)-x_{i-1,j}(t)} & \frac{x_{i,j+1}(t+T)-x_{i,j-1}(t+T)}{y_{i,j+1}(t)-y_{i,j-1}(t)} \\ \frac{y_{i+1,j}(t+T)-y_{i-1,j}(t+T)}{x_{i+1,j}(t)-x_{i-1,j}(t)} & \frac{y_{i,j+1}(t+T)-y_{i,j-1}(t+T)}{y_{i,j+1}(t)-y_{i,j-1}(t)} \end{bmatrix} \tag{22}$$

Equation (21) is the extended form of Equation (22) in which finite derivatives with values at the neighboring grid points are used where *(i, j)* denotes position of the point in the FTLE computational grid. This phenomenon was first explained by Shadden (2006). The initial position of grid particles is found to be $x_{i,j}(t), y_{i,j}(t)$. After time T passes, the particles on grid get advected to $x_{i,j}(t+T), y_{i,j}(t+T)$. Figure 7 demonstrate the spatial gradients of the flow map.

Finally, the formulation is ready to be used in calculating the FTLE field in Equation (10). The FTLE returns the edges which are ridges where the crowd flow displays distinct behavior.

Finite-time Lyapunov exponents (FTLE) measure flow particle stretching as it moves with the crowd flow for a given time T. The FTLE field is computed by advection of each sample point $\mathbf{x}_0 \in D$ at time t_0 with the flow for time T. This will result in a flow map ϕ_x, which maps a sample point x_0 to its advected position. The maximum stretching of particles can be computed from the gradient of the flow map. Maximum stretching of particles are given by the eigenvector that belongs to the square root to the largest eigenvalue.

Positive and negative integration time intervals can be used to determine forward and backward time FTLE fields, to create the repelling and attracting LCS. Particle grid is used to determine the region of the flow, where the LCS is located at the appropriate integration time, T. A denser grid for higher spatial resolution calculation will require higher computational cost. Figure 8 show the FTLE results on the running frames.

3.6 Eigenvector parallelism method

This peak-mean value, $PM_u{}^{u+T}$, represents the majority vector behavior in the flow field. Note that the method is computing the peak-mean value after every integration length |T|, after every frame, in order to update the peak-mean position. Peak-mean value contains the global flow characteristic. It is the dominant flow, and inside it, small GOPs are also propagating. Figure 9 reveal the peak-mean values calculated on its respected flow. It shows the robustness of calculated peak-mean values position wise.

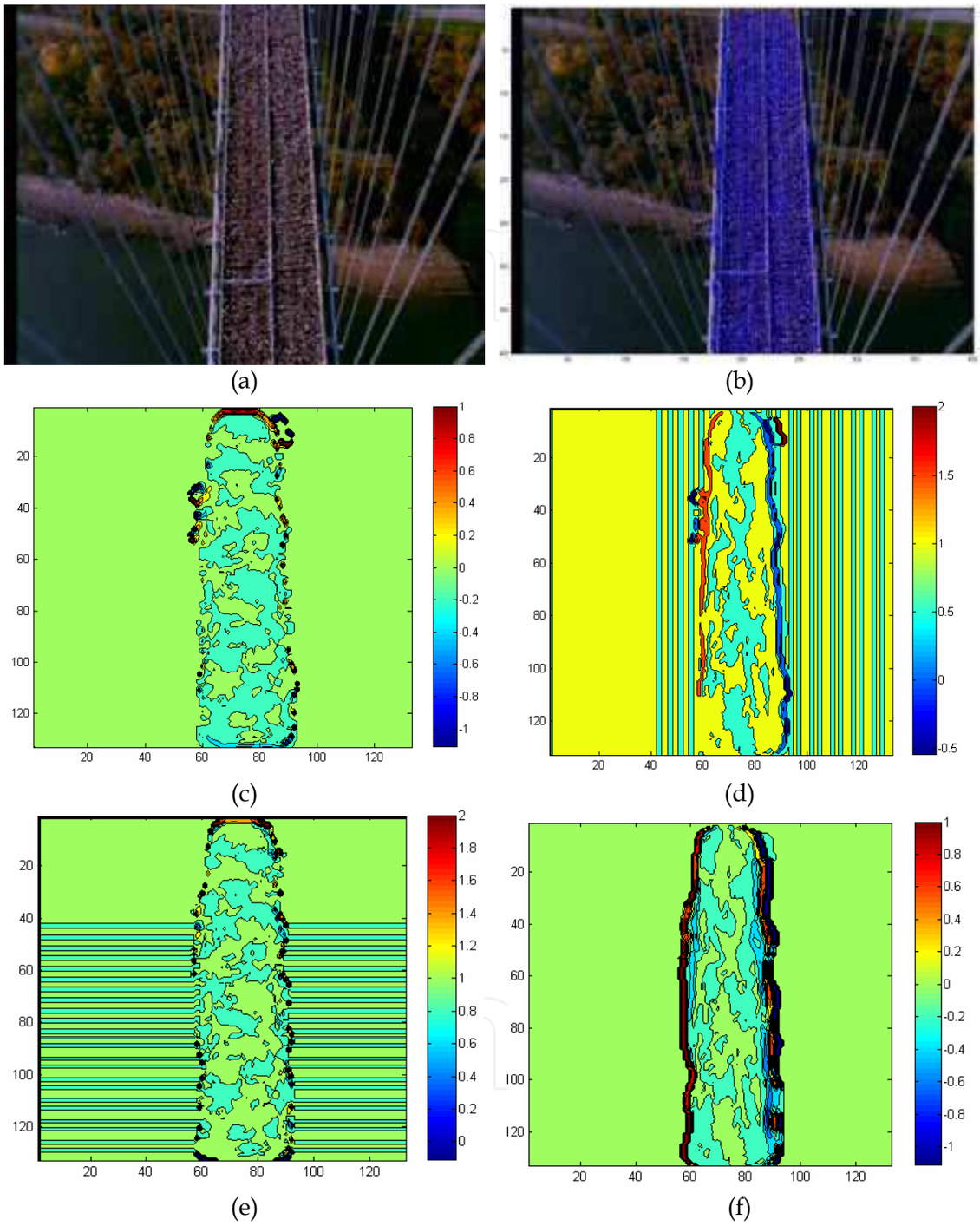


Fig. 7. Result of (a) original video frame. (b) Optical flow field. (c) Gradient of the flow map with respect to $\frac{d\phi_x}{dx}$ (d) Gradient of the flow map with respect to $\frac{d\phi_x}{dy}$ (e) Gradient of the flow map with respect to $\frac{d\phi_y}{dx}$ (f) Gradient of the flow map with respect to $\frac{d\phi_y}{dy}$

