INTELLIGENT VIDEO SURVEILLANCE SYSTEM
ARCHITECTURE FOR ABNORMAL ACTIVITY DETECTION

M. Elarbi-Boudihir, Khalid A. Al-Shalfan
Artificial Intelligence Lab, Imam University. Riyadh. KSA.
Email: elarbi-boudihir@lycos.com

ABSTRACT
Video security is becoming more and more vital today
due to rapid development of hardware equipments as
the number of installed cameras can confirm. This
paper presents the system architecture of IVSS, an
Intelligent Video Surveillance System based on IP-
cameras and deployed in an academic environment. In
fact, Video surveillance is increasingly found in
academic institutions. It is used to oversee the safety of
teachers and students, as well as to protect assets from
vandalism and theft. In this work the surveillance
system is deployed in our university environment, it is
based on a set of digital IP video cameras linked by the
university IP network infrastructure. Our system
architecture is based on efficient moving object
detecting and tracking algorithm and a robust statistical
activity recognition framework based on SVM which is
used for modeling activities. The experimental results
on real-time video streams show the feasibility of our
system and its effectiveness in human activity
recognition.

Keywords: Abnormal behavior, real-time detection,
computer vision, intelligent visual surveillance,
tracking.

1. INTRODUCTION
Modern video surveillance systems gained
attention in the wider community of computer
vision more than a decade ago. Today, the issue
receives more intense pursuit from the narrower
but more focused visual surveillance community.
Automated video surveillance systems constitute a
network of video sensors observing people as well
as other moving and interacting objects in a given
environment for patterns of normal/abnormal
activities, interesting events, and other domain-
specific goals. On the other hand, the problem of
robust object detection and tracking is even harder
to address given the requirement that the video
surveillance systems have to operate in widely
varying weather conditions and all time periods.
This situation of high performance expectations
and stringent requirements places a minimal
margin of error on the performance of these video
surveillance systems. The objective of paper is to
describe the development of an intelligent
surveillance system for urban security in an
academic environment. This prototype system
incorporates a wide range of advanced
surveillance techniques: real-time moving object
detection and tracking from stationary camera
platforms, recognition of generic object classes
and specific human abnormal behavior triggering
an alarm, object pose estimation with respect to a
geospatial site model, camera control and multi-
camera cooperative tracking, human activity
recognition and analysis, recognition of simple
multi-agent activities, real-time data
dissemination, data logging and dynamic scene
visualization. The proposed architecture takes
advantage of time-varying data from multiple
cameras to obtain point correspondences and
perform robust calibration. It tracks a moving
object in the scene and uses its location at every
time step as a single point correspondence among
multiple cameras. This paper describes the
architecture and the performances of the IVSS by
introducing the state of the art in section 2. In
section 3 the system architecture is detailed and
section 4 illustrates the visual data representation
in the system after applying the object detection,
recognition and tracking algorithms from one
camera. The extension to a multi-camera
representation is given by section 5. Finally,
section 6 presents the activity recognition process
of our system.

2. STATE OF THE ART
Growing security concerns, increasing crime rates
and terrorist activity, as well as an increasing
general demand for more protection of people and
assets are affecting the growth of the security and,
more specifically, the video surveillance market
[1]-[4]. Potential application areas range from
home monitoring, elderly care, and smart environments to security and surveillance in public or corporate buildings. Computer vision based solutions have the potential for very discriminating detection and very low false alarms [5]-[6]. Surveillance systems are typically categorized into three distinct generations [7]. The first generation uses analog equipment throughout the complete system. The second generation uses analog cameras but digital back-end equipment. The third generation completed the digital transformation and the cameras which have the ability to convert the video signal to digital before sending them sometimes over IP [8]. Over the last 2 decades, research in computer vision has been very active where scene interpretation and understanding received the lion’s share from the scientific community effort in this field [9]. That is mostly due to the specific interest of governments in automatic video surveillance for homeland security. That orientation was largely helped by the hardware (cameras and computers) becoming cheaper. Consequently, many projects were started and aimed to develop intelligent video surveillance systems like CROMATICA [10], VSAM [11], or ADVISOR [12]. However, despite investigators hard work, it is clear that a big effort is yet to be done before developing surveillance related systems that are really useful [13]. Robustness of activity detection, tracking and understanding modules, is one of the crucial problems still to be investigated in a more systematic manner [14]. These projects were started in order to build an Intelligent Visual System which will add a brick towards solving the problem of robustness. Other known problems, like handling occlusions [15] may also be investigated.

