A Proposed Data Stream Clustering Method for Detecting Anomaly Events in Crowd Scene Surveillance videos

Abdulamir A. Karim [1] and Narjis Mezaal Shati [2]

[1]University of Technology - Department of Computer Science
[2]University of Al-mustansiryah - College of Science - Department of Computer Science

[1] Ammeer_aldelphi@yahoo.com
[2]Nareen_manar@yahoo.com

Abstract

In this research, a new data stream clustering method utilizing seed based region growing technique is implemented to perform abnormal event detection in anomaly detection system in a new data stream clustering method used in abnormal detection system. This is done by applying HARRIS or FAST detectors on the frames of video clips in two publically available datasets. The first UCSD pedestrian dataset (ped1 and ped2 datasets), and the second VIRAT video dataset system to extract list of pairs of interest points. From these pairs a list of features such as: distance, direction, x-coordinate, y-coordinate obtained to use as an input to the new clustering method. This method in using HARRIS detector achieves detection rates about (9.09%, 52.17%, 61.67%), and the false alarm rates are (18.79%, 36.09%, 66.67%) by using Ped1, Ped2, and VIRAT datasets respectively. For the case of using FAST detector, the best-achieved detection rates are (7.88%, 46.09%, 58.33%), and the false alarms are (21.21%, 40.87%, 63.33%) by using the three previously mentioned benchmarks respectively.

Key words: Surveillance Systems, Anomaly Detection, Crowed Scene Detection,

Anomaly Events, Abnormal Event Detection.

A Proposed Data Stream Clustering Method for Detecting Anomaly

Events in Crowd Scene Surveillance videos

Abdulamir A. Karim and Narjis Mezaal Shati

طريقة جديدة لعنقدة دفق البيانات لتمييز الحالات الشاذة للمشاهد المزدحمة في فديوات المراقبة

عبد الأمير عبد الله كريم1 و نرجس مزعل شاتي2

1الجامعة التكنولوجية ـ قسم علوم الحاسوب

2الجامعة المستنصرية ـ كلية العلوم ـ قسم علوم الحاسوب

الخلاصة

في هذا البحث نفذت طريقة مقترحة في عنقدة دفق البيانات موظفا فيها تقنية الـــ (seed base region growing) لانجاز مهمة تمييز الحالات الشـاذة في نظام كشف الحالات الشـاذة وهذا يتم بتطبيق (HARRIS detectors) او الـــ FAST detectors) على الـــ (frames) التي تم الحصول عليها من فديوات الـــ (datasets) المعروفة عالميا وهي (UCSD pedestrian) وتتكون من قاعدتين(ped1) و(ped2) واما الثانية فهي (VIRAT). ومن هذه الـ (frames) يتم الحصول على مجموعة من ازواج نقاط الاهتمام. ومن هذه النقاط يتم الحصول على مجموعه من الـــخصائص مثل المسافة والاتجاة واحداثيات المحور والسـيني والصـادي وذلك لإدخالها كمدخل لطريقة العنقدة الجديدة. عند اسـتخدام الـــ (HARRIS detectors) بهذه الطريقة كانت نسـبة التمييز (9,09%، 52,17%، 61,67%)، ونسـبة الخطأ في التمييز (18,79%، 36,09، 66,67 %) باستخدام فديوات الـــ Ped1، Ped2، VIRAT وعلى التتابع. أما في حالة استخدام الـــ (FAST detectors) فأن افضـل نسـبة تمييز كانت (7,88%، 46,09%، 58.33 %) ونسـبة الخطأ في التمييز كانت (21,21 % ،40,87%، 63,33%) في الـ (benchmarks datasets) المذكورة سابقا.

الكلمات المفتاحية: أنظمة المراقبة، تمييز الحالات الشـاذة، تمييز المشـاهد المزدحمة، الحالات الشـاذة، تمييز الحالات غير الطبيعية.

Introduction

The increase in the number of cameras in urban areas like the advent of very high capacity digital video recorders (DVRs), the development of Internet protocol (IP) networks (IP cameras), and the technological advances in the protection of critical infrastructures lead to solve the problem of the incapability to exploit video streams in real time for the purposes of detection or anticipation. It involves having the videos analyzed by algorithms that detect and track objects of interest (usually people or vehicles) over time, and that indicate the presence of

events or suspect behavior involving these objects. The aim is to be able to alert operators in suspicious situations in real time, this called video content analysis but we may also come across the terms video analytics, intelligent video surveillance or smart video surveillance [1].

More and more surveillance cameras are being installed in different locations, including banks, government premises, railway stations, and houses, this led to development of technologies and systems in the area of intelligent surveillance in a variety of public or private areas for the sake of security and safety. Some related researches to investigate individual behavior detection are presented in [2, 3, 4] other investigating detection abnormal behavior in crowed action in [5, 6, 7, 8].

Detecting of Anomaly Events

Consequently, detecting unusual or suspicious activities, uncommon behaviors, or irregular events in a scene is the primary objective of an automated video surveillance system. Refer to this activity as anomaly detection because the sought-after situations are not observed frequently. The working definition of this term is taken to be the spatio-temporal compositions in a video or set of videos with low probability of occurrence with respect to the previous observations. The anomalies with respect to a pixel's context, means that a particular activity in a particular context would be an anomaly, while in another context it might be normal [5].

1. Detecting of Dominant and Rare Event Simultaneously

The dominant behaviors refer to the normal events observed in a scene. These are events that have a higher probability of occurrence than others in the video and generally do not attract much attention. These dominant behaviors are categorized into two classes. The first deals with foreground activities and the second describes the scene background. The second is referred to as background subtraction, but is more general and more complicated than background subtraction, because it includes the scene background while not being limited to it, that learn both normal and abnormal patterns [5].

2. Features of Human Behavior

The information provided by surveillance video feeds represented in various human behavior representations, such as commonly occupied region, human trajectory shape and bag-of-words techniques (code book representation). The first and second are used when the camera is far from the subject, but the last is suitable when human limb movements are observable and also referred to as interest points based methods [9, 10,11].

Human behavior features are divided into two major parts: Human global motion and Human local motion features. Human Global Motion features utilize the information on a subject location at various times. It is useful when human limb movements are not observable. The systems using this features considers only the tracking information of each subject's locations at a time. There is a lot of information extracted from human trajectories, such as (Commonly occupied region, Human walking path shape) [12].