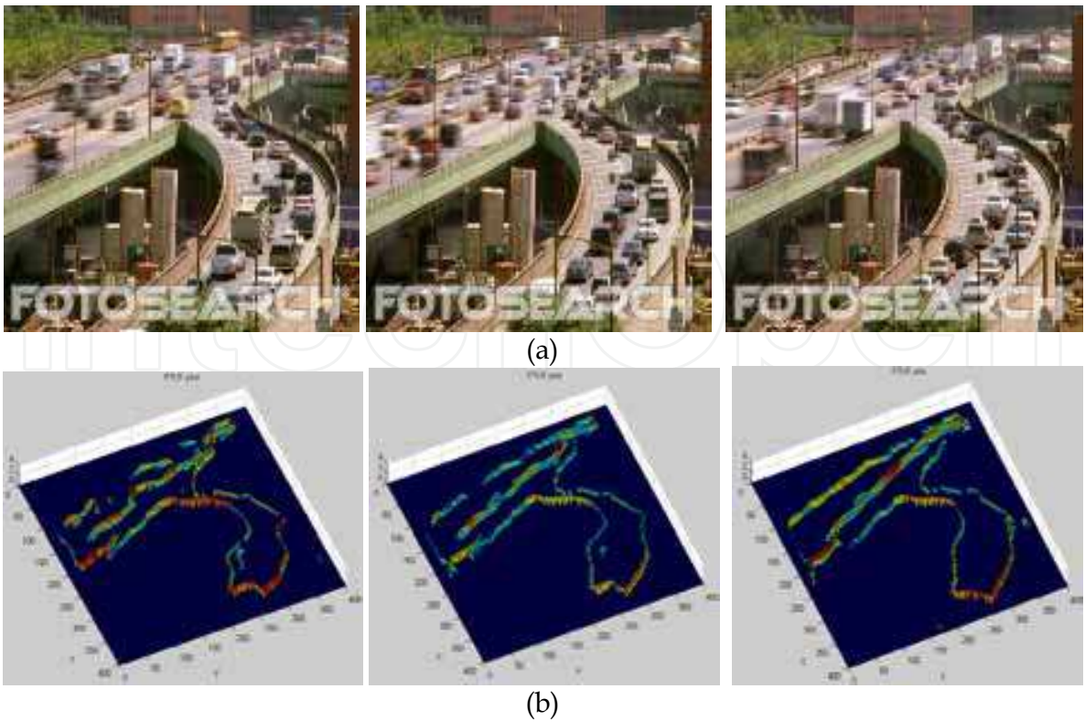


Fig. 8. Result of (a) original video frames (b) Running FTLE field, boundaries occur where flow having incoherent dynamics, inside the boundary region all particles have similar fate

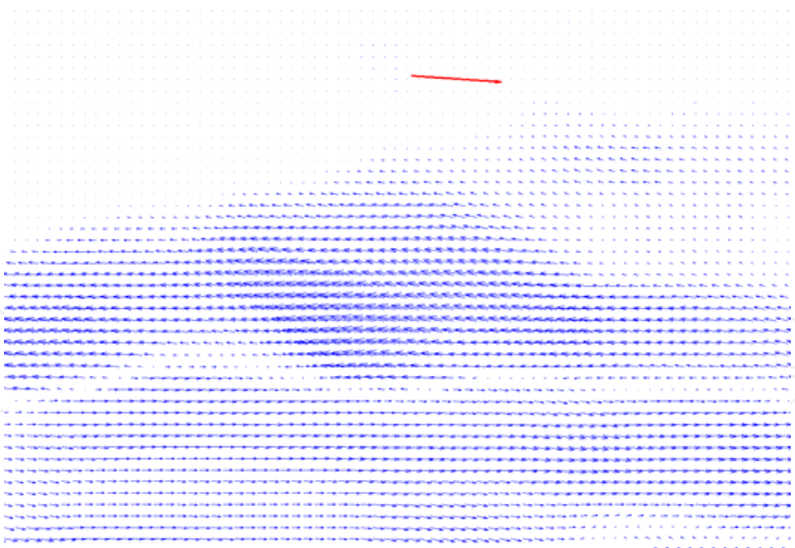
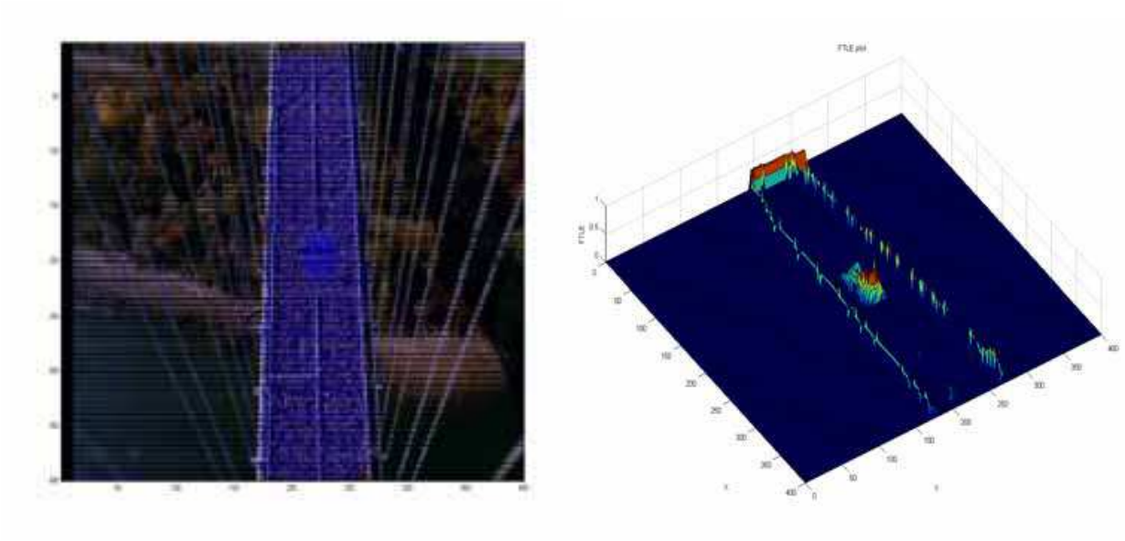
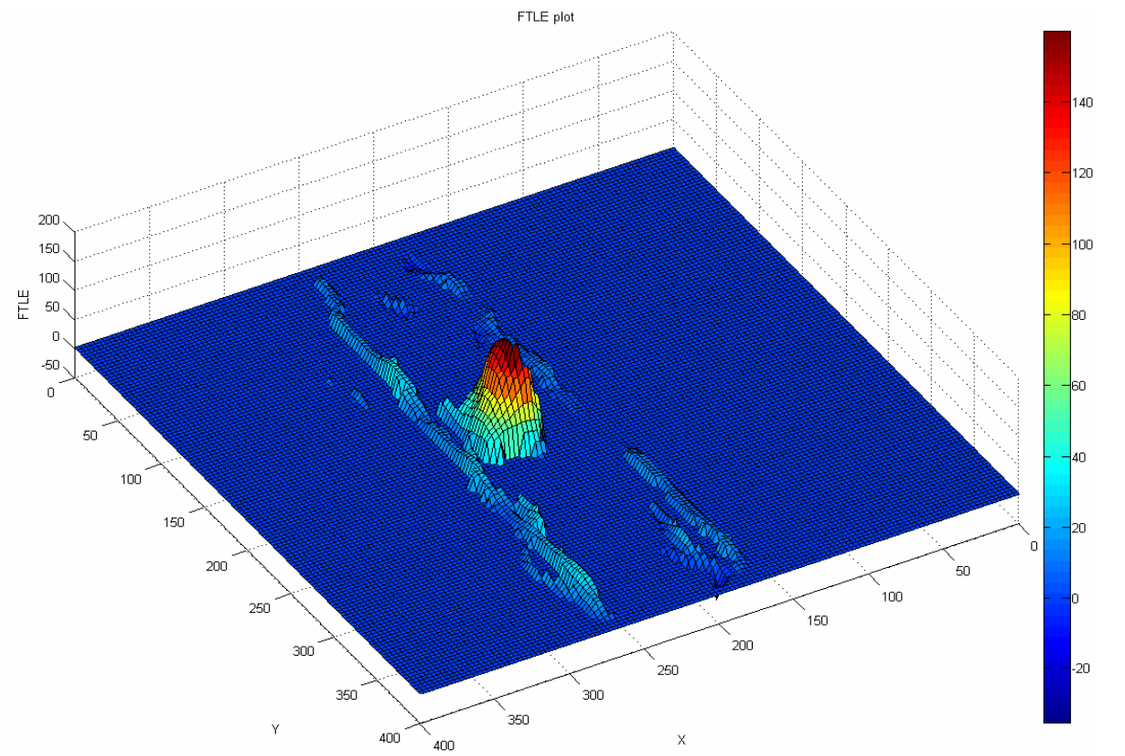