3. SYSTEM ARCHITECTURE

Video surveillance is increasingly found in academic institutions. It is used to oversee the safety of faculty members, staff and students, as well as to protect assets from vandalism and theft. Moreover, the campuses may be extensive, especially in the case of universities, and be comprised of several buildings, accesses and parking lots to monitor. In this environment, video surveillance is used in particular to:

- monitor access to the institution’s perimeter;
- monitor equipment and data;
- detect and follow acts of vandalism and theft;
- recognize license plates;
- support criminal investigations and control access.

Since educational institutions often have an IP network infrastructure, it is beneficial to set up digital video surveillance systems [16]. Due to the above reasons, we have implemented our IVSS in our University for testing. Basically, the system is composed of a set of IP cameras plugged directly in the local network hub. A human computer interface and a storage space are also plugged in this system. The main advantage of such architecture is its flexibility. The main goal is to create a robust, adaptive system that is flexible enough to handle variations in lighting, moving scene clutter, multiple moving objects and other arbitrary changes to the observed scene. Consequently, in this architecture, the system should be able to:

i) Register different viewpoints and create virtual displays of the facility or area;
ii) Track and classify objects
iii) Overlay tracking information on a virtual display constructed from the observations of multiple cameras;
iv) Learn standard and abnormal behaviors of objects;
v) Selectively store video. Low bandwidth tracking information could be continually stored allowing the observer to query the system about activities.

The architecture enables a single human operator to monitor activities over a broad area using a distributed network of video sensors. The sensor platforms are mainly autonomous, notifying the operator only of salient information as it occurs, and engaging the operator minimally to alter platform operations. And we believe that developing the capabilities to deploy and most importantly to process the data from such a big number of cameras will impact existing surveillance and monitoring methods. The architecture of our proposed system focuses on a reliable link between image processing and video content analysis as seen on figure 3.1. Hence,
integration of image processing within the digital video networked surveillance system itself is inevitable. The proposed IVSS system contains all the modules (video capture, image analysis, image understanding, event generator and field experience). Moreover, it contains an autolearning module and another module about video retrieval.

The video capture module is responsible of managing the video input data from different IP-cameras over a LAN where each camera can be accessed by its IP address. Accordingly, this module generates report about failures in the video capture process or in the network itself. The image analysis module includes all the image processing tasks applied on the video stream to extract relevant information such as motion detection, tracking, etc.. Moreover, the image understanding module represent the master piece of the IVSS, it includes all AI techniques to figure out the meaning of the scene. Among its tasks: detecting abnormal behavior of human and other moving objects in the scene. The abnormal behavior is forwarded to the event generator module, which generates an alarm for the user and helps the image analysis module to tune the image processing tasks to enhance the behavior for easier perception and monitoring. The detected events based on abnormal behaviors can be modeled and stored in the field experience module for easier access and future detection [16].

In the context of this architecture, we have build on existing framework for detecting, tracking and classifying activities, in a variety of directions. Tracking methods will be extended to incorporate multiple cameras. This will require coordination between the cameras to ensure that the same object is being tracked in each, as well as to merge statistical information about the tracked object into a coherent framework.

4. VISUAL DATA REPRESENTATION

In video surveillance object detection and tracking constitute the low level building block necessary for any surveillance system. Thus, detecting changes in a camera video-stream is the basis for all intelligent analysis. It may detect an activity in a scene under surveillance, in particular the movement of objects. It may also reveal the appearance or disappearance of an object (abandoned or stolen object). Many techniques for detecting movements used in video processing are based on detecting changes [16]-[17]. However, detecting changes in video does not specifically target the movement of objects, but may highlight an image modulation.