Human Local Motion features are useful when human limb movements can be discerned in video feeds. Based on the degree of not-rigidity of the objects, human motion are classified as rigid or not rigid motions. Also there is articulated motion in which limb motions are rigid but overall motion is not rigid [13].

Categorization of human motion approaches is based on with/ without prior knowledge about the object shape. These category listed as follows:

- Model based approaches: Using prior knowledge about the shape of an objects [14].
- Appearance based approaches: Building body representation in a bottom-up fashion by first detecting appropriate features in a sequence of frames. Silhouettes and interest points are some examples of these features. These approaches do not require a specific object model but are sensitive to noise [15, 16]. Appearance based approaches are classified into three types: *Flow based approaches, Spatial-temporal shape template base* approach, and *Interest point based* approaches

The interest points extracted by interest points detectors. It is extension of key point's concept for object detection in images. One of the advantage of using interest points is that they can be

used to search an action contained in the short query video over a large resolution without using background subtraction and tracking of the object [16,17].

3. Interest Points Features

Interest points (salient points) are the local spatio-temporal features in a video segment that is detected by using an approach to detecting and matching these points in the video segment in both the temporal and spatial domains. It is detected when there is a region producing high response values. The interest points features are formed by the interest points and their surrounding neighborhoods. So it can be able to describe the action captured in a video segment [18]. In this paper interest points features are used by utilizing HARRIS or FAST detector, for more information about these detectors see [19,20].

4. Data Stream Clustering and Seed Based Region Growing Technique

A data stream model is defined as an ordered sequence of points x1, … , xn where ($n \approx \infty$). The sequence has to be read in order and once or a small number of times [21].

The data stream model requires decisions to be made before all the data becomes available. This model is similar to online models. So these models need algorithm (data stream algorithm). These algorithms are allowed to take action after a group of points arrives [21].

Seed Based Region Growing method is Segmentation technique. The basic idea in this method is to group a collection of pixels to form a region based on some similar properties; these properties may include intensity, texture or color, as shown in figure (1) [22,23].
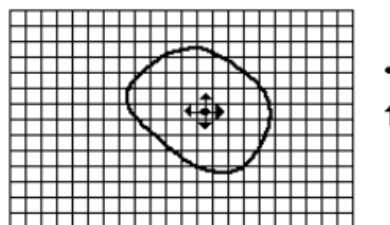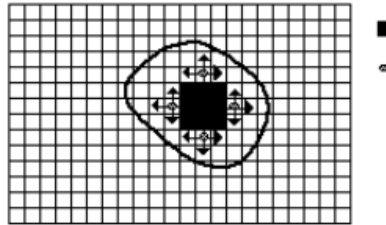

(a) Seed Pixel Direction of Growth

(b) Grown Pixels Pixels Being Consider

Figure (1): Example of region growing. (a)Start of Growing a Region (b) Growing Process after a Few Iterations

Data stream and online clustering approaches are similar in that both of them require decisions made before all data are available. But these models are not identical because online algorithm can access the first i data point (with its previous i decisions) when reacting to the $(i+1)^{th}$ point, the amount of memory available to data stream algorithm is bounded by a function of the input size (sublinear function used). Also, a data stream approach not be required to take action after the arrival of each point (after a group of points) [18].

Data stream clustering approaches store and processes large scale data efficiently because it provides summarizations of the past data, see [24, 25, 26, 27, 28, 29] for more information. In this work seed filling clustering algorithm is utilized to form new data stream algorithm.

Design of Data Stream Clustering Algorithm Based on Seed Filling Technique

Anomaly behavior detection systems data stream clustering algorithm based on seed filling technique are created were trained and applied to the same datasets. Their performance were measured using (Detection Rate (DR), False Alarm Rate (FAR), Recall (R), Precision (P), The Coverage Test (CT)), for more detail about these measurements see [30, 31].

1. The Proposed Systems Layouts

The diagram for the proposed anomaly detection system is shown in figure (2).

2. Video Dataset

The selected video datasets are two publicly available datasets. The first is the UCSD dataset: pedestrians' datasets (ped1) and (ped2); and the second is VARAT dataset. The entire set of the selected video dataset is divided into two sets of videos: (i) a training video set is used for the developments of a system, and (ii) a test video set contains videos used to measure the anomaly behavior detection performance of the system.

3. Preprocessing

Converting to gray preprocessing applied on both training and test datasets to make the data of both training and test sets more suitable and easier to analyses. Preprocessing is a necessary stage when the requirements are typically obvious and simple. In this step the frames of video dataset are converted to gray image: basically each pixel in extracted color sub-image has three components of color (red, green, blue), the value of each color component is represented by one byte. The gray value to each pixel is computed by using average method and the process of computing values of all pixels leads to convert the color frame to gray frame.

4. Feature Generation

From the training and test video clips of dataset an interest points information's such as: interest points pairs and (distance, direction, x-coordinate, y-coordinate) of these pairs are extracted. This done in feature detection and feature matching phases by utilizing interest points detectors HARRIS or FAST to extract interest points from the current frame and previous frame (according to predefined threshold PFT) and then matching these interest points from both frames to obtains list of pairs of interest points. The process illustrated in (1, 2) algorithms. After that an extraction process will be done to estimate list of features (distance, direction, x-coordinate, y-coordinate) from these pairs, see figure (3). This illustrated in algorithm (3).