Fig. 9. The Peak-Mean value zoomed in window of sequence shown in figure 4, red vector represents Peak-Mean value.

The system calculates the eigenvector divergence by computing the eigenvalue between PM_u^{u+T} and on the vectors at same position where the boundary or ridges occur in the FTLE field. This shows significant results in which, the boundaries of FTLE field having incoherent behavior with respect to global flow signify as higher eigenvalue as compared to



(a) (b)



(c)

Fig. 10. The eigenvector parallelism method result. (a) Mean field of sequence New York City marathon with synthetic instability which is highlighted by red box. (b) FTLE field of the sequence, where in middle there is another boundary shows the unstable GOP. (c) Eigenvector parallelism method result, where unstable GOP identify by higher eigenvalue.

the entire flow FTLE boundaries eigenvalue. This occurs because vectors at the boundaries of FTLE field of unstable GOP have distinct behaviors. Whereas, PM_u^{u+T} contains the majority behavior in the global flow. When the system computes the eigenvalue between PM_u^{u+T} and the vectors position at boundaries of FTLE field, as a result the stable flow boundaries shows approximately similar characteristic as the peak-mean value. Therefore, the eigenvalue minimize these boundaries, showing evidence of the analogous flow. In other case, unstable GOP boundaries are detected by the FTLE field owing to different dynamics. Vectors at these boundaries give higher eigenvalue when they are computed with PM_u^{u+T}. Figure 10 shows the results of Eigenvector Parallelism Method where unstable GOP is detected as higher eigenvalue.

This approach has been tested by changing the direction of unstable GOP on self introduced instability video sequences, and subsequently, observing the imperative results. As the incoherent or unstable GOP changes its flow direction with respect to the global flow, eigenvalue will be increased. Greater the change in direction of flow vector, higher will be the eigenvalue. It will increase until the vectors aligned at the opposite direction as compare to the peak-mean value. In such case where vectors are aligned in opposite directions, a negative eigenvalues in obtained showing unstable GOP. Figure 11 elucidates the negative eigenvalue of GOP, where GOP contains opposite direction with respect to entire flow.

3.6.1 Identification of unstable groups

Unstable GOP has dissimilar dynamical behavior and LCS is used to track their boundaries. For the convergence of the detected unstable groups, the technique employed a simple scheme to study the neighboring flow. Let the boundary box of detected unstable group under action denoted by $B_{k,l}$, where k, l is its dimensions. At this instance, patch of box $R_{k',l'}$ is launched on the $B_{k,l}$, such that size of k', l' is determined by a ratio. It can be obtained by taking a ratio between the image frames to the detected object. The resulting ratio will be an integer which will increase the dimension of k', l' up to a certain fold. The size of $R_{k',l'}$ is determined by this ratio which is minimizing the computational cost. However different patch size gives different computational cost. A comparison of computational cost is shown in Table 1. In this example, similar detected object is used, in which different ratios are tested in order to calculate the computational costs.

Ratio	20	40	60
Time (sec)	0.0469	0.0625	0.0938

Table 1. Comparison of computational cost at different ratios.

Note here that if k', l' go out to the image boundary, then the method reinstate that side of k',l' to the image frame. The maximum limit of k',l' turn out to be the image frame boundary. The typical example of patch demonstrated in Figure 12(a). This patch contains the information of the neighboring behavior of the vectors in a flow as shown in Figure 12(b). If the patch demonstrates a change in a behavior with respect to the global flow then it reveals that the unstable group exists. However in other case if patch demonstrates the similar behavior with respect to global flow it means that there is no such unstable group exists. Afterwards, the detected boundary box $B_{k,l}$ either detected as instability or merge with the global flow.