4.1 Object Detection Techniques

Object detection aims at segmenting regions corresponding to moving objects such as vehicles and humans from the rest of an image. Detecting moving regions provides a focus of attention for later processes such as tracking and behavior analysis because only these regions need be considered in the later processes. There are four main conventional approaches to object detection: background subtraction, temporal difference, optical flow and active camera. In our system, moving objects are detected in a video stream using temporal differencing. Targets are then classified according to a metric classification. These targets can be tracked using a combination of motion information and image based correlation. In the first stage, all moving objects are detected using temporal differencing algorithm. These are described as Motion regions as illustrated by Figure 4.1 in the top right images of a, b, c and d where the motion region is located...
by a green frame tracking in real-time the moving person. The video capture module delivers a video stream acquired from the camera, and then each I frame of the stream is smoothed with the second derivative in time of the temporal Gaussian function.

If \( f_n \) is the intensity of the \( n \)th I frame of the shot, then the absolute difference function \( \Delta_n \) is:

\[
\Delta_n = | f_n - f_{n-1} |
\]

The result of the difference is binarized in order to separate changed pixels from others. To do this, a threshold function is used and a motion image \( M_n \) can be extracted.

\[
M_n(u,v) = \begin{cases} 
  f_n(u,v) & \text{if } \Delta_n(u,v) \geq T \\ 
  0 & \text{if } \Delta_n(u,v) < T 
\end{cases}
\]

Where \( T \) is an appropriate threshold chosen after several tests performed on the scenes of the environment. To separate the regions of interest from the rest of image, binary statistical morphological operators (erosion and dilatation) are used as follows

- Binary statistical erosion: if the structured element \( SE \) and the filtering threshold \( th \) are fixed, the output of binary statistical erosion at pixel \( i \) is:

\[
f_e(i) = \begin{cases} 
  1 & M^1(i) \geq th \\ 
  0 & \text{otherwise} 
\end{cases}
\]

Where \( M^1(i) \) is the number of pixels of value 1 inside the SE. It allows eliminating the noisy isolated pixels returned by the change pixels detector.

- Binary mathematical dilatation: if the structuring element \( SE \) is fixed, the output of binary dilatation at pixel \( i \) is:

\[
f_d(i) = \begin{cases} 
  1 & M^1(i) \geq 1 \\ 
  0 & \text{otherwise} 
\end{cases}
\]

This operation allows recovering interesting pixels eliminated by erosion, by filling holes present inside interesting regions. Then, the moving sections must be grouped into motion regions \( R_n(i) \). This is done using a connected component criterion. It allows to group different motion sections susceptible to be a part of the same region, or allows grouping the residual motion parts into one motion region. This propriety is useful to identify a human who are not rigid and also useful in occultation of the moving object and other target. After the motion region is determined, targets are morphologically dilated (twice) and then eroded. Subsequently, moving targets are clustered into motion regions using a connected components criterion.

The algorithm works effectively and satisfactorily when the scene includes many moving objects or humans. Each time a person enters the scene, the system encapsulates the moving body shape by a numbered frame for proper tracking through time. The multi-object motion detection is illustrated by figure 4.2 where we have tried two persons coming towards each other then passing nearby each other. The system shows the two numbered frames coming closer, merging, and then separating again.
4.2 Moving Object Fuzzy Classification

The main difficulty with metrical classification is that for example, multiple humans close together can be misclassified as vehicles, or a partly occluded vehicle may look like a human, or some background clutter may appear as a vehicle. To overcome this problem, an additional hypothesis is used. The main idea is to record all potential motion regions PR\textsubscript{n} from the first frame of the stream. Each one of these potential regions must be observed along some frames of the shot to determine if they persist or not, and so decide to continue classifying them. To do this, for each new frame, each previous motion region PR\textsubscript{n-1} is matched to the spatially closest current motion region R\textsubscript{n} according to a mutual proximity rule. After this process, each previous potential motion region PR\textsubscript{n-1} which have not been matched to current region are removed from the list of accepted motion regions. And any current motion region R\textsubscript{n} which has not been matched is considered new potential region. At each frame, their new classification according to the metric operators, dispersion and ratio, are used to update the classification hypothesis. The most advantage of this method is that if an occluded object is misclassified it will be correctly classified with the passage of time. Another advantage is that the unstable motions appearing at the background will be misclassified as no-identified regions.