A Proposed Data Stream Clustering Method for Detecting Anomaly

Events in Crowd Scene Surveillance videos

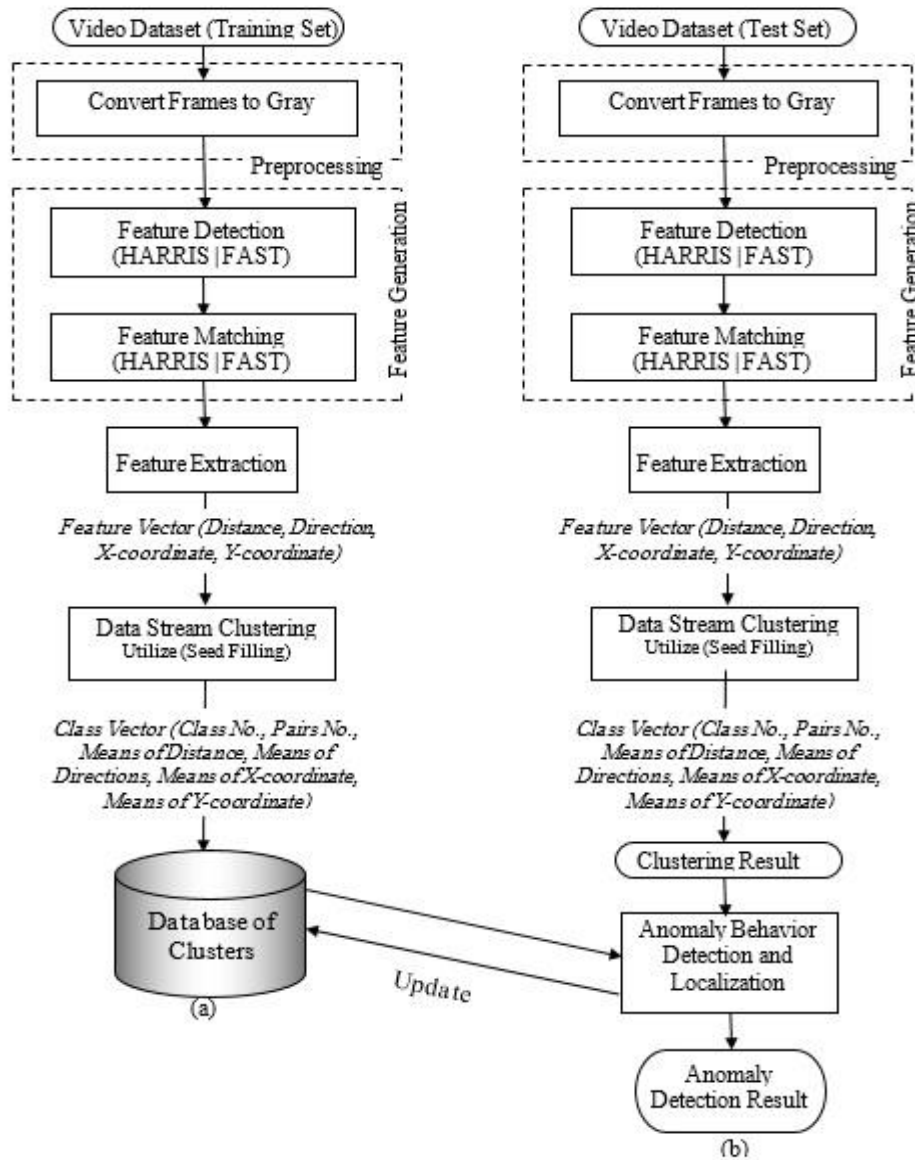**Abdulamir A. Karim and Narjis Mezaal Shati**

Figure (2) Diagram of the proposed systems: (a) Enrolment phase (b) Anomaly

detection and localization phase

Algorithm (1)// Detection and matching interest points using HARRIS
Input: Frames_List, PFT (Previous Frame Threshold), No. of Frames **Output:** LOP (List of interest points pairs)

For n = 0 **to** Frame_List.count
　　Detect interest points using HARRIS corner: Detector (Frames_List [n], Interest_Points_List [n])
　　If n > PFT **then** Matching (Frames_List [n], Frames_List [n‑PFT], Interest_Points_List [n],
　　　　　　　　　　Interest_Points_List [n‑PFT]), LOP)
　　　For i = 0 **to** LOP.count
　　　　Compute (LOP[i].distance) using Euclidean distance between the two points of the pair (LOP[i])
　　　　If LOP[i].distance < Low_Threshold or LOP[i].distance > Max_Threshold
　　　　Then Remove (LOP[i])
　　　　End If
　　　End For
　　End If
　End For
End.

Algorithm (2)// Detection and matching interest points using FAST
Input: Frames_List, PFT (Previous Frame Threshold), No. of Frames **Output:** LOP (List of interest points pairs)

For n = 0 **to** Frame_List.count
　　Detect interest points using FAST corner: Detector (Frames_List [n], Interest_Points_List [n])
　　If n > PFT **then** Matching (Frames_List [n], Frames_List [n‑PFT], Interest_Points_List [n], Interest_Points_List [n‑PFT]), LOP)
　　　For i = 0 **to** LOP.count
　　　　Compute (LOP[i].distance) using Euclidean distance between the two points of the pair (LOP[i])
　　　　If LOP[i].distance < Low_Threshold or LOP[i].distance > Max_Threshold
　　　　Then Remove (LOP[i])
　　　　End If
　　　End For
　　End If
　End For
End.
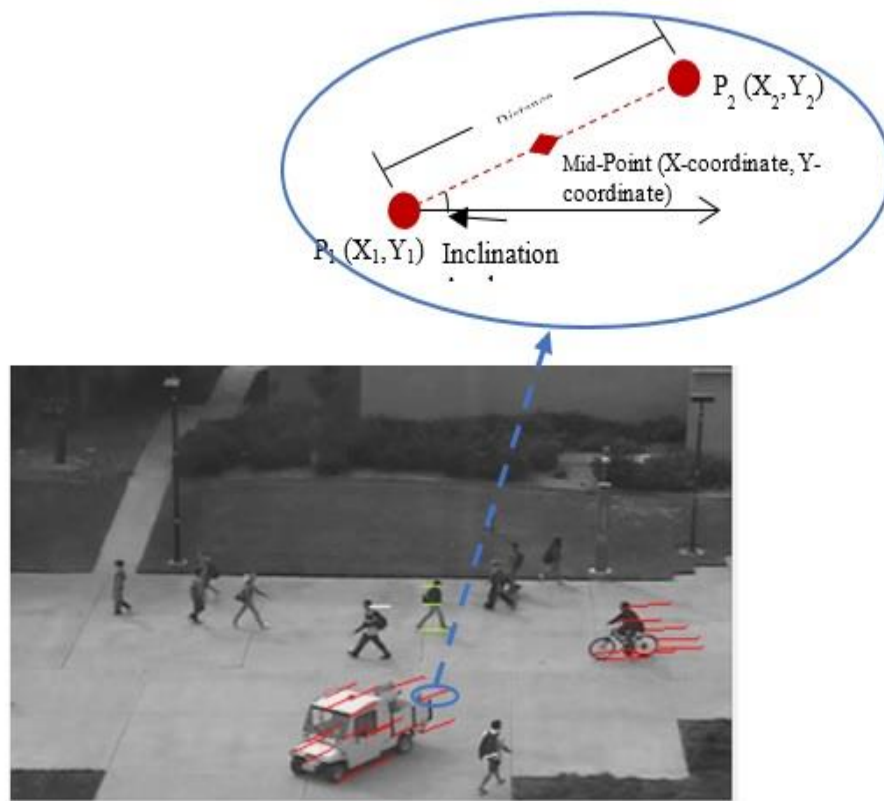
Figure (3) Illustration of distance, inclination angle, x-coordinate, and y-coordinate for two points (P₁ and P₂)