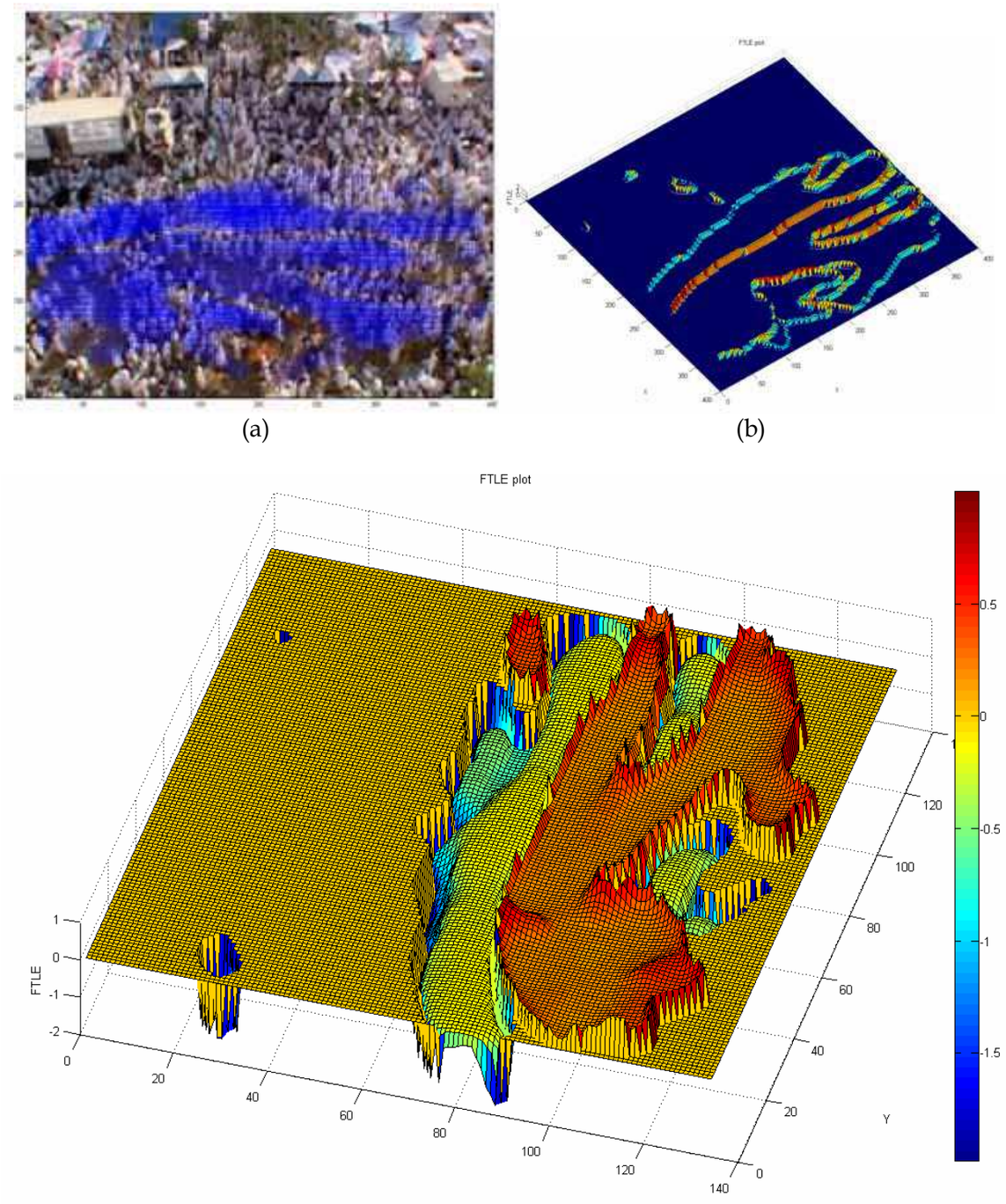
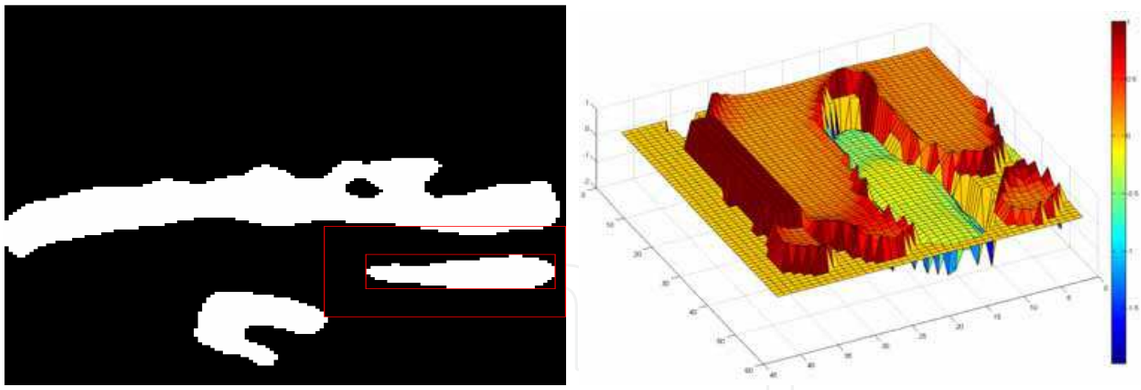


Fig. 11. The eigenvector parallelism method result (a) Mean field of sequence extracted from the National Geographic documentary, titled 'Inside Mecca' where two GOP propagates inside the big group against the direction, highlighted by red box (b) FTLE field of the sequence (c) Result shows GOP against the direction encloses by less than zero eigenvalue



(a) (b)

Fig. 12. The result of identification of exact unstable groups. (a) Detected unstable GOP blob and its patch. (b) Patch contains the neighboring behavior of detected unstable GOP.

4. Results and discussion

Group motion detection is a significant tool to study the GOP behavior in a crowded environment. We have utilized LCS to examine GOP in a large gathering. We have tested the algorithm using a range of videos taken from Ali and Shah (2007). This consists of high density crowded scene including videos during Hajj, New York City Marathon and traffic scenes.