The motivation of the use of the geometry features is that is computationally inexpensive and invariant to lighting conditions or viewpoint [18]. On the other hand, it is clear that the human, with its small and more complex shape, will have larger dispersion than a vehicle. If we define an appropriate membership function µ for the object, the area and the perimeter p of the object can be calculated as follows: Area of fuzzy sets:

\[
a(\mu) = \sum \mu
\]

Perimeter of a fuzzy set:

\[
p(\mu) = \sum_{i=1}^{M} \sum_{n=1}^{N} |\mu_{m,n} - \mu_{m+1,n}| + \sum_{n=1}^{N} \sum_{i=1}^{M-1} |\mu_{m,n} - \mu_{m+1,n}|
\]

Where M and N are the dimensions of the image.

Based on the perimeter and the area, the dispersion and the ratio of a fuzzy set can be determined as follows:

\[
\text{Dispersion} = \frac{(\text{Perimetre})^2}{\text{Area}}, \quad \text{Ratio} = \frac{\text{Length}}{\text{width}}
\]

The classified motion regions are used as templates for metrical training algorithms. The fuzzy system is based on two entrances: the dispersion and the ratio of the motion regions, and three exits: one exit for human, one exit for the vehicles and one exit for non-identified objects. For every entrance, we have two fuzzy sets: one for the category of humans and other for the category of vehicles. It’s clear that the most obvious types of targets which will be of interest in our IVSS application are Humans and Vehicles. For the time being we did not assign any outdoor camera for vehicle tracking, but this issue is among the future research objectives. So we set up the classification without testing vehicles for the time being. Many experiments have been conducted to evaluate the range of the ratio and dispersion for humans and vehicles. For the sake of meeting Saudi standards, we always experiment with people wearing Saudi clothes beside the ones wearing western clothes. For this reason, two classifiers to detect these two groups have been implemented. The metric is based on the knowledge that humans are, in general, smaller than vehicles, and that they have more complex shapes.

4.3 Object Tracking Approach

Many tracking techniques are based on mathematical methods that make it possible to predict an object’s position on a frame based on its movement in the previous frames. Tracking several objects at the same time poses many challenges. Each object detected in a frame must be associated with its corresponding object in the subsequent frame. This matching is done based on the objects characteristics (e.g., corners, area, ratios, etc.), or their model of appearance. Occlusions (regions hidden by others) represent a major difficulty for tracking objects. A video surveillance system may lose track of an object if it is totally or partially obstructed over a certain period of time. The known difficulties in object
tracking which remain largely open problems could arise from: abrupt object motion, changing appearance of objects and scenes, self-occlusion, and occlusion by structure.

Once objects have been detected, the next logical step is to track these detected objects. Tracking has a number of benefits. Firstly, the detection phase is quite computationally expensive, so by using tracking, the detection step does not need to be computed for each frame. After detecting moving objects, the IVSS track their movement over the video stream. Each task requires locating each object tracked from one image to another. The Kalman filter is another powerful tool for analyzing motion. The filter can be used to predict the real position of the blob being tracked at a better accuracy than raw sensor data. The Kalman filter uses the history of measurements to build a model of the state of the system that maximizes the probability for the position of the target based on the past measurements [20].

5. EXTENSION OF THE TRACKING METHOD TO MULTIPLE CAMERAS

The aim of tracking description in multiple-camera configuration is to make a link between the tracking and the analysis processes. It is then important to establish correspondences between the objects in different image sequences taken by different cameras. Consequently, target tracking and data fusion also need to be addressed. The success of data fusion depends on how well data is represented, how reliable and adequate the model of data used and how accurate and applicable prior knowledge is. Figure 5.1 shows the environment of our IVS which had been implemented in the college of computer science. Camera calibration seems to be a necessary step to make it possible to calculate the actual size and speed of the objects in the scene. It establishes the correspondence between the geometry of the scene and that of the image. For fixed cameras, a 2D context can be defined by the system administrator identifying areas in the image such as input/output regions, zones to ignore, etc.