Euclidean distance used to compute the distance between the two points (P1(x1,y1) and P2(x2,y2)) of each pair in list of pairs. Then calculate direction by finding the angle of inclination, as shown in the following equation and illustrated in algorithm (4).

$$Dir = \begin{cases} 0 & if\ (x_1-x_2>0)\ and\ (y_1-y_2=0) \\ 90 & if\ (x_1-x_2=0)\ and\ (y_1-y_2>0) \\ 180 & if\ (x_1-x_2<0)\ and\ (y_1-y_2=0) \\ 270 & if\ (x_1-x_2=0)\ and\ (y_1-y_2<0) \\ \tan^{-1}(y_1-y_2)/(x_1-x_2) & Otherwize \end{cases} \quad \dots\dots\dots\dots\dots (1)$$

After that, compute the x-coordinate and y-coordinate for the pair by finding the mid-point between the first and the second point of each pair, as show in the following equations.

$$x = (x_1 + x_2) / 2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(2)}$$

$$y = (y_1 + y_2) / 2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(3)}$$

Algorithm (3)// Compute distance, direction, x-coordinate, y-coordinate from interest points pairs
Input: List of interest points pairs (LOP)   **Output:** LOF (List of feature vector (distance, direction, x-coordinate, y-coordinate))

For i=0 **to** LOP.count
     Compute (LOF[i].dis) using Euclidean distance between $P_1(x_1, y_1)$ and $P_2 (x_2, y_2)$ in one pair
     Compute (Direction) using algorithm (4),
     Compute (x-coordinate, y-coordinate), **Set** LOF[i].x ← (LOP[i,1].x + LOP[i,2].x)/2
                                    Set LOF[i].y ← (LOP[i,1].y + LOP[i,2].y)/2
   End For
 End.

Algorithm (4)// Compute direction
Input: two interest points $P_1 (x_1, y_1)$ & $P_2 (x_2, y_2)$ **Output:** D (Direction)

If $(x_1 - x_2) = 0$ **then**
           If $(y_1 - y_2) > 0$ **then Set** D ← 90
           Else If $(y_1 - y_2) < 0$ **then Set** D ← 270
        Else If $(y_1 - y_2) = 0$ **then**
           If $(x_1 - x_2) > 0$ **then Set** D ← 0
           Else Set D ← 180
        Else Set D ← $\tan^{-1}(y_1 - y_2)/(x_1 - x_2)$
        End If
 End.

5. Proposed Data Stream Clustering

In this paper a data stream model is defined as four features of each pairs of interest points which are obtained from the detection and matching feature phase. The proposed algorithm is used to make decision after a group of these features arrived and then these decisions (past data of clusters) are summarized and organized in a database; that gives a summarization of all clusters in limited memory size. This system utilizes seed based region growing technique as

seen in the following steps that illustrates the implemented steps for propose data stream clustering.

Second Proposed Data Stream Clustering System Algorithm Steps

Step₁: Read Video.

Step₂: Convert color frame to gray frame.

Step₃: Detection and matching interest points using (HARRIS | FAST) by using algorithms (1, 2) respectively.

Step₄: Compute distance, direction, x-coordinate, y-coordinate from interest points pairs using algorithm (3).

Step₅: Clustering list of features by algorithm (5) using Seed filling.

Step₆: If training phase **then** Create Database_Clusters using algorithm (8)

   Else Detect and localize anomaly behavior using algorithm (9)

   End If

Starting with reading video clips from datasets (the training videos in training phase) or (testing videos in testing phase) to obtain list of video frames and number of frames, continuing by converting these frames to gray frames. And then extract list of interest pairs by applying interest points detector (HARRIS or FAST) as shown in (1 or 2) algorithms. After that extract list of interest points features such as: distance, direction, x-coordinate, y-coordinate, using (3) algorithm. The next step is clustering these extracted list of interest points features by investment seed based region growing technique in data stream clustering algorithm, this is done based on coordinates, as seen in (5) algorithm. In algorithm (5) test all pairs in list of pairs will be done, if the pair is not classified under any cluster, then it will be classified as new cluster and added to the list of classified neighbor pairs to be tested through computing the closest neighbor to it in list of pairs by using Euclidian distance, as shown in algorithm (6),