4.1 Group detection

In the sequence entitled 'Inside Makkah', as shown in Figure 11(a), a dense and random crowd flow is observed. In this video scene, the direction of global flow is from left to right, three GOP can been seen moving against the direction of the global flow which is detected by Eigenvector Parallelism Method as shown in Figure 11(c). It is noted that two GOP are having hazardous motion dynamics with respect to surrounding crowd because they are disturbing the global flow. It can computed by using method explain in Section 3.6. The third GOP is not hazardous despite having different direction because they are moving independently without propagating inside the moving crowd. Figure 13(a) is the watershed plot used for clear visualization of unstable GOP. Similarly Figure 13(b) is the watershed plot of video sequence shown in Figure 10. Now a comparison of obtained results has been performed with well known methods i.e. K-Mean on optical flow *(u,v)* field. It is necessary to mention here, that numbers of segments are pre-defined to run this method. Figure 13(c) displays result of K-Mean method which is unable to detect unstable GOP present in the scene although, number of segments has been increased up to 12 in this case. Figure 13(d) shows resulting flow segments of the same video sequence which is under discussion, but this result is obtained from Ali and Shah (2007). Figure 13(d) shows segmentation of crowd according to flow. It is unable to track small groups present at the bottom of the video frame having distinct dynamics. Their method identified the flow (at bottom flow) as single flow and represented it as a single segment. It is because there focus is on flow segmentation not on the tracking of GOP. In contrast, our method detected the flow pattern of crowd as well as tracking small GOP having distinct dynamics with respect to surrounding flow.

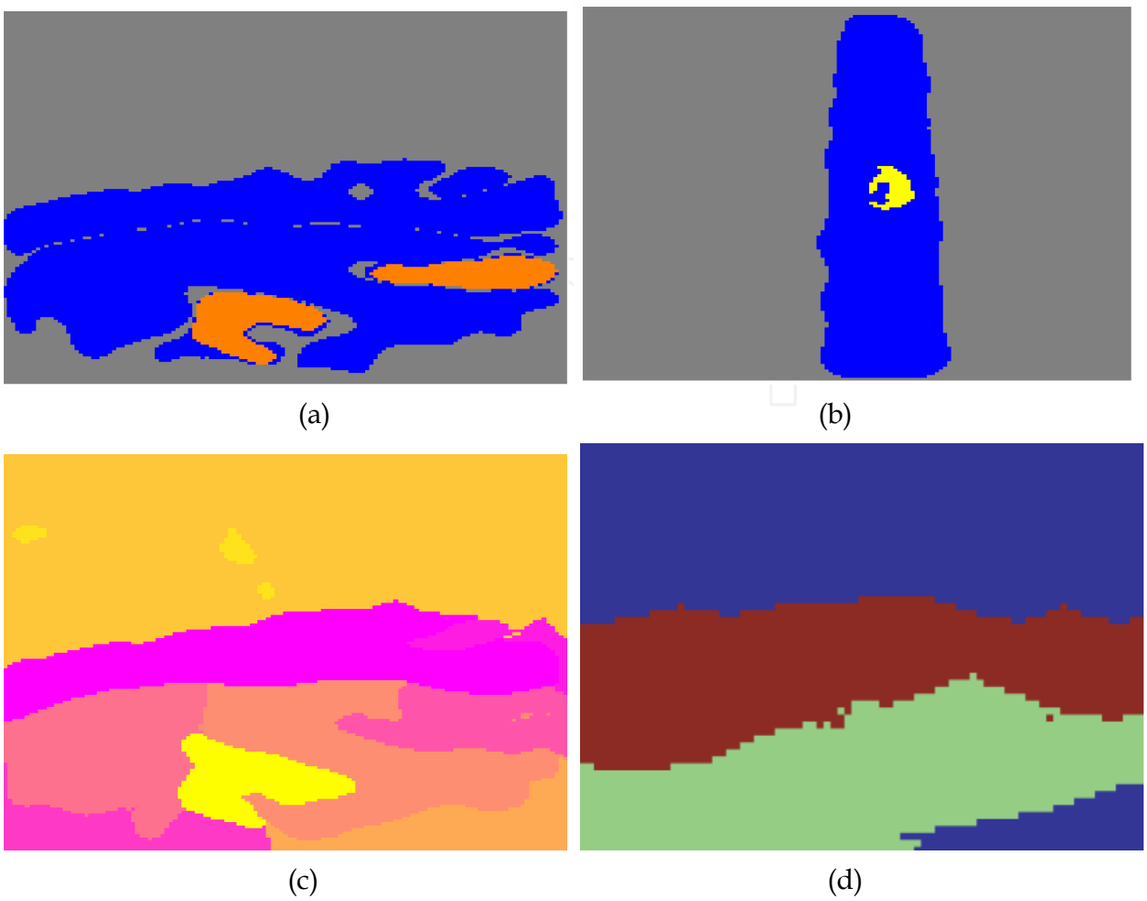


Fig. 13. Eigenvector Parallelism Method result and Comparison. (a) Watershed plot of video sequence, titled "Inside-Makkah", where unstable GOP is detected (in orange). (b) Watershed plot of video sequence New York City Marathon, where unstable GOP is detected. (c) Comparison plot with respect to K-Mean algorithm. (d) Flow segmentation result of video sequence as shown in Figure 11, result is obtained from Ali and Shah (2007).

The next result which is discussed here is shown in sequence Figure 1(b). The approach is tested on a video sequence of hurling stones at Jamarat Bridge. This video represented a highly important area where large crowd of people gather to perform a similar task. In the past, most of the deadly incidents occur at Jamarat Bridge due to collision of people in high density crowd. Detection and tracking of GOP having unstable dynamics are challenging task in this video sequence, since the flow pattern is quite complex and density of crowd is very high. The proposed method is able to work and produce results in such a highly dense crowd. The direction of global flow in this scene is from right to left. Few GOP are detected and tracked having different direction. The result of Eigenvector Parallelism Method is displayed in Figure 14(b).