The interface used in our IVSS is shown by Figure 5.2 thereby the operator can have a general view of what happening in the area under surveillance. The two cameras of type CIVS-IPC-3431 (denoted camera K and L) were installed in the server room and just in the nearby corridor for the purpose of identifying persons accessing the server room and checking for access rights. While the ten cameras of type CIVS-IPC-4300 have been installed in the corridors of the first floor of the department to cover a wide closed area where students move and access the lecture rooms, faculty offices, administration offices and toilets. The ten cameras were denoted A, B, C, D, E, F, G, H, I and J as illustrated by figure 5.1. The idea is to create an interface with a mosaic of all available cameras and when clicking on an image, we can see it in a bigger size or in full screen mode. While designing the interface, we had several choices depending on the development language we are going to choose. With Java language, we have the Swing library for developing desktop application. With C++, we have MFC, GTK or QT Framework, which offer all a complete SDK for developing portable cross-platform application especially GTK or QT. Accordingly, Java being a higher-level language, we prefer C++ for an intensive resource consuming application like video processing.
The tracking analysis is a process that generates predefined patterns like objects entering from a defined zone of the image and exiting by another one, or objects which have exceeded a certain speed limit, or also stopped objects for a minimum time which stem from another mobile object. After detecting the motion and tracking the object from frame to frame, it would be interesting to know in which camera the moving object will probably appear after it has disappeared from a given one. This will make the object tracking process easy for the operator in a multi-camera surveillance system. First, we notice that in each view the tracked object will exit from the scene in four different ways: from the left (Left Exit: LE) (Figure 5.3.a), from the right (Right Exit: RE) (Figure 5.3.b), from the bottom (Bottom Exit: BE) (Figure 5.3.c), from the top (Top Exit: TE) (Figure 5.3.d), and vanishing point (Vanishing point Exit: VE), (Figure 5.3.e). Note that, the top exit (TE) and the vanishing point exit (VE) provide the same conclusion. In fact, the top exit takes place when the camera is pointing downward from the horizontal axis. If the camera viewing axis is set horizontal, the TE and VE would be identical. The reason for pointing a bit downward the viewing camera is to cover more details of the near field of view.

Any moving object exiting from a camera field of view is very likely to appear in another one, if that object does not leave the global area under surveillance. It is then essential for tracking analysis to establish a link between the different zones.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.1: Camera link table for tracking analysis based on left exit.
After detecting blobs on the monitored area, the next step was to represent the evolution of this blobs on a Map in real-time. Bottom, top and top and VP exits can easily be derived. A Map of the area was drawn for this purpose. The position of the different cameras is visible on the map and each camera was assigned a letter to represent it. Once the camera scale is fixed, each camera tracker is given the coordinates of the blobs detected and it will be displayed on the map as shown on Figure 5.4.

![Figure 5.4: Display on the map (red point) of a tracked person (shown at bottom left) from one of the camera](image)

### 6. ACTIVITY RECOGNITION

In this section we will proceed to evaluation of the most important methods used in, either the abstraction or event modeling phases independently on the taxonomy used. However, we will use the taxonomy proposed by Lavee et al [21], only as indication for categorizing the methods used. The objective of this section is to show the strengths and shortages of some of the most important methods used which will help investigators choose their “tools” depending on their “problems”.