where the coordinate threshold is used to determine the neighbor when distance is less than it. And then, test all neighbor pairs if it is not classified to any cluster then it will be classified as the cluster of the tested pair and added to the classified neighbor pairs list to examine the neighbor pairs for it. After completing the test of the neighbor pairs list and there is more than one pair in the set a computation of new centroids for these neighbor pairs will be done and these new centroids are added to the list of centroids otherwise will be discarded and removed from the neighbor pairs list. The computation of new centroids will be done by allocating minx, miny, maxx, maxy for each group of neighbor pairs in the same cluster to determine the coordinates of the new cluster, also the average of distance and direction will be computed, as shown in (7) algorithm. Figure (4) illustrates the process of the second proposed data stream clustering algorithm based on coordinates.
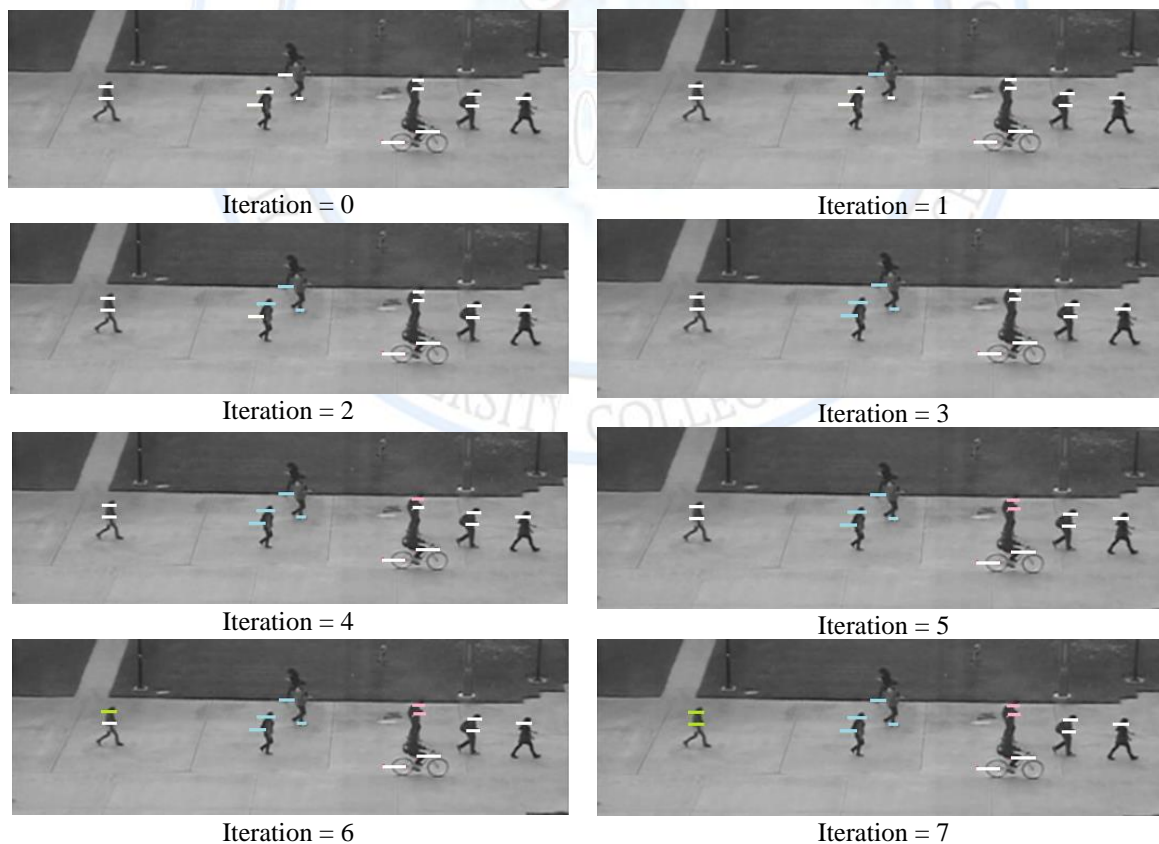
Iteration = 0	Iteration = 1
Iteration = 2	Iteration = 3
Iteration = 4	Iteration = 5
Iteration = 6	Iteration = 7

A Proposed Data Stream Clustering Method for Detecting Anomaly

Events in Crowd Scene Surveillance videos

Abdulamir A. Karim and Narjis Mezaal Shati

Iteration = 8	Iteration = 9
Iteration = 10	Iteration = 11
Iteration = 12	Iteration = 13
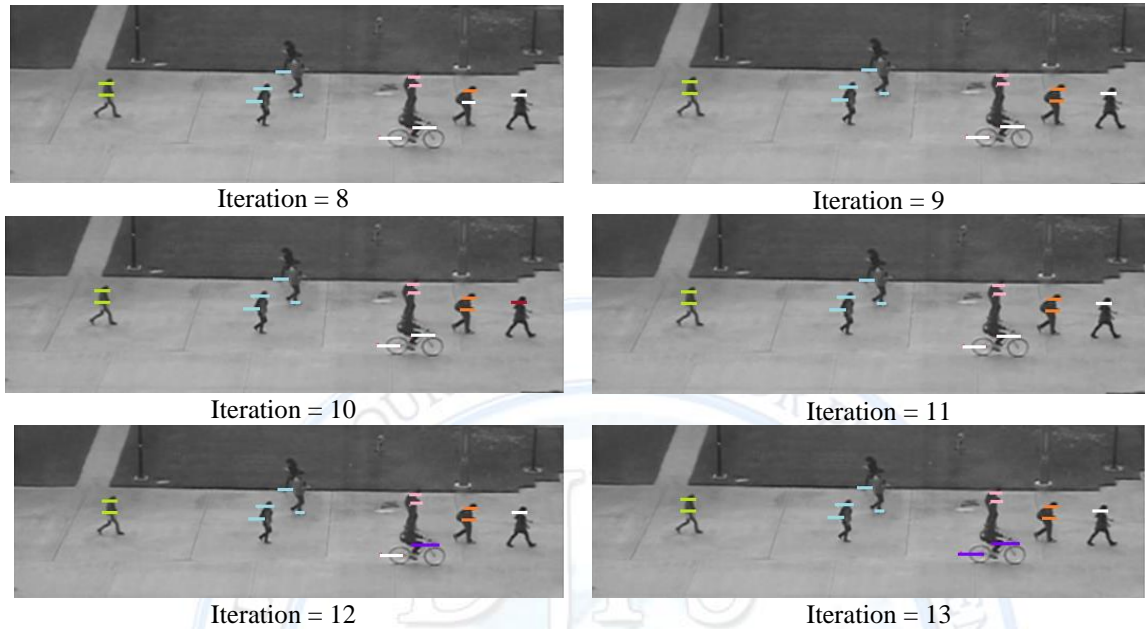
Figure (4): Illustrates proposed data stream clustering based on coordinates in various iterations, resulting in five clusters.