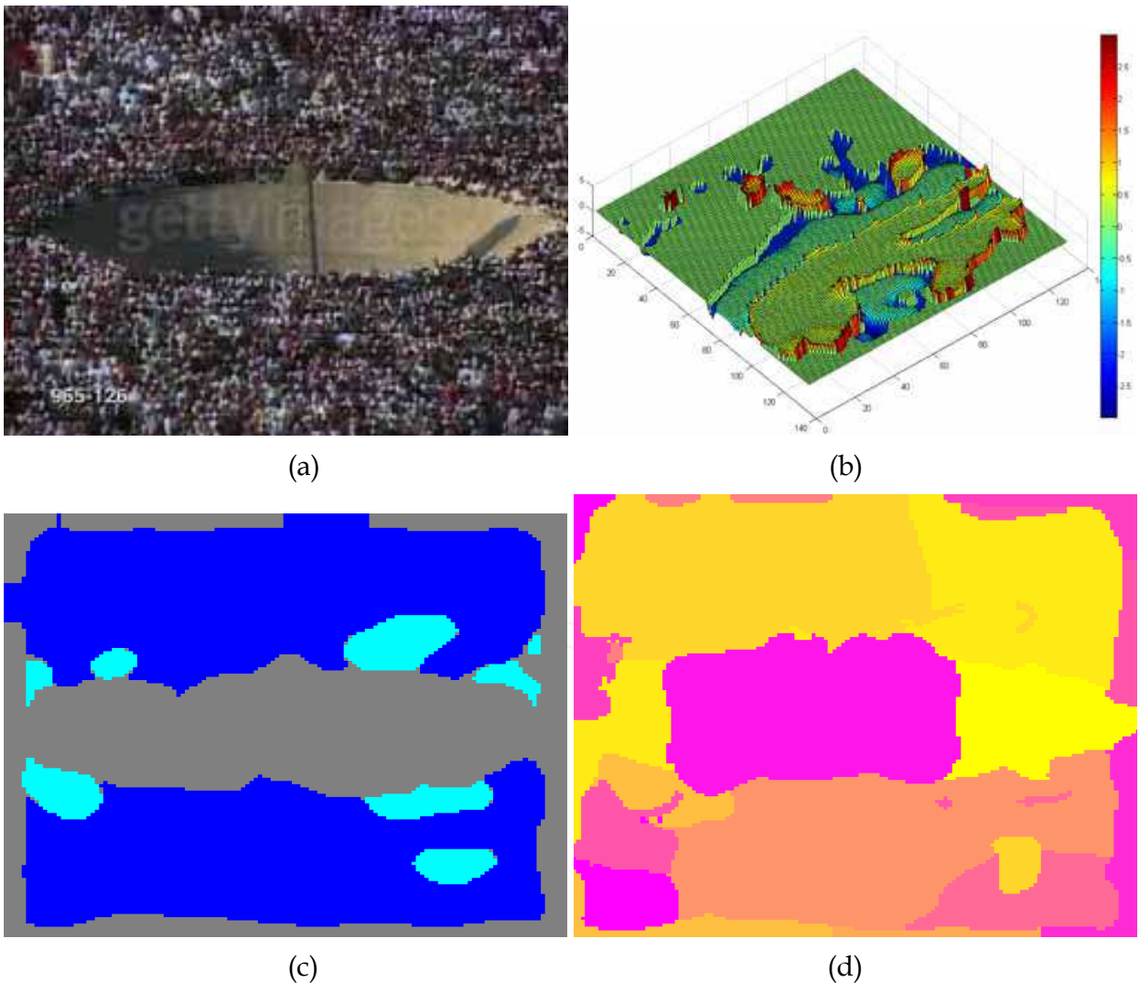
Figure 14(c) is the watershed plot used for clear visualization of unstable GOP. The result was then compared with K-Mean methods. In this case K-Mean is unable to segment the crowd flow exactly, due to complexity and density of the scene. In Figure 14(c) three hazardous groups are highlighted below vacuum region (at the bottom) while two GOP are present above vacuum region. While K-Mean method could only detects two GOP at the

bottom out of three while merged the upper groups with the global flow as shown in Figure 14(d). This shows the robustness of the proposed Eigenvector Parallelism Method.

4.2 Test set

The video sequence of New York City Marathon already used in Figure 10, in which artificial instability was generated by Ali and Shah (2007). In this sequence unstable GOP had different direction with respect to the global flow but GOP was static throughout the video. Now a moving GOP is introduced in the video sequence shown in Figure 15(a), the direction of mobile GOP is from left to right which is changing its position in every incoming frame. This continuous change in position makes the GOP a challenging object to track.

This is mathematically described as, Pt = (offset x + fr) + offset y. Where *Pt* is referred as position of patch (i.e. unstable GOP), *fr* is number of frame sequence which can be defined as *fr*= 1, 2, 3….n and *offset* is value where patch is initially launched.