The traditional techniques which fall under this title focus on the event recognition problem and do not consider the semantic representation (understanding), which makes their role simple, thus realizing good results in their scope. Minimal semantic knowledge is needed in building the event classifiers in this category using techniques like “Nearest Neighbor”, Support Vector Machines and Neural Networks. Often, they may be fully specified from training data. These techniques are often applied to the abstraction stage. The methods mentioned above all involve supervised learning. They are applicable for known scenes where the types of object motions are already known. Another member of the neural networks family, namely the self-organizing neural networks (like Kohonen networks), is suited to behavior understanding when the object motions are unrestricted. Among the abnormal behavior, we tackled those based on two parameters: Existence or not of interaction between objects (humans here) and the event being normal or abnormal (falling or running person, a punch (involving 2 persons) and a rushing crowd in the wrong direction during a given time. As the output is binary in all the scenarios, the statistical method chosen to discriminate the events is the Support Vector Machine. As a machine has to make the decisions, the broad “machine learning” topic pops up. As “OpenCV2” comes with a machine learning library “ml.lib”, we decided to use its C++ API to implement the scenarios.

#### 6.1 Support Vector Machine (SVM)

The basic idea of Support Vector Machines is to find the optimal hyperplane that splits a dataset into different categories. That hyperplane is chosen, so that the distance to the nearest data point of the classes is maximized. The following figure gives an idea about a simple example with only 2 categories in the plane.

![Figure 6.1: The red line (H2) is the optimal in this example](image)

Globally, it is seen as a set of supervised learning methods that analyze data and recognize patterns. It takes a set of input data and predicts, for each given input, which of two possible classes forms the input, making the SVM a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as
wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on [22]. In fact, using the geometry of the frame associated with the detected motion of the recognized object, we may categorize some basic activities like running, jumping and falling:

**Running:** may be detected when the speed of the frame from image to image goes beyond a predetermined threshold. In fact, the speed value less than this threshold characterize “normal” walking. The result is shown on Figures 6.2 below. This case depicts running to the right or the left but parallel to image plane. Detecting the running behavior away or towards the camera is still under implementation.

![Figure 6.2: Detection of the running behavior](image)

**Jumping:** This case is detected when the position of the frame from image to image suddenly goes up then down. Moreover, the speed of this up and down motion should be greater than a predetermined threshold, to not confuse with the normal displacement of the frame during the motion detection process. Figures 6.3 show a detection of a human jump behavior.

![Figure 6.3: Detection of jump behavior](image)

**Falling:** This behavior can easily be detected for a single person when the size of the frame suddenly changes its dimensions and becomes “squeezed” downward. Generally, the center of the tracking frame should go suddenly down relatively to the previous position. Figure 6.4 shows the detection of falling and running behavior simultaneously.

![Figure 6.4: Detection of falling (a) and running (b) behaviors](image)

Generally, most of the research done in the field of IVSS concerns mostly people wearing western clothes. This constitutes, in fact; another problem to be solved in our case, which is discerning the person wearing a white Saudi wear from the mostly white background of our environment in the college. The difficulties that we faced during this first phase of system implementation are part of the problems linked to the third generation of video systems which are multiple and need to be addressed quickly in order to push this technology to maturation.

7. **CONCLUSION**

Feature extraction and classification, even if investigations last more than 2 decades now, remain a big challenge. Many methods were used to detect moving objects like background subtraction and others. However, each of them present drawbacks like “ghosts” for the background subtraction method. We may notice in the experimental results, presented through the different sections of this paper that sometimes only two persons are present in the scene but frames are not consecutively numbered. This is mainly due to feature extraction and classification algorithm which sometimes classifies a shadow as another moving person and assigns to it another frame number. But, after a couple of seconds the frame disappears automatically due to the fact that the tracking does support the sudden deformation of the frame. In fact, finding the events of interest and identifying the behavior is not a trivial task. This is may be the bigger challenge in our IVSS. Many approaches are presented but harder work is yet to be done. The computational cost for some methods is very high which make their use difficult. Most of the Saudis are wearing white...
dress during most of the time of the year. Hence, discerning the moving person from the mostly white background of our environment in the college was a very tedious task. Actually, we shifted to the red head cover detection, but still this head cover may be white for many Saudis. At the beginning, we tried rising the threshold value used for motion image segmentation, but soon we discover that it causes the shadow and some minor light changes like a foreground blob that is moving.

Acknowledgement: The authors would like to thank the King Abdul-Aziz City for Science and Technology (KACST) for support funding to carry out this research project AT/29/314.

8. REFERENCES