Algorithm (5)// Clustering list of features using Seed Filling technique based on coordinate
Input: LOF
Output: LOF, Centroids_list

For i = 0 **to** LOF.count
   If LOF[i] is not classified **then**
      Set LOF[i] as a new cluster,
      Add LOF[i] to Cluster_List
      do
         For j = 0 **to** Cluster_List.count
           Compute closest neighbor for each Cluster_List [j] in (LOF) using Euclidean distance
           implemented by algorithm (6).
         If neighbor of Cluster_List [j] is not classified **then**
           Set neighbor (Cluster_List [j]). cluster ← Cluster_List [j]. cluster
           Add neighbor (Cluster_List [j]) to Cluster_List
         End If
        End for
     End while (listing all Cluster_List piars)
   If (Cluster_List.length > 1) **then**
      Calculate new centroids (dis+dir+cords+start+end) using algorithm (7)
      Add this centroid to centerids_list

```
            Else
                Set LOF[i] as not classified
                Empty Cluster_List
            End if
        End if
    End for
End.
```

Algorithm (6)// Compute closest neighbor
Input: LOF
Output: Neighbor_List
``` For i = 0 to LOF.count   For j=0 to LOF.count   If (LOF[i].dis - LOF[j].dis) < dis_thr and (LOF[i].dir - LOF[j].dir) < dir_thr and (Euclidean   distance between (LOF[i].coords, LOF[j]).coords) < dis_thr   Then   Set neighbor_List [i,j] ← true   End if   End for End for End. ```

Algorithm (7)// Calculate new centroids
Input: Cluster_List
Output: One centroid
``` For each pair in Cluster_List   If (minₓ > pair.start.x) than   Set minₓ ← pair.start.x   If (min_y > pair.start.y) than   Set min_y ← pair.start.y   If (maxₓ < pair.end.x) than   Set maxₓ ← pair.end.x   If (max_y < pair.end.y) than   Set max_y ← pair.end.y   Set Dis ← dis + pair.dis   Set Dir ← dir + pair.dir End for Set Centroids.dis ← dis/centroids.count Set Centroids.dir ← dir/centroids.count Set Centroids.coords.x ← (minₓ+maxₓ)/2 Set Centroids.coords.y ← (min_y+max_y)/2 Set Centroids.start.x ← minₓ Set Centroids.start.y ← min_y Set Centroids.end.x ← maxₓ Set Centroids.end.y ← max_y End. ```

Algorithm (7)// Calculate new centroids

Input: Cluster_List
Output: One centroid

```
For each pair in Cluster_List
    If (min_x > pair.start.x) than    Set min_x ← pair.start.x
    If (min_y > pair.start.y) than    Set min_y ← pair.start.y
    If (max_x < pair.end.x) than      Set max_x ← pair.end.x
    If (max_y < pair.end.y) than      Set max_y ← pair.end.y
    Set Dis ← dis + pair.dis
    Set Dir ← dir + pair.dir
End for
Set Centroids.dis ← dis/centroids.count
Set Centroids.dir ← dir/centroids.count
Set Centroids.coords.x ← (min_x+max_x)/2
Set Centroids.coords.y ← (min_y+max_y)/2
Set Centroids.start.x ← min_x
Set Centroids.start.y ← min_y
Set Centroids.end.x ← max_x
Set Centroids.end.y ← max_y
End.
```

In enrolment phase the cluster database will be created in the clustering process, see algorithm (8). While in the anomaly detection and localization phase, the anomaly event and its location will be detect with help of previously database created.

Algorithm (8)// Training
Input: Centroids_list
Output: Database_Clusters (database of clusters)

For i = 0 **to** Centroids_list.count
    If record in Database_Clusters where record is equal to Centroids_list [i]
    Then **Set** Record. Frequnce ← Record. Frequnce+1
    Else Database_Clusters.add (Centroids_list [i])
    End If
  End For
End.

6. Anomaly Behavior Detection and Localization

After completing the construction of cluster database from passing the training video dataset, it will be used in anomaly behavior detection in order to detect anomaly behavior in test video dataset and localize its position; algorithm (9) describe the anomaly behavior detection and localization. In this algorithm, the class obtained from the proposed data stream clustering algorithm in test phase and then search the database of clusters; that is obtained from the training phase; to find if this class is in the database, so it will be detected as anomaly event and its location in the frame is marked by using the coordinate from the clustering data stream process. Figure (5) shows anomaly detection in some frames.

Algorithm (9)// Anomaly behavior detection and localization
Input: Test video dataset, Database_Clusters (database of clusters)
Output: Anomaly_behavior_location

For i=0 **to** centroids_list.count
    If record in Database_Clusters where record equal to centroids_list[i] **then**
      Set Record.Frequnce ← Record. Frequnce +1
    Else Mark the position of anomaly behavior
    End If
  End For
End.

Figure (5) Sample of frames illustrating anomaly detection

Test Result

The proposed approach is tested on two datasets. The UCSD pedestrian dataset it contains video sequences from two pedestrian (Ped1 and Ped2) walkways where abnormal events occur. The dataset contains different crowd densities, and the anomalous patterns are the presence of non-pedestrians on a walkway (bicyclists, skaters, small carts, and people in wheelchairs). The second dataset is VIRAT Video Dataset was designed by Kitware Company to be more realistic, natural and challenging for video surveillance domains than existing action recognition datasets in terms of its resolution, background clutter, diversity in scenes, and human activity/event categories. Data are collected in natural scenes showing people performing normal actions in standard contexts, with uncontrolled, cluttered backgrounds. The key characteristics of these datasets are summarized in table (1).

Table (1) Key characteristics of training and testing video clips in datasets

			UCSD (Ped1)	UCSD (Ped2)	VIRAT
Entire dataset	training	Number of video clips	34	16	6
		Frame Dimension	238×158	360×240	426×240
		Frame rate	15 frames/second	15 frames/second	15 frames/second
	test	Number of video clips	36	12	8
		Frame Dimension	238×158	360×240	426×240
		Frame rate	15 frames/second	15 frames/second	15 frames/second
Anomaly Behavior Event	test	Number of anomaly events	1460	987	335

The experiment was repeated 15 times, and the values of the used data stream clustering parameters shown in table (2). Table (3) illustrate the anomaly detection test results of UCSD (Ped1) video clips sets using features extracted from the interest points pairs as an input to the anomaly behavior detection system in both method of feature extraction using HARRIS or FAST detectors and the detail of the detection system are presented in tables (4, 5) using UCSD (ped2) dataset and VIRAT dataset respectively.

Table (2) The proposed data stream clustering parameters.