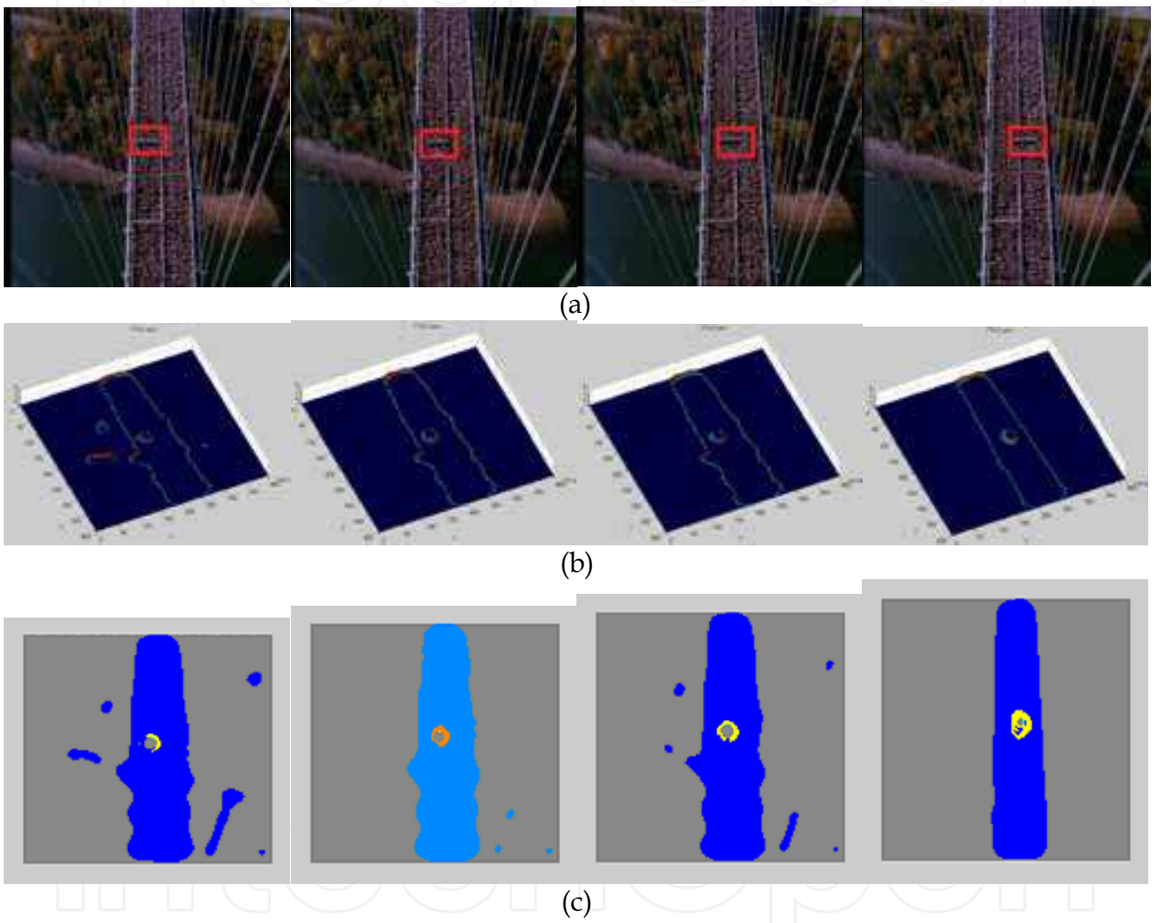


(a) (b)

(c) (d)

Fig. 14. Group detection result (a) Video frame of the sequence as shown in Figure 1(b). (b) FTLE field plot. (c) Watershed plot of the stable flow. (d) Comparison plot with respect to KMean algorithm

Now as a result of applying proposed algorithm, a mobile FTLE field is obtained, since GOP is changing its position continuously, ridges of unstable GOP boundary is also changing in correspondence. This is demonstrated in a series of resulting images as shown in Figure 15(b). These results validate the employment of technique on real time scenario and the algorithm is based on the frame wise calculation, the LCS is computed at every single frame rather than being computed on one mean field only. It is important to note that Running Mean in this methodology is programmed to reset after a few frames, so that FTLE field can exactly identify the true position of the unstable group. A significant result is shown in Figure 15(c).



Fg. 15. The real-time detection result. (a) Video frames of the synthetic instability, where GOP changing its position (in Red). (b) The FTLE field of frames. (c) Watershed plot, real-time detection of moving GOP (in yellow)

4.3 Algorithm computational time

The robustness of the work depends on the fast running algorithm that will run efficiently, in order to do real-time analysis. To apply the Lagrangian technique on the image sequences requires few numbers of frames in reading memory. These frames become the integration length |T| for the algorithm. This requirement indicates that Lagrangian methodology is only applicable for off-line analysis. The solution of off-line analysis problem is solved by

MatLab™ M-file and pre-system Simulink™ interface, both simultaneously worked together. Simulink model worked as pre-system, consist of reading frames from data acquisition box and data smoothing. While the M-file worked as post-system, contains of all the methodology of Lagrangian technique. This project is applicable for frame wise analysis rather than estimate one mean field from the integration length |T|. In this formulation, the integration length |T| represents the next frame in real-time videos. The algorithm was implemented, and all analyses have been conducted on a 3 GHz core2duo Pentium IV computer, executing Windows XP. The processing time for a single frame, size of 400x400 RGB image, is approximately 0.5 second that is 2 frames per second. This technique has assumed that the group motion is a slow performing motion because people moving in a large crowded scene. However, the execution time can be increased up to 5 frames per second by reducing the image size. Nevertheless, this will cause a loss of spatial image information.

In general Lagrangian sense, taking a denser particle grid will give higher spatial resolutions, which will lead to a higher computation time. If longer time integration is used, more of the boundaries will be exposed. Since flow particles were observed for longer time period consequently more boundaries will be exposed. This methodology cannot make use of longer time integration due to the computational cost restriction. Table 2 shows the computational cost of sub-systems for different sub-methods. Table 2 is computed on the sequence title "Inside Makkah".

Sub Methods	Mix	Min	Mean	Mode
Optical Flow	0.1073	0.0604	0.0804	0.076
Flow Map	0.0313	0.0156	0.0292	0.0313
Gradient and FTLE	0.3125	0.2656	0.2877	0.2813
Eigenvector Parallelism	0.1719	0.1406	0.1481	0.1406
Patch and Plot	0.0625	0.0313	0.0511	0.0469
Total	0.6855	0.5135	0.5965	0.5761

Table 2. Summary of the system by processing time (in seconds)

4.4 Comparison of computational time

LCS boundaries notify the distinct characteristic of crowd flow, whilst the eigenvector parallelism method characterizes the unstable group in their global flow. Figure 16 is a comparison graph plot between two methods, i.e. Normalize Cut algorithm and Eigenvector Parallelism method, computational-time wise. It can be observed, if the algorithm is executed with eigenvector parallelism method, it requires lower computational cost. In fact, three times lower as compared to Multi-scale Normalize Cut algorithm. Note at this instant, this technique is not proposing any comparative method of Normalize Cut algorithm. Eigenvector parallelism method has only used to classify the GOP having distinct characteristics on the boundary of the FTLE field.

5. Conclusion

This chapter provides a general framework for group motion detection and tracking in real-time crowded scenarios by using Lagrangian based approach. The main objective of this research is to detect and track Groups of People (GOP) that may create a calamity due to