Parameter	Value
Previous frame threshold (PFT)	3
Distance threshold	2
Direction threshold	1
Coordinate threshold	25

Table (3) Results of the anomaly behavior detection system by using UCSD (Ped1) dataset (PFT=5; Distance threshold=2; Direction threshold=1; Coordinate threshold=25; Repetition=15)

	Expt.	Training Time (ms/ frame)	Testing Time (ms/ frame)	DR (%)	FAR (%)	R	P	CT
HARRIS	Min	2.18	2.91	8.48	20.00	0.08	0.30	0.13
HARRIS	Max	2.18	3.02	9.70	18.18	0.10	0.35	0.15
HARRIS	Average	2.18	3.11	9.09	18.79	0.09	0.33	0.14
FAST	Min	2.43	2.84	7.27	23.64	0.07	0.24	0.11
FAST	Max	2.43	3.11	9.09	23.03	0.09	0.28	0.14
FAST	Average	2.43	2.40	7.88	21.21	0.08	0.27	0.12

Table (4) Results of the anomaly behavior detection system by using UCSD (Ped2) dataset (PFT=3; Distance threshold=2; Direction threshold=1; Coordinate threshold=25; Repetition=15)

|	Expt.	Training Time (ms/ frame)	Testing Time (ms/ frame)	DR (%)	FAR (%)	R	P	CT
HARRIS	Min	2.97	3.31	48.70	35.65	0.49	0.58	0.53
HARRIS	Max	2.97	3.44	54.35	34.78	0.54	0.61	0.57
HARRIS	Average	2.97	3.12	52.17	36.09	0.52	0.59	0.55
FAST	Min	2.43	2.72	44.35	39.13	0.44	0.53	0.48
FAST	Max	2.43	3.14	47.83	40.00	0.48	0.54	0.51
FAST	Average	2.43	2.67	46.09	40.87	0.46	0.53	0.49

Table (5) Results of the anomaly behavior detection system by using VIRAT dataset (PFT=3; Distance threshold=2; Direction threshold=1; Coordinate threshold=25; Repetition=15)

|	Expt.	Training Time (ms/ frame)	Testing Time (ms/ frame)	DR (%)	FAR (%)	R	P	CT
HARRIS	Min	1.98	2.02	60.00	63.33	0.60	0.49	0.54
HARRIS	Max	1.98	2.36	65.00	65.00	0.65	0.50	0.57
HARRIS	Average	1.98	2.54	61.67	66.67	0.62	0.48	0.54
FAST	Min	2.87	3.38	55.00	61.67	0.55	0.47	0.51
FAST	Max	2.87	3.16	60.00	65.00	0.60	0.48	0.53
FAST	Average	2.87	3.42	58.33	63.33	0.58	0.48	0.53

A Proposed Data Stream Clustering Method for Detecting Anomaly

Events in Crowd Scene Surveillance videos

Abdulamir A. Karim and Narjis Mezaal Shati

The test results show that the proposed anomaly detection system with HARRIS detector is capable of recognizing the anomaly event with average detection rate (9.09%, 52.17%, 61.67%) for the test video clips set from the three dataset respectively, the average false alarm rate was (18.79%, 36.09%, 66.67%). The obtained detection results are shown in figures (4.15). And with DRs= (7.88%, 46.09%, 58.33%) , FARs= (21.21%, 40.87%, 63.33%) for the three dataset respectively. figures (6, 7) illustrate the obtained R,P,CT.
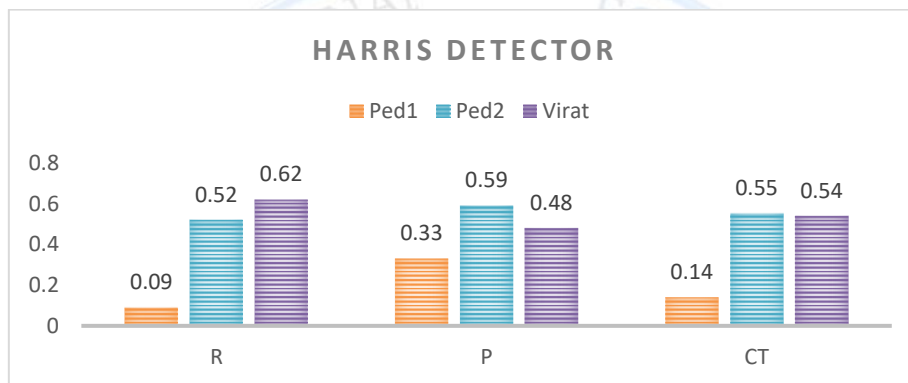
Figure (6): The bar chart result of proposed anomaly detection system using HARRIS detector
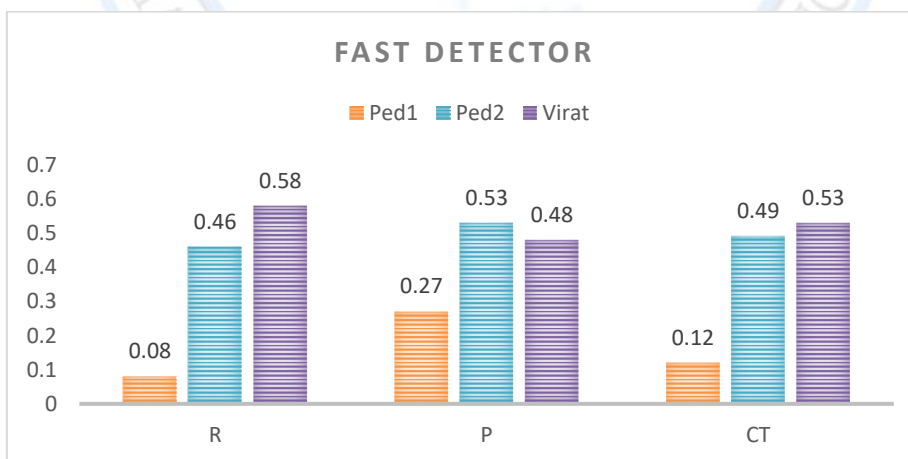
Figure (7): The bar chart result of proposed anomaly detection system using FAST detector

Conclusions

A proposed method (utilizing seed filling technique in data stream clustering algorithm) for automatic detection of anomaly events system was proposed in this paper. This approach have lower computational complexity and fewer demands of dynamic parameter adjustment. The performance of the suggested approach improved when the features extracted by using interest points pairs from HARRS detector. Also the result of this method suffer from the stability problem of the result from data set to other. In ped1 dataset the view of camera effect the result that indicate poor detection rate. In ped2 dataset the detection rate improved but with amount of false alarm rates. Finally in VIRAT dataset give good detection rate but also with high false alarms. Further work is needed to increase the detection rate and decrease the false alarm rate.

References

1. Dufour, J. Y.; "Intelligent video surveillance systems"; ISTE and WILEY; ISBN:978-1-84821-433-0; 2013.
2. Castro, J. L.; and Et al.; "Intelligent surveillance system with integration of heterogeneous information for intrusion detection"; Expert Systems with Applications; ELSEVIER; Vol. 38; PP.11182–11192; 2011.
3. Tang, Y.; Ma, B.; and Yan, H.; "Intelligent Video Surveillance System for Elderly People Living Alone Based on ODVS"; Advances in Internet of Things; SciRes; Vol. 3; PP. 44-52; 2013.
4. Khodabandeh, M.; and Et al.; "Discovering Human Interactions in Videos with Limited Data Labeling"; IEEE; 978-1-4673-6759; 2015.
5. Roshtkhari, M. J.; "Visual Event Description in Videos"; PhD. Thesis; McGill University; Faculty of Graduate Studies and Research; Department of Electrical and Computer Engineering; Montreal; 2014.
6. Lee, D.G.; and Et al.; "Motion influence map for unusual human activity detection and localization in crowded scenes"; IEEE; 1051-8215; 2015.
7. Biswas, S.; and Gupta, V.; " Abnormality detection in crowd videos by tracking sparse components"; Springer; Machine Vision and Applications; 10.1007/s00138-016-0800-8; Verlag Berlin Heidelberg; 2016.

8. Al-Nawashi, M.; Al-Hazaimeh, O. M.; and Saraee, M.; " A novel framework for intelligent surveillance system based on abnormal human activity detection in academic environments"; Neural Comput & Applic ; Springer; 2016.

9. Bregonzio, M.; Gong, S.; and Xiang, T.; "Recognising action as clouds of space-time interest points"; in IEEE Conference on Computer Vision and Pattern Recognition, CVPR; USA; pp. 1948-1955; 2009.

10. Zhang, Z.; and Et al.; "Motion context: A new representation for human action recognition"; in Computer Vision, ECCV; pp. 817-829; 2008.

11. Niebles, J.; Wang, H.; and Fei-Fei, L.; "Unsupervised learning of human action categories using spatial-temporal words"; International Journal of Computer Vision; Vol. 79; No. 3; pp. 299-318; 2008.

12. Morris, B. T., and Trivedi, M.M.; "A survey of vision based trajectory learning and analysis for surveillance"; 2008.

13. Aggarwal, J. K.; and Sangho, P.; "Human motion: modeling and recognition of actions and interactions"; in 3D Data Processing, Visualization and Transmission. 3DPVT. Proceedings. 2nd International Symposium; pp. 640-647; 2004.

14. Niyogi, S. A.; and Adelson, E. H.; "Analyzing and recognizing walking figures in xyt"; technical report, M.I.T. Media Lab Vision and Modeling Group Technical Report; No. 223; 1994.

15. Bobick, A.; and Davis J.; "The recognition of human movement using temporal templates"; Pattern Analysis and Machine Intelligence, IEEE Transactions; Vol. 23; No. 3; pp. 257-267; 2001.

16. Dollar, P.; and Et al.; "Behavior recognition via sparse spatio-temporal features"; in Visual Surveillance and Performance Evaluation of Tracking and Surveillance, IEEE International Workshop; pp. 65-72; 2005.

17. Shechtman, E.; and Irani, M.; "Space-time behavior-based correlation or how to tell if two underlying motion fields are similar without computing them"; Pattern Analysis and Machine Intelligence, IEEE Transactions; Vol. 29; No. 11; pp. 2045-2056; 2007.

18. Wiliem, A.; "Robust suspicious behaviour detection for smart surveillance systems"; PhD. Dissertation; Queensland University of Technology; Faculty of Built Environment and Engineering; School of Engineering System; Brisbane; Australia; 2010.

19. Szeliski, R.; "Computer Vision: Algorithms and Applications"; Springer; 2010.

20. Abood, S.S.; "Construction of real time panoramic video in surveillance system"; M.Sc. Thesis; Department of computer Science; university of technology; 2015.

21. Guha, S.; and Et al.; "Clustering data streams: Theory and practice"; Knowledge and Data Engineering, IEEE Transactions; Vol. 15; No. 3; pp. 515-528; 2003.

22. Matta, S.; "Review: Various image segmentation techniques"; International Journal of Computer Science and Information Technologies (IJCSIT); Vol. 5; Issue 6; pp. 7536-7539; 2014.

23. Langote, V. B.; and Chaudhari, D. S.; "Segmentation techniques for image analysis"; International Journal of Advanced Engineering Research and Studies (IJAERS); Vol. 1; Issue 11; pp. 252-255; 2012.

24. O'Callaghan, L.; and Et al.; "Streaming-data algorithms for high-quality clustering"; in Data Engineering, Proceedings. 18th International Conference; pp. 685-694; 2002.

25. Aggarwal, C. C.; and Et al.; "A framework for clustering evolving data streams"; in the 29th VLDB Conference; Germany; pp. 81-92; 2003.

26. Aggarwal, C. C.; and Yu, P. S.; "A framework for clustering uncertain data streams"; in Data Engineering. ICDE. IEEE 24th International Conference; pp. 150-159; 2008.

27. Cheng, L.; and Et al.; "An online discriminative approach to background subtraction"; In IEEE International Conference on Advanced Video and Signal based Surveillance; pp. 2; 2006.

28. Lühr, S.; and Lazarescu, M.; "Incremental clustering of dynamic data streams using connectivity based representative points"; Data and Knowledge Engineering; Vol. 68; No. 1; pp. 1-27; 2009.

29. Wiliem, A.; and Et al.; "Adaptive unsupervised learning of human actions"; in The 3rd International Conference on Imaging for Crime Detection and Prevention; Kingston University, London; 2009.

30. Shati, N. M.; "2D-objects detection using back propagation neural networks"; MSc. Thesis; Al-Mustansiriah University; College of Science; 2006.

31. Ali, A. E.; "Multi object tracking using machine learning technique"; PhD. Dissertation; University of Technology; Department of Computer Science; 2015.