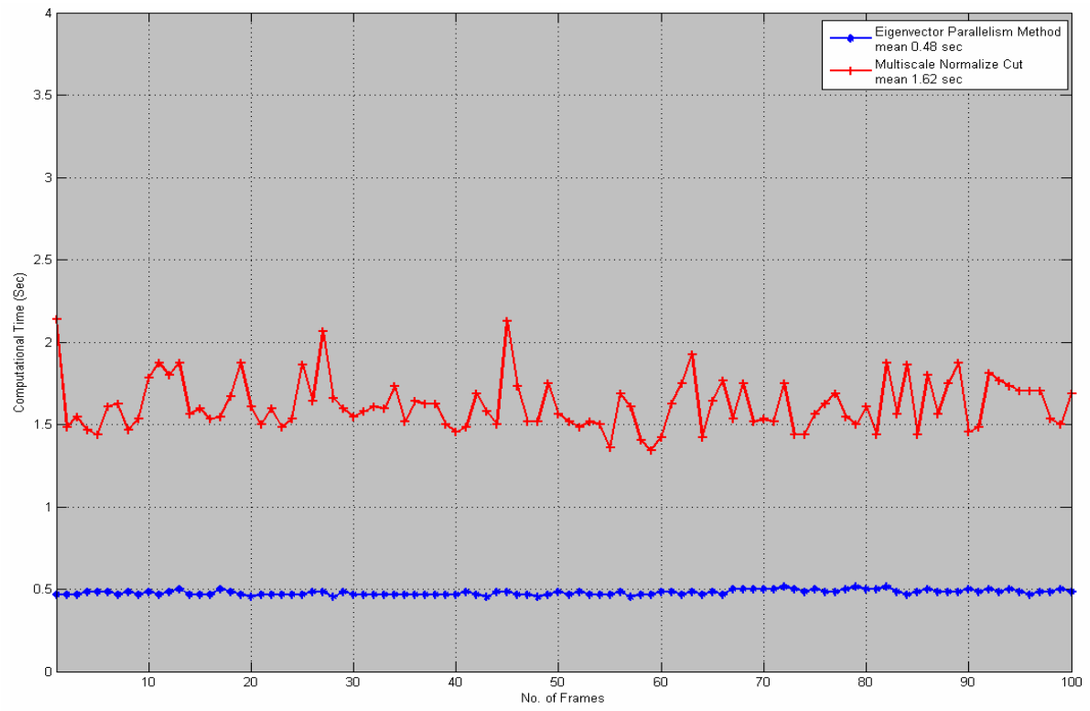


Fig. 16. Comparison graph plot between two methods, Normalize Cut algorithm and Eigenvector Parallelism method (sequence of New York City Marathon with synthetic instability)

their unstable motion. This objective has been achieved by employing a Lagrangian Coherent structure (LCS). The motivation for using LCS method came from the field of fluid mechanics which is able to reveal the spatial organization of the flow field created by the crowd flow. The Finite Time Lyapunov Exponent (FTLE) Field is one of the most appropriate approaches to obtain the LCS. It measures the maximum divergence between two particles as the flow progresses. At first the optical flow was used to give the flow field on which the system launched a particle grid for tracking the advection of crowd flow. These advected of particles are mapped by flow mapping on which spatial gradient is computed. Finally Finite-time Lyapunov Exponent (FTLE) field is computed on this gradient of flow map. The FTLE field restores the boundaries where the crowd flow experiences the dynamic change. On these boundaries, the system employed a new method of eigenvector parallelism in order to identify the precise unstable groups that are propagating in the global flow.

One of the common requirements of the motion detection algorithms in computer vision is the perspective views of camera. In this approach for detection of GOP in a crowded scene, the flow of crowd must be visualized from certain height for the visibility of the spatial organization of the flow. The variability of the image captured will also produce significant noise source to the overall analysis. In this case, stationary camera is used to minimize the capture noise factor. The developed monitoring system successfully tracks a range of objects, which can easily be used for actual monitoring requirements. The method suggested is very robust and can be adapted to various crowded scenarios such as traffic flow and fish schooling, where thousands of objects are involved.

6. References

Ali, S., and Shah, M. (2007) A Lagrangian Particle Dynamics Approach for Crowd Flow Segmentation and Stability Analysis. IEEE Computer Vision and Pattern Recognition, pp. 1-6

Crowd Dynamics Ltd, http://www.crowddynamics.com/Disasters/jamarat_bridge.htm

Haller, G. (2001) Distinguished material surfaces and coherent structures in three-dimensional fluid flows. Physica D. pp. 248-277.

Haller, G. (2005a) Lagrangian coherent structures from approximate velocity data. Physics of fluids, pp. 1851-1861

Haller, G. (2005b) An objective definition of a vortex. J. Fluid Mech. pp. 525, 1-26.

Helbing, D. Johansson, A. HE Habib Z. Al-Abideen. (2007a) Crowd turbulence: the physics of crowd disasters. The Fifth International Conference on Nonlinear Mechanics June 2007. (ICNM-V) Shanghai, pp. 967-969.

Helbing, D. Johansson, A. and H. Z. Al-Abideen. (2007b) The Dynamics of Crowd Disasters: An Empirical Study. Phys. Rev. E 75, 046109.

LEE Ris S. C. and HUGHES Roger L. (2005) Exploring trampling and crushing in a crowd. Journal of transportation engineering vol. 131, no.8, pp. 575-582.

Lucas, B. D. and Kanade, T. (1981) An iterative image registration technique with an application to stereo vision, Proceedings of imaging understanding workshop, pp. 121-130.

Shadden, S.C. Lekien, F. and Marsden, J. E. (2005) Definition and properties of Lagrangian coherent structures from finite-time Lyapunov exponents in two-dimensional aperiodic flows. Physica D. pp. 271-304.

Shadden, S. C. (2006) A dynamical systems approach to unsteady systems. PhD thesis, California Institute of Technology, USA

Shi, J. and Malik, J. (2000) Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 888-905

IntechOpen



New Trends in Technologies: Control, Management, Computational Intelligence and Network Systems
Edited by Meng Joo Er

ISBN 978-953-307-213-5
Hard cover, 438 pages
Publisher Sciyo
Published online 02, November, 2010
Published in print edition November, 2010

The grandest accomplishments of engineering took place in the twentieth century. The widespread development and distribution of electricity and clean water, automobiles and airplanes, radio and television, spacecraft and lasers, antibiotics and medical imaging, computers and the Internet are just some of the highlights from a century in which engineering revolutionized and improved virtually every aspect of human life. In this book, the authors provide a glimpse of the new trends of technologies pertaining to control, management, computational intelligence and network systems.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Mohd Rizal Arshad and Umair Soori (2010). Group Tracking Algorithm for Crowded Scene, New Trends in Technologies: Control, Management, Computational Intelligence and Network Systems, Meng Joo Er (Ed.), ISBN: 978-953-307-213-5, InTech, Available from: http://www.intechopen.com/books/new-trends-in-technologies--control--management--computational-intelligence-and-network-systems/group-tracking-algorithm-for-crowded-scene

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen